

# Spartan-3E ライブラリ ガイド (HDL 用)

UG617 (v12.2) 2010 年 7 月 23 日



Xilinx is disclosing this user guide, manual, release note, and/or specification (the “Documentation”) to you solely for use in the development of designs to operate with Xilinx hardware devices. You may not reproduce, distribute, republish, download, display, post, or transmit the Documentation in any form or by any means including, but not limited to, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Xilinx. Xilinx expressly disclaims any liability arising out of your use of the Documentation. Xilinx reserves the right, at its sole discretion, to change the Documentation without notice at any time. Xilinx assumes no obligation to correct any errors contained in the Documentation, or to advise you of any corrections or updates. Xilinx expressly disclaims any liability in connection with technical support or assistance that may be provided to you in connection with the Information.

THE DOCUMENTATION IS DISCLOSED TO YOU “AS-IS” WITH NO WARRANTY OF ANY KIND. XILINX MAKES NO OTHER WARRANTIES, WHETHER EXPRESS, IMPLIED, OR STATUTORY, REGARDING THE DOCUMENTATION, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT OF THIRD-PARTY RIGHTS. IN NO EVENT WILL XILINX BE LIABLE FOR ANY CONSEQUENTIAL, INDIRECT, EXEMPLARY, SPECIAL, OR INCIDENTAL DAMAGES, INCLUDING ANY LOSS OF DATA OR LOST PROFITS, ARISING FROM YOUR USE OF THE DOCUMENTATION.

© Copyright 2002–2010 Xilinx Inc. All Rights Reserved. XILINX, the Xilinx logo, the Brand Window and other designated brands included herein are trademarks of Xilinx, Inc. All other trademarks are the property of their respective owners.

本資料は英語版 (v.12.2) を翻訳したもので、内容に相違が生じる場合には原文を優先します。  
資料によっては英語版の更新に対応していないものがあります。  
日本語版は参考用としてご使用の上、最新情報につきましては、必ず最新英語版をご参照ください。

# このマニュアルについて

---

HDL 用ライブラリ ガイドは、ISE® のオンライン マニュアルの 1 つです。回路図を使用して設計する場合は、回路図用ライブラリ ガイドを参照してください。

このマニュアルには、次の内容が含まれます。

- ・ 概要
- ・ このアーキテクチャでサポートされるプリミティブとマクロのファンクション別リスト
- ・ 各プリミティブの詳細説明

## デザイン エLEMENT

このバージョンのライブラリ ガイドでは、このアーキテクチャのザイリンクス ユニファイド ライブラリを構成するプリミティブとインスタンス化コード例が含まれます。また、インスタンス化 テンプレートは、ISE/doc/usenglish/isehelp のインストール ディレクトリにも個別 ZIP ファイルとして含まれています。

プリミティブは、ターゲットにしている FPGA デバイス用のザイリンクス コンポーネントです。プリミティブをインスタンス化して変換プロセスを実行すると、変換後のファイルには同一のコンポーネントが含まれます。たとえば、ISERDES\_NODELAY という Virtex-5 エLEMENTをユーザープリミティブとしてインスタンス化し、変換 (NGDBuild) を実行すると、ISERDES\_NODELAY がそのまま残ります。一方 Virtex-5 デバイスで ISERDES を使用していると、自動的に Virtex-5 用の ISERDES\_NODELAY に変換されます。このため、「プリミティブ」の概念は、同じ分野でもユーザーによって異なります。

ザイリンクスでは、さまざまなデバイス アーキテクチャに対応した多数のデザイン エLEMENT (ユニマクロおよびプリミティブ) を含むソフトウェア ライブラリを提供しています。開発システムソフトウェアのリリースごとに、新しいデザイン エLEMENTが組み込まれます。このガイドは、デザイン エLEMENTをアーキテクチャごとに説明しているライブラリ ガイド シリーズの 1 つです。

## デザインの入力方法

このガイドでは、各デザイン エLEMENT で 4 つの使用方法を評価して、その中から最適なソリューションを示します。次にこの 4 つの使用方法を示します。

- ・ **インスタンス化**：デザインにコンポーネントが直接インスタンス化されます。これは、各ブロックをユーザーが正確に配置する場合に有効な方法です。
- ・ **推論**：コンポーネントはサポートされる合成ツールで推論されます。コードは柔軟性およびポータビリティに優れているので、複数のアーキテクチャに使用できます。推論を実行すると、パフォーマンス、エリア、電力などをユーザーが合成ツールで指定したとおりに最適化できます。
- ・ **CORE Generator およびウィザード**：コンポーネントは CORE Generator またはウィザードから使用できます。推論ができない FPGA を使用して大型なブロックを構築する場合には、この方法を使用してください。このフローを使用する場合は、ターゲットにするアーキテクチャごとにコアを再生成する必要があります。
- ・ **マクロのサポート**：使用可能な UniMacro があります。これらのコンポーネントはザイリンクス ツールの UniMacro ライブラリに含まれ、プリミティブだけでは複雑すぎてインスタンス化しにくいようなプリミティブをインスタンス化する際に使用します。合成ツールでは、この UniMacro が自動的に下位プリミティブに展開されます。

# 目次

---

このマニュアルについて.....	3
デザイン エlement .....	3
デザインの入力方法 .....	4
1: ファンクション別分類 .....	9
2: デザイン エlement .....	17
BSCAN_SPARTAN3 .....	18
BUFCF .....	21
BUFG .....	23
BUFGCE .....	25
BUFGCE_1 .....	27
BUFGMUX .....	29
BUFGMUX_1 .....	31
CAPTURE_SPARTAN3 .....	33
DCM_SP .....	35
FDCE .....	42
FDCE_1 .....	44
FDCPE .....	45
FDCPE_1 .....	48
FDRSE .....	50
FDRSE_1 .....	52
IBUF .....	54
IBUFDS .....	56
IBUFG .....	59
IBUFGDS .....	61
IDDR2 .....	63
IOBUF .....	66
IOBUFDS .....	69
KEEPER .....	72
LDCPE .....	74
LUT1 .....	77
LUT1_D .....	79
LUT1_L .....	81
LUT2 .....	83
LUT2_D .....	85
LUT2_L .....	87

LUT3 .....	89
LUT3_D .....	92
LUT3_L .....	95
LUT4 .....	98
LUT4_D .....	101
LUT4_L .....	104
MULT_AND.....	107
MULT18X18.....	109
MULT18X18S .....	111
MULT18X18SIO.....	113
MUXCY .....	116
MUXCY_D.....	118
MUXCY_L.....	120
MUXF5.....	122
MUXF5_D .....	124
MUXF5_L.....	126
MUXF6.....	128
MUXF6_D .....	130
MUXF6_L.....	132
MUXF7.....	134
MUXF7_D .....	136
MUXF7_L.....	138
MUXF8.....	140
MUXF8_D .....	142
MUXF8_L.....	144
OBUF .....	146
OBUFDS.....	148
OBUFT .....	150
OBUFTDS .....	153
ODDR2.....	155
PULLDOWN .....	158
PULLUP .....	160
RAM16X1D .....	162
RAM16X1D_1 .....	165
RAM16X1S.....	169
RAM16X1S_1 .....	171
RAM16X2S.....	173
RAM16X4S.....	176

RAM16X8S.....	178
RAM32X1S.....	180
RAM32X1S_1 .....	183
RAM32X2S.....	186
RAM32X4S.....	189
RAM32X8S.....	191
RAM64X1S.....	193
RAM64X1S_1 .....	195
RAM64X2S.....	198
RAMB16_S1 .....	200
RAMB16_S1_S1.....	206
RAMB16_S1_S18.....	217
RAMB16_S1_S2.....	229
RAMB16_S1_S36.....	240
RAMB16_S1_S4.....	252
RAMB16_S1_S9.....	264
RAMB16_S18 .....	276
RAMB16_S18_S18 .....	283
RAMB16_S18_S36 .....	295
RAMB16_S2 .....	307
RAMB16_S2_S18.....	313
RAMB16_S2_S2.....	321
RAMB16_S2_S36.....	333
RAMB16_S2_S4.....	345
RAMB16_S2_S9.....	357
RAMB16_S36 .....	369
RAMB16_S36_S36 .....	376
RAMB16_S4.....	388
RAMB16_S4_S18.....	394
RAMB16_S4_S36.....	406
RAMB16_S4_S4.....	418
RAMB16_S4_S9.....	429
RAMB16_S9.....	441
RAMB16_S9_S18.....	448
RAMB16_S9_S36.....	460
RAMB16_S9_S9.....	472
ROM128X1 .....	484
ROM16X1 .....	487

ROM256X1 .....	490
ROM32X1 .....	493
ROM64X1 .....	496
SRL16 .....	499
SRL16_1.....	501
SRL16E .....	503
SRL16E_1.....	506
SRLC16 .....	508
SRLC16_1 .....	511
SRLC16E .....	513
SRLC16E_1.....	516
STARTUP_SPARTAN3E.....	519
XORCY .....	521
XORCY_D .....	523
XORCY_L .....	525



## ファンクション別分類

このセクションでは、デバイスに含まれるデザイン エLEMENTをファンクション別に分類して示します。ELEMENT (プリミティブおよびマクロのインプリメンテーション) は、各カテゴリでアルファベット順にリストしています。

四則演算ファンクション	RAM/ROM
クロック コンポーネント	レジスタおよびラッチ
コンフィギュレーション/BSCAN コンポーネント	シフト レジスタ
I/O コンポーネント	スライス/CLB プリミティブ

### 四則演算ファンクション

デザイン エLEMENT	説明
MULT18X18	プリミティブ : 18 x 18 Signed Multiplier
MULT18X18S	プリミティブ : 18 x 18 Signed Multiplier — Registered Version
MULT18X18SIO	プリミティブ : 18 x 18 Cascadable Signed Multiplier with Optional Input and Output Registers, Clock Enable, and Synchronous Reset

### クロック コンポーネント

デザイン エLEMENT	説明
BUFG	プリミティブ : Global Clock Buffer
BUFGCE	プリミティブ : Global Clock Buffer with Clock Enable
BUFGCE_1	プリミティブ : Global Clock Buffer with Clock Enable and Output State 1
BUFGMUX	プリミティブ : Global Clock MUX Buffer
BUFGMUX_1	プリミティブ : Global Clock MUX Buffer with Output State 1
DCM_SP	プリミティブ : Digital Clock Manager
IBUFG	プリミティブ : Dedicated Input Clock Buffer
IBUFGDS	プリミティブ : Differential Signaling Dedicated Input Clock Buffer and Optional Delay

### コンフィギュレーション/BSCAN コンポーネント

デザイン エLEMENT	説明
BSCAN_SPARTAN3	プリミティブ：Spartan®-3 and Spartan-3E JTAG Boundary Scan Logic Access Circuit
CAPTURE_SPARTAN3	プリミティブ：Spartan®-3 Register State Capture for Bitstream Readback
STARTUP_SPARTAN3E	プリミティブ：Spartan®-3E User Interface to the GSR, GTS, Configuration Startup Sequence and Multi-Boot Trigger Circuitry

### I/O コンポーネント

デザイン エLEMENT	説明
IBUF	プリミティブ：Input Buffer
IBUFDS	プリミティブ：Differential Signaling Input Buffer
IBUFG	プリミティブ：Dedicated Input Clock Buffer
IBUFGDS	プリミティブ：Differential Signaling Dedicated Input Clock Buffer and Optional Delay
IOBUF	プリミティブ：Bi-Directional Buffer
IOBUFDS	プリミティブ：3-State Differential Signaling I/O Buffer with Active Low Output Enable
KEEPER	プリミティブ：KEEPER Symbol
OBUF	プリミティブ：Output Buffer
OBUFDS	プリミティブ：Differential Signaling Output Buffer
OBUFFT	プリミティブ：3-State Output Buffer with Active Low Output Enable
OBUFFTDS	プリミティブ：3-State Output Buffer with Differential Signaling, Active-Low Output Enable
PULLDOWN	プリミティブ：Resistor to GND for Input Pads, Open-Drain, and 3-State Outputs
PULLUP	プリミティブ：Resistor to VCC for Input PADS, Open-Drain, and 3-State Outputs

### RAM/ROM

デザイン エLEMENT	説明
RAM16X1D	プリミティブ：16-Deep by 1-Wide Static Dual Port Synchronous RAM
RAM16X1D_1	プリミティブ：16-Deep by 1-Wide Static Dual Port Synchronous RAM with Negative-Edge Clock
RAM16X1S	プリミティブ：16-Deep by 1-Wide Static Synchronous RAM

デザイン エLEMENT	説明
RAM16X1S_1	プリミティブ：16-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock
RAM16X2S	プリミティブ：16-Deep by 2-Wide Static Synchronous RAM
RAM16X4S	プリミティブ：16-Deep by 4-Wide Static Synchronous RAM
RAM16X8S	プリミティブ：16-Deep by 8-Wide Static Synchronous RAM
RAM32X1S	プリミティブ：32-Deep by 1-Wide Static Synchronous RAM
RAM32X1S_1	プリミティブ：32-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock
RAM32X2S	プリミティブ：32-Deep by 2-Wide Static Synchronous RAM
RAM32X4S	プリミティブ：32-Deep by 4-Wide Static Synchronous RAM
RAM32X8S	プリミティブ：32-Deep by 8-Wide Static Synchronous RAM
RAM64X1S	プリミティブ：64-Deep by 1-Wide Static Synchronous RAM
RAM64X1S_1	プリミティブ：64-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock
RAM64X2S	プリミティブ：64-Deep by 2-Wide Static Synchronous RAM
RAMB16_S1	プリミティブ：16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 1-bit Port
RAMB16_S1_S1	プリミティブ：16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit Ports
RAMB16_S1_S18	プリミティブ：16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 18-bit Ports
RAMB16_S1_S2	プリミティブ：16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 2-bit Ports
RAMB16_S1_S36	プリミティブ：16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 36-bit Ports
RAMB16_S1_S4	プリミティブ：16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 4-bit Ports
RAMB16_S1_S9	プリミティブ：16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 9-bit Ports
RAMB16_S18	プリミティブ：16K-bit Data + 2K-bit Parity Memory, Single-Port Synchronous Block RAM with 18-bit Port
RAMB16_S18_S18	プリミティブ：16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 18-bit Ports
RAMB16_S18_S36	プリミティブ：16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 18-bit and 36-bit Ports
RAMB16_S2	プリミティブ：16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 2-bit Port
RAMB16_S2_S18	プリミティブ：16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 18-bit Ports
RAMB16_S2_S2	プリミティブ：16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit Ports
RAMB16_S2_S36	プリミティブ：16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 36-bit Ports

デザイン エLEMENT	説明
<a href="#">RAMB16_S2_S4</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 4-bit Ports
<a href="#">RAMB16_S2_S9</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 9-bit Ports
<a href="#">RAMB16_S36</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 36-bit Port
<a href="#">RAMB16_S36_S36</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with Two 36-bit Ports
<a href="#">RAMB16_S4</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 4-bit Port
<a href="#">RAMB16_S4_S18</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit and 18-bit Ports
<a href="#">RAMB16_S4_S36</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit and 36-bit Ports
<a href="#">RAMB16_S4_S4</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit Ports
<a href="#">RAMB16_S4_S9</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit and 9-bit Ports
<a href="#">RAMB16_S9</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 9-bit Port
<a href="#">RAMB16_S9_S18</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 9-bit and 18-bit Ports
<a href="#">RAMB16_S9_S36</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 9-bit and 36-bit Ports
<a href="#">RAMB16_S9_S9</a>	プリミティブ : 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 9-bit Ports
<a href="#">ROM128X1</a>	プリミティブ : 128-Deep by 1-Wide ROM
<a href="#">ROM16X1</a>	プリミティブ : 16-Deep by 1-Wide ROM
<a href="#">ROM256X1</a>	プリミティブ : 256-Deep by 1-Wide ROM
<a href="#">ROM32X1</a>	プリミティブ : 32-Deep by 1-Wide ROM
<a href="#">ROM64X1</a>	プリミティブ : 64-Deep by 1-Wide ROM

## レジスタおよびラッチ

デザイン エLEMENT	説明
FDCE	プリミティブ：D Flip-Flop with Clock Enable and Asynchronous Clear
FDCE_1	プリミティブ：D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Clear
FDCPE	プリミティブ：D Flip-Flop with Clock Enable and Asynchronous Preset and Clear
FDCPE_1	プリミティブ：D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Preset and Clear
FDRSE	プリミティブ：D Flip-Flop with Synchronous Reset and Set and Clock Enable
FDRSE_1	プリミティブ：D Flip-Flop with Negative-Clock Edge, Synchronous Reset and Set, and Clock Enable
IDDR2	プリミティブ：Double Data Rate Input D Flip-Flop with Optional Data Alignment, Clock Enable and Programmable Synchronous or Asynchronous Set/Reset
LDCPE	プリミティブ：Transparent Data Latch with Asynchronous Clear and Preset and Gate Enable
ODDR2	プリミティブ：Dual Data Rate Output D Flip-Flop with Optional Data Alignment, Clock Enable and Programmable Synchronous or Asynchronous Set/Reset

## シフト レジスタ

デザイン エLEMENT	説明
SRL16	プリミティブ：16-Bit Shift Register Look-Up Table (LUT)
SRL16_1	プリミティブ：16-Bit Shift Register Look-Up Table (LUT) with Negative-Edge Clock
SRL16E	プリミティブ：16-Bit Shift Register Look-Up Table (LUT) with Clock Enable
SRL16E_1	プリミティブ：16-Bit Shift Register Look-Up Table (LUT) with Negative-Edge Clock and Clock Enable
SRLC16	プリミティブ：16-Bit Shift Register Look-Up Table (LUT) with Carry
SRLC16_1	プリミティブ：16-Bit Shift Register Look-Up Table (LUT) with Carry and Negative-Edge Clock
SRLC16E	プリミティブ：16-Bit Shift Register Look-Up Table (LUT) with Carry and Clock Enable
SRLC16E_1	プリミティブ：16-Bit Shift Register Look-Up Table (LUT) with Carry, Negative-Edge Clock, and Clock Enable

## スライス/CLB プリミティブ

デザイン エLEMENT	説明
BUFCF	プリミティブ：Fast Connect Buffer

デザイン エLEMENT	説明
LUT1	プリミティブ：1-Bit Look-Up Table with General Output
LUT1_D	プリミティブ：1-Bit Look-Up Table with Dual Output
LUT1_L	プリミティブ：1-Bit Look-Up Table with Local Output
LUT2	プリミティブ：2-Bit Look-Up Table with General Output
LUT2_D	プリミティブ：2-Bit Look-Up Table with Dual Output
LUT2_L	プリミティブ：2-Bit Look-Up Table with Local Output
LUT3	プリミティブ：3-Bit Look-Up Table with General Output
LUT3_D	プリミティブ：3-Bit Look-Up Table with Dual Output
LUT3_L	プリミティブ：3-Bit Look-Up Table with Local Output
LUT4	プリミティブ：4-Bit Look-Up-Table with General Output
LUT4_D	プリミティブ：4-Bit Look-Up Table with Dual Output
LUT4_L	プリミティブ：4-Bit Look-Up Table with Local Output
MULT_AND	プリミティブ：Fast Multiplier AND
MUXCY	プリミティブ：2-to-1 Multiplexer for Carry Logic with General Output
MUXCY_D	プリミティブ：2-to-1 Multiplexer for Carry Logic with Dual Output
MUXCY_L	プリミティブ：2-to-1 Multiplexer for Carry Logic with Local Output
MUXF5	プリミティブ：2-to-1 Look-Up Table Multiplexer with General Output
MUXF5_D	プリミティブ：2-to-1 Look-Up Table Multiplexer with Dual Output
MUXF5_L	プリミティブ：2-to-1 Look-Up Table Multiplexer with Local Output
MUXF6	プリミティブ：2-to-1 Look-Up Table Multiplexer with General Output
MUXF6_D	プリミティブ：2-to-1 Look-Up Table Multiplexer with Dual Output
MUXF6_L	プリミティブ：2-to-1 Look-Up Table Multiplexer with Local Output
MUXF7	プリミティブ：2-to-1 Look-Up Table Multiplexer with General Output
MUXF7_D	プリミティブ：2-to-1 Look-Up Table Multiplexer with Dual Output
MUXF7_L	プリミティブ：2-to-1 look-up table Multiplexer with Local Output
MUXF8	プリミティブ：2-to-1 Look-Up Table Multiplexer with General Output
MUXF8_D	プリミティブ：2-to-1 Look-Up Table Multiplexer with Dual Output

デザイン エLEMENT	説明
<a href="#">MUXF8_L</a>	プリミティブ : 2-to-1 Look-Up Table Multiplexer with Local Output
<a href="#">XORCY</a>	プリミティブ : XOR for Carry Logic with General Output
<a href="#">XORCY_D</a>	プリミティブ : XOR for Carry Logic with Dual Output
<a href="#">XORCY_L</a>	プリミティブ : XOR for Carry Logic with Local Output





## デザイン エLEMENT

---

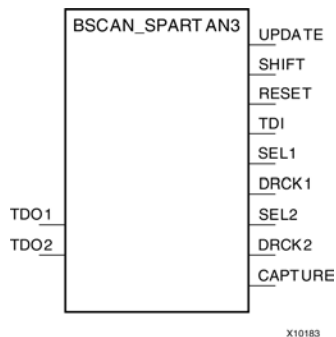
このセクションでは、このアーキテクチャで利用できるデザイン エLEMENTについて説明します。デザイン エLEMENTは、アルファベット順に並べられています。

各ライブラリ エLEMENTについて、次の情報を示します。

- ・ 名称
- ・ 説明
- ・ 回路図シンボル (該当するELEMENTでのみ)
- ・ 論理表 (該当するELEMENTでのみ)
- ・ ポートの説明
- ・ デザインの入力方法
- ・ 使用可能な属性 (該当するELEMENTでのみ)
- ・ インスタンス化コードの例
- ・ その他のリソース

## BSCAN\_SPARTAN3

: Spartan®-3 and Spartan-3E JTAG Boundary Scan Logic Access Circuit



### 概要

このデザイン エLEMENTを使用すると、JTAG バウンダリ スキャン ロジック コントローラを介して内部ロジックへアクセスできるので、内部実行デザインと FPGA の専用 JTAG ピン間の通信を可能にします。

**メモ：** 各アーキテクチャのバウンダリ スキャンの詳細については、データシートを参照してください。

### ポートの説明

ポート名	方向	幅	機能
TDI	出力	1	FPGA の TDI 入力ピンと同じ値を出力します。
DRCK1、DRK2	出力	1	JTAG USER 命令が読み込まれるとアクティブになり、JTAG TAP コントローラが SHIFT-DR ステートになると TCK ピンと同じ値を出力します。DRK1 は USER1 ロジックに適用され、DRK2 は USER2 に適用されます。
RESET	出力	1	USER 命令が読み込まれるとアクティブになり、JTAG TAP コントローラが TEST-LOGIC-RESET ステートになると High にアサートされます。
SEL1、SEL2	出力	1	JTAG 命令レジスタに USER1 または USER2 命令が読み込まれたことを示します。UPDATE-IR ステートになるとアクティブになり、新しい命令が読み込まれるまでアクティブのままになります。
SHIFT	出力	1	USER 命令が読み込まれるとアクティブになり、JTAG TAP コントローラが SHIFT-DR ステートになると High にアサートされます。
CAPTURE	出力	1	USER 命令が読み込まれるとアクティブになり、JTAG TAP コントローラが CAPTURE-DR ステートになると High にアサートされます。
UPDATE	出力	1	USER 命令が読み込まれるとアクティブになり、JTAG TAP コントローラが UPDATE-DR ステートになると High にアサートされます。
TDO1、TDO2	入力	1	USER1 または USER2 命令が読み込まれるとアクティブになり、外部 JTAG TDO ピンにコンポーネントの TDO1 (USER1) または TDO2 (USER2) ピンへのデータ入力の値が反映されます。

## デザインの入力方法

インスタンス化	推奨
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BSCAN_SPARTAN3: Boundary Scan primitive for connecting internal logic to
--                  JTAG interface.
--                  Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BSCAN_SPARTAN3_inst : BSCAN_SPARTAN3
port map (
    CAPTURE => CAPTURE, -- CAPTURE output from TAP controller
    DRCK1  => DRCK1,    -- Data register output for USER1 functions
    DRCK2  => DRCK2,    -- Data register output for USER2 functions
    RESET  => RESET,    -- Reset output from TAP controller
    SEL1   => SEL1,     -- USER1 active output
    SEL2   => SEL2,     -- USER2 active output
    SHIFT  => SHIFT,    -- SHIFT output from TAP controller
    TDI    => TDI,      -- TDI output from TAP controller
    UPDATE => UPDATE,   -- UPDATE output from TAP controller
    TDO1   => TDO1,    -- Data input for USER1 function
    TDO2   => TDO2,    -- Data input for USER2 function
);

-- End of BSCAN_SPARTAN3_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BSCAN_SPARTAN3: Boundary Scan primitive for connecting internal logic to
--                  JTAG interface.
--                  Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BSCAN_SPARTAN3_inst : BSCAN_SPARTAN3
port map (
    CAPTURE => CAPTURE, -- CAPTURE output from TAP controller
    DRCK1  => DRCK1,    -- Data register output for USER1 functions
    DRCK2  => DRCK2,    -- Data register output for USER2 functions
    RESET  => RESET,    -- Reset output from TAP controller
    SEL1   => SEL1,     -- USER1 active output
    SEL2   => SEL2,     -- USER2 active output
    SHIFT  => SHIFT,    -- SHIFT output from TAP controller
    TDI    => TDI,      -- TDI output from TAP controller
    UPDATE => UPDATE,   -- UPDATE output from TAP controller
    TDO1   => TDO1,    -- Data input for USER1 function
    TDO2   => TDO2,    -- Data input for USER2 function
);

-- End of BSCAN_SPARTAN3_inst instantiation
```

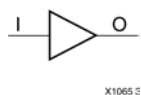
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## BUFCF

: Fast Connect Buffer

BUFCF



### 概要

このデザイン エLEMENTは、一部の専用ロジックと LUT の出力を別の LUT の入力に直接接続するために使用する、単一の高速結合バッファです。このバッファを使用すると、CLB パックも行われます。LUT は、4 つまで 1 つのグループとして接続できます。

### デザインの入力方法

インスタンス化	推奨
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BUFCF: Fast connect buffer used to connect the outputs of the LUTs
--       and some dedicated logic directly to the input of another LUT.
--       Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BUFCF_inst: BUFCF (
port map (
    O => O, -- Connect to the output of a LUT
    I => I  -- Connect to the input of a LUT
);

-- End of BUFCF_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BUFCE: Fast connect buffer used to connect the outputs of the LUTs
--       and some dedicated logic directly to the input of another LUT.
--       Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BUFCE_inst: BUFCE (
  port map (
    O => O, -- Connect to the output of a LUT
    I => I  -- Connect to the input of a LUT
  );

-- End of BUFCE_inst instantiation
```

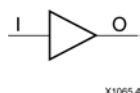
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## BUFG

: Global Clock Buffer

BUFG



### 概要

このデザイン エLEMENTは、ファンアウトが大きいバッファで、スキューを抑えて信号を分散するために、グローバル配線リソースへの信号に接続します。BUFG は、通常セット/リセットやクロック イネーブルなどのファンアウトの大きいネットやクロック ネットに使用されます。

### ポートの説明

ポート名	タイプ	幅	機能
I	入力	1	クロック バッファ出力
O	出力	1	クロック バッファ入力

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BUFG: Global Clock Buffer (source by an internal signal)
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BUFG_inst : BUFG
port map (
    O => O,      -- Clock buffer output
    I => I        -- Clock buffer input
);

-- End of BUFG_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;  
use UNISIM.vcomponents.all;  
  
-- BUFG: Global Clock Buffer (source by an internal signal)  
--      Spartan-3E  
-- Xilinx HDL Libraries Guide, version 12.2  
  
BUFG_inst : BUFG  
port map (  
    O => O,      -- Clock buffer output  
    I => I        -- Clock buffer input  
);  
  
-- End of BUFG_inst instantiation
```

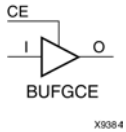
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## BUFGCE

: Global Clock Buffer with Clock Enable



### 概要

このデザイン エLEMENTは、クロック イネーブル付きグローバル クロック バッファです。O 出力は、クロック イネーブル (CE) が Low (非アクティブ) のときに 0 になります。CE が High になると、I 入力の値が O に出力されます。

### 論理表

入力		出力
I	CE	O
X	0	0
I	1	I

### デザインの入力方法

インスタンシエーション	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンシエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BUFGCE: Global Clock Buffer with Clock Enable (active high)
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BUFGCE_inst : BUFGCE
port map (
    O => O,    -- Clock buffer output
    CE => CE,  -- Clock enable input
    I => I     -- Clock buffer input
);

-- End of BUFGCE_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BUFGCE: Global Clock Buffer with Clock Enable (active high)
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BUFGCE_inst : BUFGCE
port map (
  O => O,    -- Clock buffer output
  CE => CE,   -- Clock enable input
  I => I      -- Clock buffer input
);

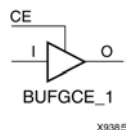
-- End of BUFGCE_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## BUFGCE\_1

: Global Clock Buffer with Clock Enable and Output State 1



### 概要

このデザイン エLEMENTは、クロック イネーブル付きグローバル クロック バッファです。O 出力は、クロック イネーブル (CE) が Low (非アクティブ) のときに High (1) になります。CE が High になると、I 入力の値が O に出力されます。

### 論理表

入力		出力
I	CE	O
X	0	1
I	1	I

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BUFGCE_1: Global Clock Buffer with Clock Enable (active low)
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BUFGCE_1_inst : BUFGCE_1
port map (
    O => O,    -- Clock buffer output
    CE => CE,  -- Clock enable input
    I => I     -- Clock buffer input
);

-- End of BUFGCE_1_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BUFGCE_1: Global Clock Buffer with Clock Enable (active low)
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BUFGCE_1_inst : BUFGCE_1
port map (
  O => O,    -- Clock buffer output
  CE => CE,   -- Clock enable input
  I => I      -- Clock buffer input
);

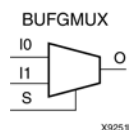
-- End of BUFGCE_1_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## BUFGMUX

: Global Clock MUX Buffer



### 概要

BUFGMUX はマルチプレクサの機能を持つグローバル クロック バッファで、2 つの入力クロック (I0 および I1) のいずれかを選択できます。セレクト入力 (S) が Low の場合、I0 の信号が出力 (O) に選択されます。S が High の場合は、I1 の信号が O に選択されます。

BUFGMUX および BUFGMUX\_1 では、S の値が変化した後クロックが切り替わるまでに保持される出力ステートが異なります。BUFGMUX は出力ステートが 0 に、BUFGMUX\_1 は出力ステートが 1 に保持されます。

**メモ:** BUFGMUX では、S がトグルされると、次のアクティブ クロック エッジ (I0 または I1) まで、出力のステートが非アクティブのまま保持されます。

### 論理表

入力			出力
I0	I1	S	O
I0	X	0	I0
X	I1	1	I1
X	X	↑	0
X	X	↓	0

### ポートの説明

ポート名	タイプ	幅	機能
I0	入力	1	クロック 0 入力
I1	入力	1	クロック 1 入力
O	出力	1	クロック MUX 出力
S	入力	1	クロック セレクト入力

### デザインの入力方法

インスタンス化	推奨
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BUFGMUX: Global Clock Buffer 2-to-1 MUX
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BUFGMUX_inst : BUFGMUX
port map (
    O => O,      -- Clock MUX output
    I0 => I0,     -- Clock0 input
    I1 => I1,     -- Clock1 input
    S => S       -- Clock select input
);

-- End of BUFGMUX_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BUFGMUX: Global Clock Buffer 2-to-1 MUX
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BUFGMUX_inst : BUFGMUX
port map (
    O => O,      -- Clock MUX output
    I0 => I0,     -- Clock0 input
    I1 => I1,     -- Clock1 input
    S => S       -- Clock select input
);

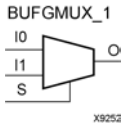
-- End of BUFGMUX_inst instantiation
```

## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)

## BUFGMUX\_1

: Global Clock MUX Buffer with Output State 1



### 概要

このデザイン エLEMENTは、マルチプレクサの機能を持つグローバル クロック バッファで、2 つの入力クロック (I0 および I1) のいずれかを選択できます。セレクト入力 (S) が Low の場合、I0 の信号が出力 (O) に選択されます。S が High の場合は、I1 の信号が O に選択されます。

このデザイン エLEMENTと BUFGMUX では、S の値が変化した後クロックが切り替わるまでに保持される出力ステートが異なります。BUFGMUX は出力ステートが 0 に、BUFGMUX\_1 は出力ステートが 1 に保持されます。

### 論理表

入力			出力
I0	I1	S	O
I0	X	0	I0
X	I1	1	I1
X	X	↑	1
X	X	↓	1

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BUFGMUX_1: Global Clock Buffer 2-to-1 MUX (inverted select)
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BUFGMUX_1_inst : BUFGMUX_1
port map (
    O => O,      -- Clock MUX output
    I0 => I0,    -- Clock0 input
    I1 => I1,    -- Clock1 input
    S => S       -- Clock select input
);

-- End of BUFGMUX_1_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- BUFGMUX_1: Global Clock Buffer 2-to-1 MUX (inverted select)
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

BUFGMUX_1_inst : BUFGMUX_1
port map (
  O => O,    -- Clock MUX output
  I0 => I0,   -- Clock0 input
  I1 => I1,   -- Clock1 input
  S => S      -- Clock select input
);

-- End of BUFGMUX_1_inst instantiation
```

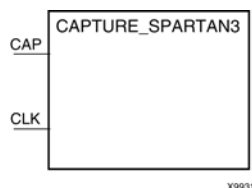
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## CAPTURE\_SPARTAN3

: Spartan®-3 Register State Capture for Bitstream Readback



### 概要

このデザイン エLEMENTは、レジスタ（フリップフロップとラッチ）情報のキャプチャ方法およびそのタイミングを制御します。リードバック機能は、専用のコンフィギュレーション ポート命令により提供されます。このELEMENTを使用しない場合は、データはコンフィギュレーション クロックに同期してリードバックされます。レジスタ（フリップフロップとラッチ）の値のみをキャプチャできます。LUT RAM、SRL、ブロック RAM の値もリードバックされますが、キャプチャできません。

CAP 信号を High にアサートすると、次にクロックが Low から High に切り替わる時にデバイス内のレジスタがキャプチャされます。デフォルトでは、トリガ（CAP をアサートしているときの CLK の遷移）のたびにデータがキャプチャされます。リードバック処理を 1 回のデータ キャプチャだけに制限するには、このELEMENTに ONESHOT=TRUE 属性を追加します。

### ポートの説明

ポート名	方向	幅	機能
CAP	入力	1	リードバック キャプチャトリガ
CLK	入力	1	リードバック キャプチャ クロック

### デザインの入力方法

インスタンシエーション	推奨
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

正しく動作するように、入力および出力をすべてデザインに接続します。

### 使用可能な属性

属性	タイプ	値	デフォルト	説明
ONESHOT	ブール代数	TRUE、FALSE	TRUE	CAP トリガごとに 1 回のリードバックを実行します。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- CAPTURE_SPARTAN3: Register State Capture for Bitstream Readback
--                      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

CAPTURE_SPARTAN3_inst : CAPTURE_SPARTAN3
port map (
    CAP => CAP,    -- Capture input
    CLK => CLK     -- Clock input
);

-- End of CAPTURE_SPARTAN3_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- CAPTURE_SPARTAN3: Register State Capture for Bitstream Readback
--                      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

CAPTURE_SPARTAN3_inst : CAPTURE_SPARTAN3
port map (
    CAP => CAP,    -- Capture input
    CLK => CLK     -- Clock input
);

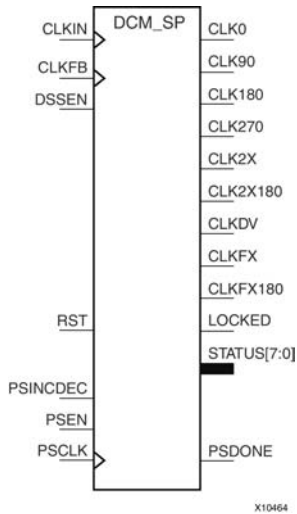
-- End of CAPTURE_SPARTAN3_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## DCM\_SP

: Digital Clock Manager



## 概要

このデザイン エLEMENTは、さまざまな機能を備えたデジタル クロック マネージャで、クロック遅延ロック ループ (DLL)、デジタル周波数合成 (DFS)、デジタル位相シフト (DPS) といった機能をインプリメントできます。DCM\_SP は、オンチップおよびオフチップからのクロック遅延をなくしたり、データ キャプチャを向上するためにクロック位相をシフトしたり、異なる周波数のクロックを生成させる場合などに便利です。

## ポートの説明

ポート名	タイプ	幅	機能
CLKDV	出力	1	CLKDV_DIVIDE 属性で制御される分周クロック出力。CLKDV_DIVIDE 属性で整数以外の値が設定されていない限り、CLKDV 出力のデューティサイクルは 50% になります。
CLKFB	入力	1	DCM へのクロック フィードバック入力。DFS 出力、CLKFX、または CLKFX180 がスタンダオンで使用されない限り、フィードバック入力が必要です。CLKFB 入力は、DCM の CLK0 または CLK2X 出力から供給し、それに応じて CLK_FEEDBACK を 1X または 2X に設定する必要があります。NONE に設定すると、CLKFB は使用されず Low に保持する必要があります。フィードバック ポイントには、内部または外部でクロック分配ネットワークに追加される遅延が含まれているのが理想的です。
CLKFX	出力	1	CLKFX_MULTIPLY および CLKFX_DIVIDE 属性で制御される同期クロック出力。デューティサイクルは常に 50% で、位相関係が不要な場合は、クロック フィードバックも不要です。
CLKFX180	出力	1	180 度位相がシフトされている同期クロック出力 CLKFX (CLKFX の反転バージョン)。デューティサイクルは常に 50% で、位相関係が不要な場合は、フィードバック ループも不要です。
CLKIN	入力	1	DCM へのクロック入力。常に必要で、CLKIN 周波数およびジッタがデータシートに記載されている範囲内である必要があります。
CLK0	出力	1	位相シフトが 0 の (位相がシフトされていない) CLKIN と同じ周波数。Spartan®-6 FPGA ではデューティサイクルが 50% になるように調整されます。CLK_FEEDBACK は CLK0 のスキューを調整するように 1X または 2X に設定する必要があります。

ポート名	タイプ	幅	機能
CLK2X	出力	1	位相シフトが 0 の (位相がシフトされていない) 2 倍の周波数クロック出力。CLK2X 出力のデューティ サイクルは常に 50% です。DLL 機能のフィードバックソースに CLK0 または CLK2X のいずれかが必要です。
CLK2X180	出力	1	位相シフトが 180 の (位相がシフトされていない) 2 倍の周波数クロック出力。CLK2X180 出力のデューティ サイクルは常に 50% です。
CLK90	出力	1	90 度 (1/4 周期) 位相がシフトされている CLKIN と同じ周波数。Spartan®-6 FPGA ではデューティ サイクルが 50% になるように調整されます。
CLK180	出力	1	180 度 (1/2 周期) 位相がシフトされている CLKIN と同じ周波数。Spartan®-6 FPGA ではデューティ サイクルが 50% になるように調整されます。
CLK270	出力	1	270 度 (3/4 周期) 位相がシフトされている CLKIN と同じ周波数。Spartan®-6 FPGA ではデューティ サイクルが 50% になるように調整されます。
LOCKED	出力	1	すべての DCM 機能が CLKIN 周波数にロックされています。クロック出力は有効で、CLKIN が特定の範囲内にあることが想定されます。 <ul style="list-style-type: none"> <li>0 : DCM で CLKIN 周波数にロックが試みられます。DCM クロック出力は無効です。</li> <li>1 : DCM が CLKIN 周波数にロックされています。DCM クロック出力は有効です。</li> <li>1 が 0 になるとき : DCM で LOCK が失われ、DCM がリセットされます。</li> </ul>
PSCLK	入力	1	立ち上がりエッジでクロックが供給される可変位相シフタへのクロック入力。グローバル クロック バッファを使用している場合、PSCLK を駆動できるのは上部の 8 つの BUFGMUX のみです (BUFGMUX_X2Y1、BUFGMUX_X2Y2、BUFGMUX_X2Y3、BUFGMUX_X2Y4、BUFGMUX_X3Y5、BUFGMUX_X3Y6、BUFGMUX_X3Y7、および BUFGMUX_X3Y8)。
PSDONE	出力	1	可変位相シフトの完了 <ul style="list-style-type: none"> <li>0 : 位相シフト操作が実行されていないか、位相シフト操作を実行中です。</li> <li>1 : 要求された位相シフト操作が完了しています。1 RSCLK サイクル間 High になります。次の可変位相シフト操作を開始できます。</li> </ul>
PSEN	入力	1	可変位相シフト イネーブル。DCM ブロック内で反転可能です。次に、反転しない場合の動作を示します。 <ul style="list-style-type: none"> <li>0 : 可変位相シフトをディスエーブルにします。位相シフタへの入力が無視されます。</li> <li>1 : 次の PSCLK クロックの立ち上がりエッジで可変位相シフト操作をイネーブルにします。</li> </ul>
PSINCDEC	入力	1	可変位相シフトを増分または減分します。DCM ブロック内で反転可能です。次に、反転しない場合の動作を示します。 <ul style="list-style-type: none"> <li>0 : 次にイネーブルにされている PSCLK クロックの立ち上がりエッジで位相シフト値を減分します。</li> <li>1 : 次にイネーブルにされている PSCLK クロックの立ち上がりエッジで位相シフト値を増分します。</li> </ul>

ポート名	タイプ	幅	機能
RST	入力	1	<p>非同期リセット入力。DCM ロジックをコンフィギュレーション後の状態にリセットします。これにより、DCM は CLKIN 入力に再度ロックされます。DCM ブロック内で反転可能です。次に、反転しない場合の動作を示します。</p> <ul style="list-style-type: none"> <li>0：影響なし</li> <li>1：DCM ブロックのリセット。最低 CLKIN 3 サイクル間 RST を High に保持します。</li> </ul>
STATUS[7:0]	出力	8	<p>ステータス出力バスにより DCM のステータスが示されます。</p> <ul style="list-style-type: none"> <li>STATUS[0]：可変位相シフト オーバーフロー。可変ファイン位相シフトの制御出力。可変位相シフトが許容される最低値または最大値に達しています。位相シフトが遅延ラインの最後に到達している場合は、許容範囲は +/-255 以下です。 <ul style="list-style-type: none"> <li>0：位相シフトが制限値に達成していません。</li> <li>1：位相シフトが制限値に達成しました。</li> </ul> </li> <li>STATUS[1]：CLKIN 入力の停止。CLKFB フィードバック入力が接続されているときのみ使用できます。LOCKED 出力がアサートされるまでリセットに保持されます。アクティブになるには最低 CLKIN が 1 サイクル必要です。CLKIN がトグルしない場合はアサートされません。 <ul style="list-style-type: none"> <li>0：CLKIN 入力が入力されています。</li> <li>1：CLKIN 入力は、LOCKED 出力が High になることが可能なときでもトグルしていません。</li> </ul> </li> <li>STATUS[2]：CLKFX または CLKFX180 出力の停止 <ul style="list-style-type: none"> <li>0：CLKFX および CLKFX180 出力が入力されています。</li> <li>1：CLKFX および CLKFX180 出力は、LOCKED 出力が High になることが可能なときでもトグルしていません。</li> </ul> </li> <li>STATUS[7:3]：予約</li> </ul>

## デザインの入力方法

インスタンシエーション	可
推論	不可
CORE Generator™ およびウィザード	推奨
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
CLK_FEEDBACK	文字列	1X、2X、NONE	1X	<p>DCM フィードバック モードを定義します。</p> <ul style="list-style-type: none"> <li>1X：CLK0 をフィードバックにします。</li> <li>2X：CLK2X をフィードバックにします。</li> </ul>

属性	タイプ	値	デフォルト	説明
CLKDV_DIVIDE	1 上位ビット FLOAT	2.0、1.5、2.5、3.0、 3.5、4.0、4.5、5.0、 5.5、6.0、6.5、7.0、 7.5、8.0、9.0、10.0、 11.0、12.0、13.0、 14.0、15.0、16.0	2.0	CLKDLL、CLKDLLE、CLKDLLHF、 DCM_SP のクロック分周器 (CLKDV 出力) の分周比を指定
CLKFX_DIVIDE	整数	1 ~ 32	1	CLKFX 出力の分周比を指定
CLKFX_MULTIPLY	整数	2 ~ 32	4	CLKFX 出力の通倍比を指定
CLKIN_DIVIDE_BY_2	ブール代数	FALSE、TRUE	FALSE	CLKIN を 2 で分周
CLKIN_PERIOD	文字列	0 ビット文字列	10.0	CLKIN 入力への入力周期を指定 (ns)
CLKOUT_PHASE_SHIFT	文字列	NONE、FIXED、 VARIABLE	NONE	位相シフト モードを指定 <ul style="list-style-type: none"> <li>・ NONE : 位相シフト機能なし。設定されている値は反映されません。</li> <li>・ FIXED : DCM の出力は CLKIN から決まった位相だけシフトされたものになります。値は PHASE_SHIFT 属性で指定されます。</li> <li>・ VARIABLE : DCM 出力が CLKIN に相対して正および負の範囲にシフトできます。開始値は PHASE_SHIFT 属性で指定されます。</li> </ul>
DESKEW_ADJUST	文字列	SYSTEM_ SYNCHRONOUS、 SOURCE_ SYNCHRONOUS	SYSTEM_ SYNCHRONOUS	DCM_SP クロック出力と FPGA のクロック入力ピン間のクロック遅延の配分に影響する コンフィギュレーション ビットを設定
DFS_FREQUENCY_MODE	文字列	LOW、HIGH	LOW	この属性はレガシ属性です。DCM は常に 自動周波数検索モードになります。High または Low に設定しても、無視されます。
DLL_FREQUENCY_MODE	文字列	LOW、HIGH	LOW	この属性はレガシ属性です。DCM は常に 自動周波数検索モードになります。High または Low に設定しても、無視されます。
DSS_MODE	文字列	NONE、 SPREAD_2、 SPREAD_4、 SPREAD_6、 SPREAD_8	NONE	<b>メモ</b> : Spartan®-6 では使用されません。 出力クロックの周波数拡散を指定します。 <ul style="list-style-type: none"> <li>・ NONE : デフォルト。拡散係数が指定されていません。デジタル スペクトラム拡散はディスエーブルにされています。</li> <li>・ SPREAD_2 : 現時点のクロック周期 +/- 50ps のクロック周期を作成します。</li> <li>・ SPREAD_4 : 現時点のクロック周期 +/- 100ps のクロック周期を作成します。</li> <li>・ SPREAD_6 : 現時点のクロック周期 +/- 150ps のクロック周期を作成します。</li> </ul>

属性	タイプ	値	デフォルト	説明
				<ul style="list-style-type: none"> <li>SPREAD_8：現時点のクロック周期 <math>\pm 200\text{ps}</math> のクロック周期を作成します。</li> </ul> <p>SPREAD_# の数値が増えると、拡散は累積的に追加されます。たとえば、SPREAD_2 と指定すると、入力クロック周波数 <math>\pm 50\text{ps}</math> の 2 つのクロック周波数が追加されます。SPREAD_4 と指定すると、さらに <math>\pm 100\text{ps}</math> の 2 つのクロック周波数が追加されます。</p>
DUTY_CYCLE_CORRECTION	ブール代数	TRUE、FALSE	TRUE	サポートなし
FACTORY_JF	16 進数	16'h8080 ~ 16'hffff	16'hc080	サポートなし
PHASE_SHIFT	整数	-255 ~ 255	0	<p>この属性は、CLKOUT_PHASE_SHIFT 属性が FIXED または VARIABLE に設定されている場合のみ使用できます。コンフィギュレーションでの CLKIN とすべての DCM クロック出力間の立ち上がりエッジ スキューを定義し、DCM クロック出力の位相をシフトします。スキューまたは位相シフト値は、ファイン位相シフトの式で表現されているように、クロック周期の係数を表す整数で指定します。実際に許容される値は、入力クロックの周波数によって異なります。TCLKIN が FINE_SHIFT_RANGE より大きいとき実際の範囲は狭くなります。FINE_SHIFT_RANGE は、遅延ラインのすべてのタップの総遅延を示します。</p>
STARTUP_WAIT	ブール代数	FALSE、TRUE	FALSE	<p>FPGA コンフィギュレーション DONE 信号を High にするのを、DCM の LOCKED 信号がアサートされるまで待つかどうかを指定します。</p> <ul style="list-style-type: none"> <li>FALSE：デフォルト値。DCM の LOCKED 信号がアサートされるのを待たずにコンフィギュレーションの最後にアサートされます。</li> <li>TRUE：DONE 信号は関連する DCM の LOCKED 信号が High になるまで High になりません。</li> </ul> <p>STARTUP_WAIT の指定にかかわらず、LOCKED 信号は High になります。FPGA のスタートアップ シーケンスも変更し、延期サイクルの前に LCK (ロック) サイクルを挿入する必要があります。DONE サイクルまたは GWE サイクルが一般的です。複数の DCM をコンフィギュレーションする場合は、すべての DCM がロックされるまで DONE ピンは High になりません。</p>

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- DCM_SP: Digital Clock Manager Circuit
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

DCM_SP_inst : DCM_SP
generic map (
  CLKDV_DIVIDE => 2.0, -- Divide by: 1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0,5.5,6.0,6.5
                        --      7.0,7.5,8.0,9.0,10.0,11.0,12.0,13.0,14.0,15.0 or 16.0
  CLKFX_DIVIDE => 1,   -- Can be any integer from 1 to 32
  CLKFX_MULTIPLY => 4, -- Can be any integer from 1 to 32
  CLKIN_DIVIDE_BY_2 => FALSE, -- TRUE/FALSE to enable CLKIN divide by two feature
  CLKIN_PERIOD => 0.0, -- Specify period of input clock
  CLKOUT_PHASE_SHIFT => "NONE", -- Specify phase shift of "NONE", "FIXED" or "VARIABLE"
  CLK_FEEDBACK => "1X", -- Specify clock feedback of "NONE", "1X" or "2X"
  DESKEW_ADJUST => "SYSTEM_SYNCHRONOUS", -- "SOURCE_SYNCHRONOUS", "SYSTEM_SYNCHRONOUS" or
                        --      an integer from 0 to 15
  DLL_FREQUENCY_MODE => "LOW", -- "HIGH" or "LOW" frequency mode for DLL
  DUTY_CYCLE_CORRECTION => TRUE, -- Duty cycle correction, TRUE or FALSE
  PHASE_SHIFT => 0, -- Amount of fixed phase shift from -255 to 255
  STARTUP_WAIT => FALSE) -- Delay configuration DONE until DCM_SP LOCK, TRUE/FALSE
port map (
  CLK0 => CLK0, -- 0 degree DCM CLK ouptput
  CLK180 => CLK180, -- 180 degree DCM CLK output
  CLK270 => CLK270, -- 270 degree DCM CLK output
  CLK2X => CLK2X, -- 2X DCM CLK output
  CLK2X180 => CLK2X180, -- 2X, 180 degree DCM CLK out
  CLK90 => CLK90, -- 90 degree DCM CLK output
  CLKDV => CLKDV, -- Divided DCM CLK out (CLKDV_DIVIDE)
  CLKFX => CLKFX, -- DCM CLK synthesis out (M/D)
  CLKFX180 => CLKFX180, -- 180 degree CLK synthesis out
  LOCKED => LOCKED, -- DCM LOCK status output
  PSDONE => PSDONE, -- Dynamic phase adjust done output
  STATUS => STATUS, -- 8-bit DCM status bits output
  CLKFB => CLKFB, -- DCM clock feedback
  CLKIN => CLKIN, -- Clock input (from IBUFG, BUFG or DCM)
  PSCLK => PSCLK, -- Dynamic phase adjust clock input
  PSEN => PSEN, -- Dynamic phase adjust enable input
  PSINCDEC => PSINCDEC, -- Dynamic phase adjust increment/decrement
  RST => RST -- DCM asynchronous reset input
);

-- End of DCM_SP_inst instantiation
```



## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- DCM_SP: Digital Clock Manager Circuit
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

DCM_SP_inst : DCM_SP
generic map (
    CLKDV_DIVIDE => 2.0, -- Divide by: 1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0,5.5,6.0,6.5
                        --      7.0,7.5,8.0,9.0,10.0,11.0,12.0,13.0,14.0,15.0 or 16.0
    CLKFX_DIVIDE => 1,  -- Can be any integer from 1 to 32
    CLKFX_MULTIPLY => 4, -- Can be any integer from 1 to 32
    CLKIN_DIVIDE_BY_2 => FALSE, -- TRUE/FALSE to enable CLKIN divide by two feature
    CLKIN_PERIOD => 0.0, -- Specify period of input clock
    CLKOUT_PHASE_SHIFT => "NONE", -- Specify phase shift of "NONE", "FIXED" or "VARIABLE"
    CLK_FEEDBACK => "1X", -- Specify clock feedback of "NONE", "1X" or "2X"
    DESKEW_ADJUST => "SYSTEM_SYNCHRONOUS", -- "SOURCE_SYNCHRONOUS", "SYSTEM_SYNCHRONOUS" or
                        -- an integer from 0 to 15
    DLL_FREQUENCY_MODE => "LOW", -- "HIGH" or "LOW" frequency mode for DLL
    DUTY_CYCLE_CORRECTION => TRUE, -- Duty cycle correction, TRUE or FALSE
    PHASE_SHIFT => 0, -- Amount of fixed phase shift from -255 to 255
    STARTUP_WAIT => FALSE) -- Delay configuration DONE until DCM_SP LOCK, TRUE/FALSE
port map (
    CLK0 => CLK0, -- 0 degree DCM CLK output
    CLK180 => CLK180, -- 180 degree DCM CLK output
    CLK270 => CLK270, -- 270 degree DCM CLK output
    CLK2X => CLK2X, -- 2X DCM CLK output
    CLK2X180 => CLK2X180, -- 2X, 180 degree DCM CLK out
    CLK90 => CLK90, -- 90 degree DCM CLK output
    CLKDV => CLKDV, -- Divided DCM CLK out (CLKDV_DIVIDE)
    CLKFX => CLKFX, -- DCM CLK synthesis out (M/D)
    CLKFX180 => CLKFX180, -- 180 degree CLK synthesis out
    LOCKED => LOCKED, -- DCM LOCK status output
    PSDONE => PSDONE, -- Dynamic phase adjust done output
    STATUS => STATUS, -- 8-bit DCM status bits output
    CLKFB => CLKFB, -- DCM clock feedback
    CLKIN => CLKIN, -- Clock input (from IBUFG, BUFG or DCM)
    PSCLK => PSCLK, -- Dynamic phase adjust clock input
    PSEN => PSEN, -- Dynamic phase adjust enable input
    PSINCDEC => PSINCDEC, -- Dynamic phase adjust increment/decrement
    RST => RST -- DCM asynchronous reset input
);

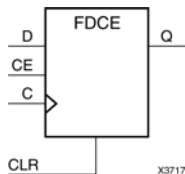
-- End of DCM_SP_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## FDCE

: D Flip-Flop with Clock Enable and Asynchronous Clear



### 概要

このデザイン エレメントは、クロック イネーブルと非同期クリアがある単一の D タイプ フリップフロップです。クロック イネーブル (CE) が High、非同期クリア (CLR) が Low の場合、クロック (C) が Low から High に切り替わるときにデータ入力 (D) の値がデータ出力 (Q) に送られます。CLR が High になると、ほかのすべての入力は無視され、出力 (Q) の値が Low にリセットされます。CE が Low の場合、クロック遷移は無視されます。

電力を供給すると、このフリップフロップは非同期にクリアされ、出力が Low になります。FPGA では、グローバル セット / リセット (GSR) をアクティブにすると、電源投入時の状態をシミュレーションできます。GSR のデフォルトはアクティブ High ですが、STARTUP\_architecture シンボルの GSR 入力の前にインバータを追加するとアクティブ Low にできます。

### 論理表

入力				出力
CLR	CE	D	C	Q
1	X	X	X	0
0	0	X	X	変化なし
0	1	D	↑	D

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	2 進数	0、1	0	コンフィギュレーション後の Q 出力の初期値を指定。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- FDCE: Single Data Rate D Flip-Flop with Asynchronous Clear and
--       Clock Enable (posedge clk).
--       Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

FDCE_inst : FDCE
generic map (
  INIT => '0') -- Initial value of register ('0' or '1')
port map (
  Q => Q,      -- Data output
  C => C,      -- Clock input
  CE => CE,    -- Clock enable input
  CLR => CLR,  -- Asynchronous clear input
  D => D       -- Data input
);

-- End of FDCE_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- FDCE: Single Data Rate D Flip-Flop with Asynchronous Clear and
--       Clock Enable (posedge clk).
--       Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

FDCE_inst : FDCE
generic map (
  INIT => '0') -- Initial value of register ('0' or '1')
port map (
  Q => Q,      -- Data output
  C => C,      -- Clock input
  CE => CE,    -- Clock enable input
  CLR => CLR,  -- Asynchronous clear input
  D => D       -- Data input
);

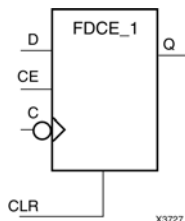
-- End of FDCE_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## FDCE\_1

: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Clear



## 概要

このデザイン エLEMENTは、データ (D)、クロック イネーブル (CE)、非同期クリア (CLR) の各入力とデータ出力 (Q) のある単一の D タイプ フリップフロップです。非同期 CLR が High になると、ほかのすべての入力は無視され、Q 出力が Low になります。CLR が Low、CE が High の場合、クロック (C) が High から Low に切り替わるときに D 入力の値がフリップフロップにロードされます。CE が Low の場合、クロック遷移は無視されます。

電力を供給すると、このフリップフロップは非同期にクリアされ、出力が Low になります。FPGA では、グローバル セット / リセット (GSR) をアクティブにすると、電源投入時の状態をシミュレーションできます。GSR のデフォルトはアクティブ High ですが、STARTUP\_architecture シンボルの GSR 入力の前にインバータを追加するとアクティブ Low にできます。

## 論理表

入力				出力
CLR	CE	D	C	Q
1	X	X	X	0
0	0	X	X	変化なし
0	1	D	↓	D

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

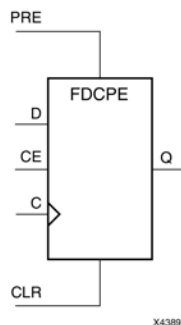
属性	タイプ	値	デフォルト	説明
INIT	2 進数	0、1	0	コンフィギュレーション後の Q 出力の初期値を指定。

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## FDCPE

: D Flip-Flop with Clock Enable and Asynchronous Preset and Clear



### 概要

このデザイン エLEMENTは、データ (D)、クロック イネーブル (CE)、非同期プリセット (PRE)、非同期クリア (CLR) の各入力とデータ出力 (Q) がある単一の D タイプ フリップフロップです。非同期 PRE が High になると Q 出力が High にセットされ、CLR が High になると出力が Low にリセットされます (CLR 入力 が PRE 入力よりも優先される)。PRE と CLR が Low で CE が High の場合、クロック (C) が Low から High に切り替わるときに D 入力の値がフリップフロップにロードされます。CE が Low の場合クロック遷移は無視され、以前の値が保持されます。FDCPE は通常、スライスまたは IOB レジスタとしてインプリメントされます。

FPGA では電源が投入されると、INIT 属性を使用して指定した初期値に設定されます。GSR (グローバル セット/リセット) をアサートすると、INIT で指定した初期値に非同期で設定されます。

**メモ**：非同期のセットおよびリセットの使用がサポートされていますが、これらの使用は通常お勧めしません。非同期信号を使用するとタイミングの問題が検出および制御しにくく、またロジックの最適化に悪影響を及ぼし、同期セットまたはリセットを使用した場合に比べて消費電力が大きくなる場合があります。

### 論理表

入力					出力
CLR	PRE	CE	D	C	Q
1	X	X	X	X	0
0	1	X	X	X	1
0	0	0	X	X	変化なし
0	0	1	D	↑	D

### ポートの説明

ポート名	方向	幅	機能
Q	出力	1	データ出力
C	入力	1	クロック入力
CE	入力	1	クロック イネーブル入力
CLR	入力	1	非同期クリア入力
D	入力	1	データ入力
PRE	入力	1	非同期セット入力

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	2 進数	0、1	0	コンフィギュレーション後および GSR 入力時の Q 出力の初期値を指定

## VHDL 記述（インスタンス化）

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- FDCPE: Single Data Rate D Flip-Flop with Asynchronous Clear, Set and
--       Clock Enable (posedge clk).
--       Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

FDCPE_inst : FDCPE
generic map (
  INIT => '0') -- Initial value of register ('0' or '1')
port map (
  Q => Q,        -- Data output
  C => C,        -- Clock input
  CE => CE,      -- Clock enable input
  CLR => CLR,    -- Asynchronous clear input
  D => D,        -- Data input
  PRE => PRE     -- Asynchronous set input
);

-- End of FDCPE_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- FDCPE: Single Data Rate D Flip-Flop with Asynchronous Clear, Set and
--       Clock Enable (posedge clk).
--       Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

FDCPE_inst : FDCPE
generic map (
  INIT => '0') -- Initial value of register ('0' or '1')
port map (
  Q => Q,      -- Data output
  C => C,      -- Clock input
  CE => CE,    -- Clock enable input
  CLR => CLR,  -- Asynchronous clear input
  D => D,      -- Data input
  PRE => PRE   -- Asynchronous set input
);

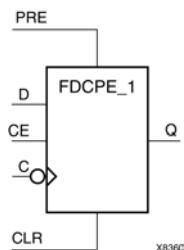
-- End of FDCPE_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## FDCPE\_1

: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Preset and Clear



## 概要

FDCPE\_1 は、データ (D)、クロック イネーブル (CE)、非同期プリセット (PRE)、非同期クリア (CLR) の各入力とデータ出力 (Q) がある単一の D フリップフロップです。非同期 PRE が High になると、Q 出力が High にセットされます。CLR が High になると、出力が Low にリセットされます。PRE と CLR が Low で CE が High の場合、クロック (C) が High から Low に切り替わるときに D 入力の値がフリップフロップにロードされます。CE が Low の場合、クロック遷移は無視されます。

電力を供給すると、このフリップフロップは非同期にクリアされ、出力が Low になります。FPGA では、グローバル セット / リセット (GSR) をアクティブにすると、電源投入時の状態をシミュレーションできます。GSR のデフォルトはアクティブ High ですが、STARTUP\_architecture シンボルの GSR 入力の前にインバータを追加するとアクティブ Low にできます。

## 論理表

入力					出力
CLR	PRE	CE	D	C	Q
1	X	X	X	X	0
0	1	X	X	X	1
0	0	0	X	X	変化なし
0	0	1	D	↓	D

## ポートの説明

ポート名	方向	幅	機能
Q	出力	1	データ出力
C	入力	1	クロック入力
CE	入力	1	クロック イネーブル入力
CLR	入力	1	非同期クリア入力
D	入力	1	データ入力
PRE	入力	1	非同期セット入力



## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

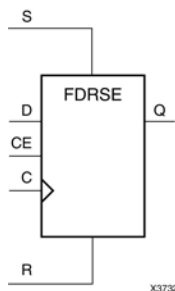
属性	タイプ	値	デフォルト	説明
INIT	2 進数	0、1	0	コンフィギュレーション後および GSR 入力時の Q 出力の初期値を指定

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## FDRSE

: D Flip-Flop with Synchronous Reset and Set and Clock Enable



### 概要

FDRSE は、同期リセット (R)、同期セット (S)、クロック イネーブル (CE) の各入力とデータ出力 (Q) がある単一の D タイプ フリップフロップです。同期リセット入力 (R) が High になると、ほかの入力は無視され、クロック (C) が Low から High に切り替わる時に出力 (Q) が Low にリセットされます (リセットがセットよりも優先される)。セット入力 (S) が High、R が Low の場合、クロック (C) が Low から High に切り替わる時にフリップフロップがセットされ、出力が High になります。R と S が Low、CE が High の場合、クロックが Low から High に切り替わる時に D 入力の値がフリップフロップにロードされます。

電源が投入されると、INIT 属性を使用して指定した初期値に設定されます。GSR (グローバル セット/リセット) をアサートすると、INIT で指定した初期値に非同期で設定されます。

### 論理表

入力					出力
R	S	CE	D	C	Q
1	X	X	X	↑	0
0	1	X	X	↑	1
0	0	0	X	X	変化なし
0	0	1	1	↑	1
0	0	1	0	↑	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	2 進数	0、1	0	コンフィギュレーション後および GSR 入力時の Q 出力の初期値を指定

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- FDRSE: Single Data Rate D Flip-Flop with Synchronous Clear, Set and
--       Clock Enable (posedge clk).
--       Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

FDRSE_inst : FDRSE
generic map (
  INIT => '0') -- Initial value of register ('0' or '1')
port map (
  Q => Q,      -- Data output
  C => C,      -- Clock input
  CE => CE,    -- Clock enable input
  D => D,      -- Data input
  R => R,      -- Synchronous reset input
  S => S       -- Synchronous set input
);

-- End of FDRSE_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- FDRSE: Single Data Rate D Flip-Flop with Synchronous Clear, Set and
--       Clock Enable (posedge clk).
--       Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

FDRSE_inst : FDRSE
generic map (
  INIT => '0') -- Initial value of register ('0' or '1')
port map (
  Q => Q,      -- Data output
  C => C,      -- Clock input
  CE => CE,    -- Clock enable input
  D => D,      -- Data input
  R => R,      -- Synchronous reset input
  S => S       -- Synchronous set input
);

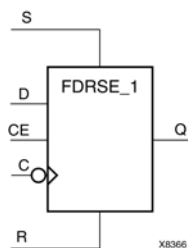
-- End of FDRSE_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## FDRSE\_1

: D Flip-Flop with Negative-Clock Edge, Synchronous Reset and Set, and Clock Enable



## 概要

FDRSE\_1 は、同期リセット (R)、同期セット (S)、クロック イネーブル (CE) の各入力とデータ出力 (Q) がある単一の D タイプ フリップフロップです。同期リセット入力 (R) が High になると、ほかの入力は無視され、クロック (C) が High から Low に切り替わるときに、出力 (Q) が Low にリセットされます (リセットがセットよりも優先される)。S が High、R が Low の場合、クロック (C) が High から Low に切り替わるときにフリップフロップがセットされ、出力が High になります。R と S が Low で CE が High の場合、クロックが High から Low に切り替わるときに D 入力の値がフリップフロップにロードされます。

電力を供給すると、このフリップフロップは非同期にクリアされ、出力が Low になります。FPGA では、グローバル セット / リセット (GSR) をアクティブにすると、電源投入時の状態をシミュレーションできます。GSR のデフォルトはアクティブ High ですが、STARTUP\_architecture シンボルの GSR 入力の前にインバータを追加するとアクティブ Low にできます。

## 論理表

入力					出力
R	S	CE	D	C	Q
1	X	X	X	↓	0
0	1	X	X	↓	1
0	0	0	X	X	変化なし
0	0	1	D	↓	D

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

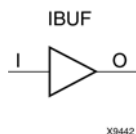
属性	タイプ	値	デフォルト	説明
INIT	2 進数	0、1	0	コンフィギュレーション後および GSR 入力時の Q 出力の初期値を指定

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリ データシート](#)

## IBUF

: Input Buffer



## 概要

このデザイン エレメントは、最上位の入力ポートまたは入出力ポートに接続されている信号に自動的に挿入されます。このバッファは通常、合成ツールで推論しますが、必要に応じてインスタンスエートすることも可能です。インスタンスエートするには、入力ポート (I) を関連する最上位の入力ポートまたは入出力ポートに接続し、出力ポート (O) をそのポートをソースとする FPGA ロジックに接続します。必要なジェネリック マップ (VHDL) またはパラメータ値代入 (Verilog) に変更を加えて、コンポーネントのデフォルトのビヘイビアを変更します。

## ポートの説明

ポート名	方向	幅	機能
O	出力	1	バッファの出力
I	入力	1	バッファの入力

## デザインの入力方法

インスタンスエーション	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

このエレメントは通常、デザインの最上位入力ポートに対して推論されます。通常はソース コードで指定する必要はありませんが、必要に応じてインスタンスエートできます。このコンポーネントをインスタンスエートするには、該当するライブラリ ガイドに含まれるインスタンスエーション コードを最上位エンティティ/モジュールに挿入します。デザイン階層を保つために、すべての I/O コンポーネントを必ずデザインの最上位に配置してください。I ポートをデザインの最上位入力ポートに、O ポートをこの入力供給されるロジックに直接接続します。generic/defparam 値を設定し、バッファのビヘイビアを適切に設定してください。

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
IBUF_DELAY_VALUE	文字列	0 ~ 12	0	IOB 外のレジスタを介さないパスに追加する遅延を指定
IFD_DELAY_VALUE	文字列	AUTO、0 ~ 6	AUTO	IOB 内のレジスタを介するパスに追加する遅延を指定
IOSTANDARD	文字列	データシートを参照	DEFAULT	エレメントに I/O 規格を割り当て

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IBUF: Single-ended Input Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IBUF_inst : IBUF
generic map (
    IBUF_DELAY_VALUE => "0", -- Specify the amount of added input delay for buffer,
                           -- "0"-"12" (Spartan-3E)
    IFD_DELAY_VALUE  => "AUTO", -- Specify the amount of added delay for input register,
                           -- "AUTO", "0"-"6"
    IOSTANDARD       => "DEFAULT")
port map (
    O => O,      -- Buffer output
    I => I       -- Buffer input (connect directly to top-level port)
);

-- End of IBUF_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IBUF: Single-ended Input Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IBUF_inst : IBUF
generic map (
    IBUF_DELAY_VALUE => "0", -- Specify the amount of added input delay for buffer,
                           -- "0"-"12" (Spartan-3E)
    IFD_DELAY_VALUE  => "AUTO", -- Specify the amount of added delay for input register,
                           -- "AUTO", "0"-"6"
    IOSTANDARD       => "DEFAULT")
port map (
    O => O,      -- Buffer output
    I => I       -- Buffer input (connect directly to top-level port)
);

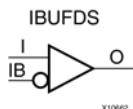
-- End of IBUF_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## IBUFDS

: Differential Signaling Input Buffer



## 概要

このデザイン エレメントは、低電圧差動信号を使用する入力バッファです。IBUFDS では、デザイン レベルのインターフェイス信号は、一方がマスタで、もう一方がスレーブとなる 2 つの異なるポート (I、IB) で表されます。マスタとスレーブは MYNET\_P と MYNET\_N のように、同じ論理信号の反対の状態を示します。オプションで、プログラム可能な差動終端機能を使用すると、シグナル インテグリティが向上し、外部コンポーネントの数を減らすことができます。

## 論理表

入力		出力
I	IB	O
0	0	変化なし
0	1	0
1	0	1
1	1	変化なし

## ポートの説明

ポート名	タイプ	幅	機能
I	入力	1	Diff_p バッファの入力
IB	入力	1	Diff_p バッファの入力
O	出力	1	バッファの出力

## デザインの入力方法

インスタンス化	推奨
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

デザイン階層を保つために、すべての I/O コンポーネントを必ずデザインの最上位に配置してください。I ポートを直接デザインの最上位のマスタとなる入力ポートに、IB ポートを最上位のスレーブとなる入力ポートに、O ポートをこの入力供給されるロジックに接続します。generic/defparam 値を設定し、バッファのビヘイビアを適切に設定してください。



## 使用可能な属性

属性	タイプ	値	デフォルト	説明
DIFF_TERM	ブール代数	TRUE、FALSE	FALSE	ビルトインの差動終端抵抗をイネーブル
IBUF_DELAY_VALUE	文字列	0 ～ 12	0	IOB 外のレジスタを介さないパスに追加する遅延を指定
IFD_DELAY_VALUE	文字列	AUTO、0 ～ 6	AUTO	IOB 内のレジスタを介するパスに追加する遅延を指定
IOSTANDARD	文字列	データシートを参照	DEFAULT	エレメントに I/O 規格を割り当て

## VHDL 記述（インスタンスエーション）

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IBUFDS: Differential Input Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IBUFDS_inst : IBUFDS
generic map (
    DIFF_TERM => FALSE, -- Differential Termination
    IBUF_DELAY_VALUE => "0", -- Specify the amount of added input delay for buffer,
                          -- "0"-"12"
    IFD_DELAY_VALUE => "AUTO", -- Specify the amount of added delay for input register,
                          -- "AUTO", "0"-"6"
    IOSTANDARD => "DEFAULT")
port map (
    O => O, -- Buffer output
    I => I, -- Diff_p buffer input (connect directly to top-level port)
    IB => IB -- Diff_n buffer input (connect directly to top-level port)
);

-- End of IBUFDS_inst instantiation
```

## Verilog 記述（インスタンス化）

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IBUFDS: Differential Input Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IBUFDS_inst : IBUFDS
generic map (
  DIFF_TERM => FALSE, -- Differential Termination
  IBUF_DELAY_VALUE => "0", -- Specify the amount of added input delay for buffer,
                        -- "0"-"12"
  IFD_DELAY_VALUE => "AUTO", -- Specify the amount of added delay for input register,
                        -- "AUTO", "0"-"6"
  IOSTANDARD => "DEFAULT")
port map (
  O => O, -- Buffer output
  I => I, -- Diff_p buffer input (connect directly to top-level port)
  IB => IB -- Diff_n buffer input (connect directly to top-level port)
);

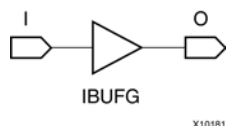
-- End of IBUFDS_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## IBUFG

: Dedicated Input Clock Buffer



### 概要

IBUFG は、FPGA への入力クロックをグローバル クロック配線リソースに接続するために使用する専用入力です。DCM\_SP および BUFG への専用接続となり、デバイスのクロック遅延とジッタが最小限に抑えられます。IBUFG の入力は、グローバル クロック ピンでのみ駆動できます。IBUFG の出力は、DCM\_SP、BUFG、または指定したロジックの CLKIN を駆動できます。

### ポートの説明

ポート名	方向	幅	機能
O	出力	1	クロック バッファ出力
I	入力	1	クロック バッファ入力

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### 使用可能な属性

属性	タイプ	値	デフォルト	説明
IBUF_DELAY_VALUE	文字列	0 ~ 12	0	IOB からのレジスタを介さないパスに追加する遅延を指定
IOSTANDARD	文字列	データシートを参照	DEFAULT	エレメントに I/O 規格を割り当て

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IBUFG: Single-ended global clock input buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IBUFG_inst : IBUFG
generic map (
    IBUF_DELAY_VALUE => "0", -- Specify the amount of added input delay for buffer,
                           -- "0"-"12"
    IOSTANDARD => "DEFAULT")
port map (
    O => O, -- Clock buffer output
    I => I -- Clock buffer input (connect directly to top-level port)
);

-- End of IBUFG_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IBUFG: Single-ended global clock input buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IBUFG_inst : IBUFG
generic map (
    IBUF_DELAY_VALUE => "0", -- Specify the amount of added input delay for buffer,
                           -- "0"-"12"
    IOSTANDARD => "DEFAULT")
port map (
    O => O, -- Clock buffer output
    I => I -- Clock buffer input (connect directly to top-level port)
);

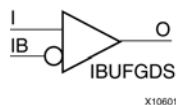
-- End of IBUFG_inst instantiation
```

## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)

## IBUFGDS

: Differential Signaling Dedicated Input Clock Buffer and Optional Delay



### 概要

このデザイン エLEMENT は、クロック バッファ (BUFG) または に接続するための専用の差動信号入力バッファです。IBUFGDS では、デザイン レベルのインターフェイス信号は、一方が「マスタ」で、もう一方が「スレーブ」となる 2 つの異なるポート (I、IB) で表されます。マスタとスレーブは MYNET\_P と MYNET\_N のように、同じ論理信号の反対の状態を示します。オプションで、プログラム可能な差動終端機能を使用すると、シグナル インテグリティが向上し、外部コンポーネントの数を減らすことができます。デバイスへの入力データの取り込みには、プログラマブル遅延を使用することもできます。

### 論理表

入力		出力
I	IB	O
0	0	変化なし
0	1	0
1	0	1
1	1	変化なし

### ポートの説明

ポート名	方向	幅	機能
O	出力	1	クロック バッファ出力
IB	入力	1	Diff_n クロック バッファの入力
I	入力	1	Diff_p クロック バッファの入力

### デザインの入力方法

インスタンスエーション	推奨
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

デザイン階層を保つために、すべての I/O コンポーネントを必ずデザインの最上位に配置してください。I ポートを直接デザインの最上位のマスタとなる入力ポートに、IB ポートを最上位のスレーブとなる入力ポートに、O ポートをこの入力をソースとする、BUFG、またはロジックに接続してください。一部の合成ツールでは、IBUFG を FPGA のクロックリソースに接続すると、必要に応じて BUFG が自動的に推論されます。generic/defparam 値を設定し、バッファのビヘイビアを適切に設定してください。

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
DIFF_TERM	ブール代数	TRUE、FALSE	FALSE	ビルトインの差動終端抵抗をイネーブル
IBUF_DELAY_VALUE	文字列	0 ~ 12	0	IOB からのレジスタを介さないパスに追加する遅延を指定
IOSTANDARD	文字列	データシートを参照	DEFAULT	エレメントに I/O 規格を割り当て

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IBUFGDS: Differential Global Clock Input Buffer
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IBUFGDS_inst : IBUFGDS
generic map (
    DIFF_TERM => FALSE, -- Differential Termination (Spartan-3E/5 only)
    IBUF_DELAY_VALUE => "0", -- Specify the amount of added input delay for buffer,
                             -- "0"-"12"
    IOSTANDARD => "DEFAULT")
port map (
    O => O, -- Clock buffer output
    I => I, -- Diff_p clock buffer input (connect directly to top-level port)
    IB => IB -- Diff_n clock buffer input (connect directly to top-level port)
);

-- End of IBUFGDS_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IBUFGDS: Differential Global Clock Input Buffer
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IBUFGDS_inst : IBUFGDS
generic map (
    DIFF_TERM => FALSE, -- Differential Termination (Spartan-3E/5 only)
    IBUF_DELAY_VALUE => "0", -- Specify the amount of added input delay for buffer,
                             -- "0"-"12"
    IOSTANDARD => "DEFAULT")
port map (
    O => O, -- Clock buffer output
    I => I, -- Diff_p clock buffer input (connect directly to top-level port)
    IB => IB -- Diff_n clock buffer input (connect directly to top-level port)
);

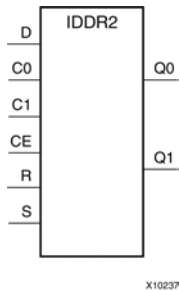
-- End of IBUFGDS_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## IDDR2

: Double Data Rate Input D Flip-Flop with Optional Data Alignment, Clock Enable and Programmable Synchronous or Asynchronous Set/Reset



### 概要

このデザイン エLEMENTは、ザイリンクス FPGA で外部デュアル データレート (DDR) 信号を受信するための専用入力レジスタです。C0 と C1 の 2 つのクロックを使用してコンポーネントに接続されるので、C0 および C1 の両方の立ち上がりエッジでデータが取り込まれます。IDDR2 は、レジスタの動作を停止するために使用できるアクティブ High のクロック イネーブル (CE) ポート、対応するクロックに同期または非同期になるよう設定できるセット/リセット ポートを備えています。また、オプションの調整機能を使用すると、コンポーネントへの両方の出力データ ポートを 1 つのクロックに揃えることができます。

### 論理表

入力						出力	
S	R	CE	D	C0	C1	Q0	Q1
1	X	X	X	X	X	INIT_Q0	INIT_Q1
0	1	X	X	X	X	not INIT_Q0	not INIT_Q1
0	0	0	X	X	X	変化なし	変化なし
0	0	1	D	↑	X	D	変化なし
0	0	1	D	X	↑	変化なし	D

セット/リセットは SRTYPE 値で同期に設定可能

### デザインの入力方法

インスタンス化	推奨
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

デフォルトの動作を変更するには、ジェネリック マップ (VHDL) またはパラメータ値代入 (Verilog) を使用して、インスタンス化されたコンポーネントの一部として属性を変更します。IDDR2 は、入力バッファが推論されるデザインの最上位入力ポートに接続するか、インスタンス化された IBUF、IOBUF、IBUFDS、または IOBUFDS のいずれかに直接接続できます。このコンポーネントのすべての入力と出力は、接続しておく必要があります。

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
DDR_ALIGNMENT	文字列	NONE、C0、C1	NONE	DDR レジスタの出力アライメントを設定 <ul style="list-style-type: none"> <li>NONE：対応する C0 または C1 の立ち上がりエッジのすぐ後で、Q0 および Q1 にデータが出力されます。</li> <li>C0：Q0 と Q1 両方のデータが C0 クロックの立ち上がりエッジに同期します。</li> <li>C1：Q0 と Q1 両方のデータが C1 クロックの立ち上がりエッジに同期します。</li> </ul>
INIT_Q0	整数	0、1	0	Q0 出力の初期値を 0 または 1 に設定
INIT_Q1	整数	0、1	0	Q1 出力の初期値を 0 または 1 に設定
SRTYPE	文字列	SYNC、ASYNC	SYNC	セット/リセットを SYNC または ASYNC に設定

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IDDR2: Input Double Data Rate Input Register with Set, Reset
--      and Clock Enable.
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IDDR2_inst : IDDR2
generic map(
  DDR_ALIGNMENT => "NONE", -- Sets output alignment to "NONE", "C0", "C1"
  INIT_Q0 => '0', -- Sets initial state of the Q0 output to '0' or '1'
  INIT_Q1 => '0', -- Sets initial state of the Q1 output to '0' or '1'
  SRTYPE => "SYNC") -- Specifies "SYNC" or "ASYNC" set/reset
port map (
  Q0 => Q0, -- 1-bit output captured with C0 clock
  Q1 => Q1, -- 1-bit output captured with C1 clock
  C0 => C0, -- 1-bit clock input
  C1 => C1, -- 1-bit clock input
  CE => CE, -- 1-bit clock enable input
  D => D, -- 1-bit data input
  R => R, -- 1-bit reset input
  S => S -- 1-bit set input
);

-- End of IDDR2_inst instantiation
```



## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IDDR2: Input Double Data Rate Input Register with Set, Reset
--       and Clock Enable.
--       Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IDDR2_inst : IDDR2
generic map(
  DDR_ALIGNMENT => "NONE", -- Sets output alignment to "NONE", "C0", "C1"
  INIT_Q0 => '0', -- Sets initial state of the Q0 output to '0' or '1'
  INIT_Q1 => '0', -- Sets initial state of the Q1 output to '0' or '1'
  SRTYPE => "SYNC") -- Specifies "SYNC" or "ASYN" set/reset
port map (
  Q0 => Q0, -- 1-bit output captured with C0 clock
  Q1 => Q1, -- 1-bit output captured with C1 clock
  C0 => C0, -- 1-bit clock input
  C1 => C1, -- 1-bit clock input
  CE => CE, -- 1-bit clock enable input
  D => D,   -- 1-bit data input
  R => R,   -- 1-bit reset input
  S => S    -- 1-bit set input
);

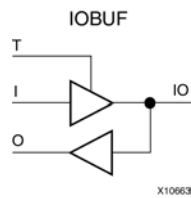
-- End of IDDR2_inst instantiation
```

## 詳細情報

[Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)

## IOBUF

: Bi-Directional Buffer



## 概要

このデザイン エLEMENTは双方向でシングルエンドの I/O バッファで、内部ロジックを外部双方向ピンに接続する場合に使用します。

## 論理表

入力		双方向	出力
T	I	I/O	O
1	X	Z	I/O
0	1	1	1
0	0	0	0

## ポートの説明

ポート名	方向	幅	機能
O	出力	1	バッファの出力
I/O	入出力	1	バッファの入出力
I	入力	1	バッファの入力
T	入力	1	トリステート イネーブル入力

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
DRIVE	整数	2、4、6、8、12、16、24	12	I/O 規格として LVTTTL、LVCMOS12、LVCMOS15、LVCMOS18、LVCMOS25 または LVCMOS33 を使用する SelectIO™ バッファの出力の駆動電流 (mA) を選択
IBUF_DELAY_VALUE	文字列	0 ～ 12	0	IOB からのレジスタを介さないパスに追加する遅延を指定
IFD_DELAY_VALUE	文字列	AUTO、0 ～ 6	AUTO	IOB 内のレジスタを介するパスに追加する遅延を指定
IOSTANDARD	文字列	データシートを参照	DEFAULT	エレメントに I/O 規格を割り当て
SLEW	文字列	SLOW、FAST、QUIETIO	SLOW	出力の立ち上がり時間と立ち下がり時間を設定。この属性の最適な設定方法は、データシートを参照してください。

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IOBUF: Single-ended Bi-directional Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IOBUF_inst : IOBUF
generic map (
  DRIVE => 12,
  IBUF_DELAY_VALUE => "0", -- Specify the amount of added input delay for buffer,
                           -- "0"-"12"
  IFD_DELAY_VALUE => "AUTO", -- Specify the amount of added delay for input register,
                           -- "AUTO", "0"-"6"
  IOSTANDARD => "DEFAULT",
  SLEW => "SLOW")
port map (
  O => O,      -- Buffer output
  IO => IO,    -- Buffer inout port (connect directly to top-level port)
  I => I,      -- Buffer input
  T => T       -- 3-state enable input, high=input, low=output
);

-- End of IOBUF_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IOBUF: Single-ended Bi-directional Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IOBUF_inst : IOBUF
generic map (
  DRIVE => 12,
  IBUF_DELAY_VALUE => "0", -- Specify the amount of added input delay for buffer,
                           -- "0"-"12"
```

```
IFD_DELAY_VALUE => "AUTO", -- Specify the amount of added delay for input register,
                        -- "AUTO", "0"-"6"
IOSTANDARD => "DEFAULT",
SLEW => "SLOW")
port map (
  O => O,      -- Buffer output
  IO => IO,    -- Buffer inout port (connect directly to top-level port)
  I => I,      -- Buffer input
  T => T       -- 3-state enable input, high=input, low=output
);

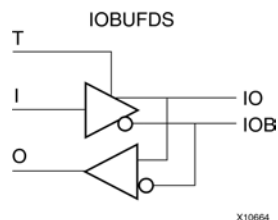
-- End of IOBUF_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## IOBUFDS

： 3-State Differential Signaling I/O Buffer with Active Low Output Enable



## 概要

このデザイン エLEMENTは、低電圧差動信号を使用する双方向バッファです。IOBUFDS では、デザイン レベルのインターフェイス信号は、一方が「マスタ」で、もう一方が「スレーブ」となる 2 つの異なるポート (IO、IOB) で表されます。マスタとスレーブは MYNET\_P と MYNET\_N のように、同じ論理信号の反対の状態を示します。オプションで、プログラム可能な差動終端機能を使用すると、シグナル インテグリティが向上し、外部コンポーネントの数を減らすことができます。デバイスへの入力データの取り込みには、プログラマブル遅延を使用することもできます。

## 論理表

入力		双方向		出力
I	T	I/O	IOB	O
X	1	Z	Z	変化なし
0	0	0	1	0
1	0	1	0	1

## ポートの説明

ポート名	方向	幅	機能
O	出力	1	バッファの出力
I/O	入出力	1	Diff_p 入出力
IOB	入出力	1	Diff_n 入出力
I	入力	1	バッファの入力
T	入力	1	トライステート イネーブル入力

## デザインの入力方法

インスタンス化	推奨
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
IBUF_DELAY_VALUE	文字列	0 ～ 12	0	IOB からのレジスタを介さないパスに追加する遅延を指定
IFD_DELAY_VALUE	文字列	AUTO、 0 ～ 6	AUTO	IOB 内のレジスタを介するパスに追加する遅延を指定
IOSTANDARD	文字列	データシートを参照	DEFAULT	エレメントに I/O 規格を割り当て

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- IOBUFDS: Differential Bi-directional Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IOBUFDS_inst : IOBUFDS
generic map (
    IBUF_DELAY_VALUE => "0", -- Specify the amount of added input delay for buffer,
                           -- "0"-"12"
    IFD_DELAY_VALUE  => "AUTO", -- Specify the amount of added delay for input register,
                           -- "AUTO", "0"-"6"
    IOSTANDARD        => "DEFAULT")
port map (
    O => O,      -- Buffer output
    IO => IO,    -- Diff_p inout (connect directly to top-level port)
    IOB => IOB,  -- Diff_n inout (connect directly to top-level port)
    I => I,      -- Buffer input
    T => T      -- 3-state enable input, high=input, low=output
);

-- End of IOBUFDS_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- IOBUFDS: Differential Bi-directional Buffer
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

IOBUFDS_inst : IOBUFDS
generic map (
  IBUF_DELAY_VALUE => "0", -- Specify the amount of added input delay for buffer,
                           -- "0"-"12"
  IFD_DELAY_VALUE  => "AUTO", -- Specify the amount of added delay for input register,
                           -- "AUTO", "0"-"6"
  IOSTANDARD       => "DEFAULT")
port map (
  O => O,      -- Buffer output
  IO => IO,    -- Diff_p inout (connect directly to top-level port)
  IOB => IOB,  -- Diff_n inout (connect directly to top-level port)
  I => I,      -- Buffer input
  T => T       -- 3-state enable input, high=input, low=output
);

-- End of IOBUFDS_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## KEEPER

: KEEPER Symbol



### 概要

このデザイン エLEMENTは、双方向出力ピンに接続されるネットの値を保持するウィークキーパ エLEMENTです。たとえば、ネットに対して論理値 1 を駆動すると、KEEPER はそのネットにウィーク/抵抗値 1 を駆動します。その後、ネットドライバがトライステートになっても、KEEPER はウィーク/抵抗値 1 を駆動し続けます。

### ポートの説明

ポート名	方向	幅	機能
O	出力	1 ビット	キーパ出力

### デザインの入力方法

インスタンス化	可
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- KEEPER: I/O Buffer Weak Keeper
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

KEEPER_inst : KEEPER
port map (
  O => O      -- Keeper output (connect directly to top-level port)
);

-- End of KEEPER_inst instantiation
```



## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- KEEPER: I/O Buffer Weak Keeper
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

KEEPER_inst : KEEPER
port map (
  O => O      -- Keeper output (connect directly to top-level port)
);

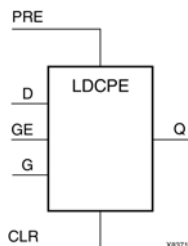
-- End of KEEPER_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## LDCPE

: Transparent Data Latch with Asynchronous Clear and Preset and Gate Enable



### 概要

このデザイン エLEMENTは、データ (D)、非同期クリア (CLR)、非同期プリセット (PRE)、ゲート イネーブル (GE) がある透過データ ラッチです。CLR が High になると、ほかの入力は無視され、データ出力 (Q) が Low にリセットされます。PRE が High、CLR が Low の場合、データ出力 (Q) は High にプリセットされます。ゲート入力 (G) と GE が High で CLR と PRE が Low の場合、Q にはデータ入力 (D) の値が出力されます。D 入力の値は、G が High から Low に切り替わるときにラッチ内に格納されます。Q 出力の値は、G または GE が Low の間は変化しません。

電力を供給すると、ラッチは非同期にクリアされ、出力が Low になります。FPGA では、グローバル セット/リセット (GSR) をアクティブにすると、電源投入時の状態をシミュレーションできます。GSR のデフォルトはアクティブ High ですが、STARTUP\_architecture シンボルの GSR 入力の前にインバータを追加するとアクティブ Low にできます。

### 論理表

入力					出力
CLR	PRE	GE	G	D	Q
1	X	X	X	X	0
0	1	X	X	X	1
0	0	0	X	X	変化なし
0	0	1	1	0	0
0	0	1	1	1	1
0	0	1	0	X	変化なし
0	0	1	↓	D	D

### ポートの説明

ポート名	方向	幅	機能
Q	出力	1	データ出力
CLR	入力	1	非同期クリア/リセット入力
D	入力	1	データ入力
G	入力	1	ゲート入力
GE	入力	1	ゲート イネーブル入力
PRE	入力	1	非同期プリセット/セット入力

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	整数	0、1	0	コンフィギュレーション後の Q 出力の初期値を指定。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LDCPE: Transparent latch with Asynchronous Reset, Preset and
--       Gate Enable.
--       Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LDCPE_inst : LDCPE
generic map (
  INIT => '0') -- Initial value of latch ('0' or '1')
port map (
  Q => Q,      -- Data output
  CLR => CLR,  -- Asynchronous clear/reset input
  D => D,      -- Data input
  G => G,      -- Gate input
  GE => GE,    -- Gate enable input
  PRE => PRE   -- Asynchronous preset/set input
);

-- End of LDCPE_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LDCPE: Transparent latch with Asynchronous Reset, Preset and
--       Gate Enable.
--       Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LDCPE_inst : LDCPE
generic map (
  INIT => '0') -- Initial value of latch ('0' or '1')
port map (
  Q => Q,        -- Data output
  CLR => CLR,    -- Asynchronous clear/reset input
  D => D,        -- Data input
  G => G,        -- Gate input
  GE => GE,     -- Gate enable input
  PRE => PRE    -- Asynchronous preset/set input
);

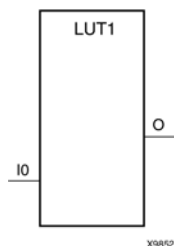
-- End of LDCPE_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## LUT1

: 1-Bit Look-Up Table with General Output



### 概要

このデザイン エLEMENTは一般出力 (O) を持つ 1 ビットのルックアップ テーブル (LUT) です。

LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。このELEMENTは、バッファまたはインバータの機能を果たします。これらのELEMENTは基本ブロックで、各 CLB スライスに 2 つ、各 CLB に 4 つずつあります。LUT には複数のバリエーションがあり、異なるタイミング モデルでレイアウト前のタイミング予測をより正確に行う必要がある場合に使用できます。

FPGA LUT プリミティブでは、INIT パラメータで論理値が設定されます。デフォルトは 0 で、入力値にかかわらず出力を 0 に駆動します (グラウンドとして機能)。ただし多くの場合、LUT プリミティブのロジック ファンクションを指定するために、新しい INIT の値を決定する必要があります。LUT の値を指定する方法には、次の 2 つがあります。

**論理表を使用する方法：**LUT の INIT 値を決定する一般的な方法。バイナリの論理表にすべての入力をリストして出力のロジック値を指定し、これらの出力値から、初期値を作成します。

**論理式を使用する方法：**リストされた論理表の値に対応する LUT の各入力にパラメータを定義し、パラメータを元にロジックの論理式を生成します。概念を理解してしまえばこの方法の方が簡単で、前出の方法のようにパラメータの指定にコードを使用する必要がありません。

### 論理表

入力	出力
I0	O
0	INIT[0]
1	INIT[1]
INIT = INIT 属性に割り当てられた 2 進数値	

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	2 ビット値	すべてゼロ	ルックアップ テーブルの初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT1: 1-input Look-Up Table with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT1_inst : LUT1
generic map (
  INIT => "00")
port map (
  O => O,    -- LUT general output
  I0 => I0   -- LUT input
);

-- End of LUT1_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT1: 1-input Look-Up Table with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT1_inst : LUT1
generic map (
  INIT => "00")
port map (
  O => O,    -- LUT general output
  I0 => I0   -- LUT input
);

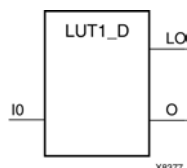
-- End of LUT1_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## LUT1\_D

: 1-Bit Look-Up Table with Dual Output



### 概要

このデザイン エLEMENTは 1 ビットのルックアップ テーブル (LUT) で、同じ機能を持つ O および LO という 2 つの出力があります。このELEMENTはバッファまたはインバータの機能を果たします。

出力 O は一般的なインターコネクトです。LO 出力は同じ CLB スライス内の別の出力、または高速バッファに接続します。LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。

FPGA LUT プリミティブでは、INIT パラメータで論理値が設定されます。デフォルトは 0 で、入力値にかかわらず出力を 0 に駆動します (グラウンドとして機能)。ただし多くの場合、LUT プリミティブのロジック ファンクションを指定するために、新しい INIT の値を決定する必要があります。LUT の値を指定する方法には、次の 2 つがあります。

**論理表を使用する方法：**LUT の INIT 値を決定する一般的な方法。バイナリの論理表にすべての入力をリストして出力のロジック値を指定し、これらの出力値から、初期値を作成します。

**論理式を使用する方法：**リストされた論理表の値に対応する LUT の各入力にパラメータを定義し、パラメータを元にロジックの論理式を生成します。概念を理解してしまえばこの方法の方が簡単で、前出の方法のようにパラメータの指定にコードを使用する必要がありません。

### 論理表

入力	出力	
I0	O	LO
0	INIT[0]	INIT[0]
1	INIT[1]	INIT[1]
INIT = INIT 属性に割り当てられた 2 進数値		

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	2 ビット値	すべてゼロ	ルックアップ テーブルの初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT1_D: 1-input Look-Up Table with general and local outputs
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT1_D_inst : LUT1_D
generic map (
  INIT => "00")
port map (
  LO => LO, -- LUT local output
  O  => O,  -- LUT general output
  IO => IO  -- LUT input
);

-- End of LUT1_D_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT1_D: 1-input Look-Up Table with general and local outputs
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT1_D_inst : LUT1_D
generic map (
  INIT => "00")
port map (
  LO => LO, -- LUT local output
  O  => O,  -- LUT general output
  IO => IO  -- LUT input
);

-- End of LUT1_D_inst instantiation
```

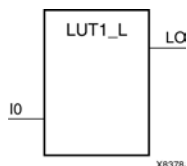
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## LUT1\_L

: 1-Bit Look-Up Table with Local Output



### 概要

このデザイン エLEMENT は、1 ビットのルックアップ テーブル (LUT) で、同じ CLB スライス内にある別の出力および高速バッファへの接続に使用するローカル出力 (LO) があります。このELEMENTはバッファまたはインバータの機能を果たします。

LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。

FPGA LUT プリミティブでは、INIT パラメータで論理値が設定されます。デフォルトは 0 で、入力値にかかわらず出力を 0 に駆動します (グラウンドとして機能)。ただし多くの場合、LUT プリミティブのロジック ファンクションを指定するために、新しい INIT の値を決定する必要があります。LUT の値を指定する方法には、次の 2 つがあります。

**論理表を使用する方法：**LUT の INIT 値を決定する一般的な方法。バイナリの論理表にすべての入力をリストして出力のロジック値を指定し、これらの出力値から、初期値を作成します。

**論理式を使用する方法：**リストされた論理表の値に対応する LUT の各入力にパラメータを定義し、パラメータを元にロジックの論理式を生成します。概念を理解してしまえばこの方法の方が簡単で、前出の方法のようにパラメータの指定にコードを使用する必要がありません。

### 論理表

入力	出力
I0	LO
0	INIT[0]
1	INIT[1]
INIT = INIT 属性に割り当てられた 2 進数値	

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	2 ビット値	すべてゼロ	ルックアップ テーブルの初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT1_L: 1-input Look-Up Table with local output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT1_L_inst : LUT1_L
generic map (
  INIT => "00")
port map (
  LO => LO, -- LUT local output
  IO => IO  -- LUT input
);

-- End of LUT1_L_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT1_L: 1-input Look-Up Table with local output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT1_L_inst : LUT1_L
generic map (
  INIT => "00")
port map (
  LO => LO, -- LUT local output
  IO => IO  -- LUT input
);

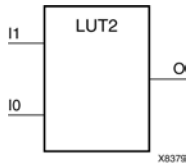
-- End of LUT1_L_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## LUT2

： 2-Bit Look-Up Table with General Output



### 概要

このデザイン エLEMENTは一般出力 (O) を持つ 2 ビットのルックアップ テーブル (LUT) です。

LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。このELEMENTは、バッファまたはインバータの機能を果たします。これらのELEMENTは基本ブロックで、各 CLB スライスに 2 つ、各 CLB に 4 つずつあります。LUT には複数のバリエーションがあり、異なるタイミング モデルでレイアウト前のタイミング予測をより正確に行う必要がある場合に使用できます。

FPGA LUT プリミティブでは、INIT パラメータで論理値が設定されます。デフォルトは 0 で、入力値にかかわらず出力を 0 に駆動します (グラウンドとして機能)。ただし多くの場合、LUT プリミティブのロジック ファンクションを指定するために、新しい INIT の値を決定する必要があります。LUT の値を指定する方法には、次の 2 つがあります。

**論理表を使用する方法：**LUT の INIT 値を決定する一般的な方法。バイナリの論理表にすべての入力をリストして出力のロジック値を指定し、これらの出力値から、初期値を作成します。

**論理式を使用する方法：**リストされた論理表の値に対応する LUT の各入力にパラメータを定義し、パラメータを元にロジックの論理式を生成します。概念を理解してしまえばこの方法の方が簡単で、前出の方法のようにパラメータの指定にコードを使用する必要がありません。

### 論理表

入力		出力
I1	I0	O
0	0	INIT[0]
0	1	INIT[1]
1	0	INIT[2]
1	1	INIT[3]
INIT = INIT 属性で指定された 16 進数値を 2 進数で表した値		

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	4 ビット値	すべてゼロ	ルックアップ テーブルの初期値を指定

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT2: 2-input Look-Up Table with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT2_inst : LUT2
generic map (
  INIT => X"0")
port map (
  O => O,    -- LUT general output
  I0 => I0,  -- LUT input
  I1 => I1   -- LUT input
);

-- End of LUT2_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT2: 2-input Look-Up Table with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT2_inst : LUT2
generic map (
  INIT => X"0")
port map (
  O => O,    -- LUT general output
  I0 => I0,  -- LUT input
  I1 => I1   -- LUT input
);

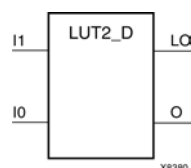
-- End of LUT2_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## LUT2\_D

: 2-Bit Look-Up Table with Dual Output



### 概要

このデザイン エLEMENTは 2 ビットのルックアップ テーブル (LUT) で、同じ機能を持つ O および LO という 2 つの出力があります。

出力 O は一般的なインターコネクトです。LO 出力は同じ CLB スライス内の別の出力、または高速バッファに接続します。LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。

FPGA LUT プリミティブでは、INIT パラメータで論理値が設定されます。デフォルトは 0 で、入力値にかかわらず出力を 0 に駆動します (グラウンドとして機能)。ただし多くの場合、LUT プリミティブのロジック ファンクションを指定するために、新しい INIT の値を決定する必要があります。LUT の値を指定する方法には、次の 2 つがあります。

**論理表を使用する方法：**LUT の INIT 値を決定する一般的な方法。バイナリの論理表にすべての入力をリストして出力のロジック値を指定し、これらの出力値から、初期値を作成します。

**論理式を使用する方法：**リストされた論理表の値に対応する LUT の各入力にパラメータを定義し、パラメータを元にロジックの論理式を生成します。概念を理解してしまえばこの方法の方が簡単で、前出の方法のようにパラメータの指定にコードを使用する必要がありません。

### 論理表

入力		出力	
I1	I0	O	LO
0	0	INIT[0]	INIT[0]
0	1	INIT[1]	INIT[1]
1	0	INIT[2]	INIT[2]
1	1	INIT[3]	INIT[3]
INIT = INIT 属性で指定された 16 進数値を 2 進数で表した値			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	4 ビット値	すべてゼロ	ルックアップ テーブルの初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT2_D: 2-input Look-Up Table with general and local outputs
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT2_D_inst : LUT2_D
generic map (
  INIT => X"0")
port map (
  LO => LO, -- LUT local output
  O  => O,  -- LUT general output
  I0 => I0, -- LUT input
  I1 => I1  -- LUT input
);

-- End of LUT2_D_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT2_D: 2-input Look-Up Table with general and local outputs
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT2_D_inst : LUT2_D
generic map (
  INIT => X"0")
port map (
  LO => LO, -- LUT local output
  O  => O,  -- LUT general output
  I0 => I0, -- LUT input
  I1 => I1  -- LUT input
);

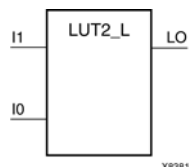
-- End of LUT2_D_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## LUT2\_L

: 2-Bit Look-Up Table with Local Output



### 概要

このデザイン エLEMENT は、2 ビットのルックアップ テーブル (LUT) で、同じ CLB スライス内にある別の出力および高速バッファへの接続に使用するローカル出力 (LO) があります。このELEMENTはバッファまたはインバータの機能を果たします。

LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。

FPGA LUT プリミティブでは、INIT パラメータで論理値が設定されます。デフォルトは 0 で、入力値にかかわらず出力を 0 に駆動します (グラウンドとして機能)。ただし多くの場合、LUT プリミティブのロジック ファンクションを指定するために、新しい INIT の値を決定する必要があります。LUT の値を指定する方法には、次の 2 つがあります。

**論理表を使用する方法：**LUT の INIT 値を決定する一般的な方法。バイナリの論理表にすべての入力をリストして出力のロジック値を指定し、これらの出力値から、初期値を作成します。

**論理式を使用する方法：**リストされた論理表の値に対応する LUT の各入力にパラメータを定義し、パラメータを元にロジックの論理式を生成します。概念を理解してしまえばこの方法の方が簡単で、前出の方法のようにパラメータの指定にコードを使用する必要がありません。

### 論理表

入力		出力
I1	I0	LO
0	0	INIT[0]
0	1	INIT[1]
1	0	INIT[2]
1	1	INIT[3]
INIT = INIT 属性で指定された 16 進数値を 2 進数で表した値		

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	4 ビット値	すべてゼロ	ルックアップ テーブルの初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT2_L: 2-input Look-Up Table with local output
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT2_L_inst : LUT2_L
generic map (
  INIT => X"0")
port map (
  LO => LO, -- LUT local output
  I0 => I0, -- LUT input
  I1 => I1  -- LUT input
);

-- End of LUT2_L_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT2_L: 2-input Look-Up Table with local output
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT2_L_inst : LUT2_L
generic map (
  INIT => X"0")
port map (
  LO => LO, -- LUT local output
  I0 => I0, -- LUT input
  I1 => I1  -- LUT input
);

-- End of LUT2_L_inst instantiation
```

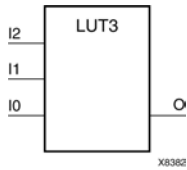
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## LUT3

： 3-Bit Look-Up Table with General Output



### 概要

このデザイン エLEMENTは一般出力 (O) を持つ 3 ビットのルックアップ テーブル (LUT) です。LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。

LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。このELEMENTは、バッファまたはインバータの機能を果たします。これらのELEMENTは基本ブロックで、各 CLB スライスに 2 つ、各 CLB に 4 つずつあります。LUT には複数のバリエーションがあり、異なるタイミング モデルでレイアウト前のタイミング予測をより正確に行う必要がある場合に使用できます。

FPGA LUT プリミティブでは、INIT パラメータで論理値が設定されます。デフォルトは 0 で、入力値にかかわらず出力を 0 に駆動します (グラウンドとして機能)。ただし多くの場合、LUT プリミティブのロジック ファンクションを指定するために、新しい INIT の値を決定する必要があります。LUT の値を指定する方法には、次の 2 つがあります。

**論理表を使用する方法：** LUT の INIT 値を決定する一般的な方法。バイナリの論理表にすべての入力をリストして出力のロジック値を指定し、これらの出力値から、初期値を作成します。

**論理式を使用する方法：** リストされた論理表の値に対応する LUT の各入力にパラメータを定義し、パラメータを元にロジックの論理式を生成します。概念を理解してしまえばこの方法の方が簡単で、前出の方法のようにパラメータの指定にコードを使用する必要がありません。

### 論理表

入力			出力
I2	I1	I0	O
0	0	0	INIT[0]
0	0	1	INIT[1]
0	1	0	INIT[2]
0	1	1	INIT[3]
1	0	0	INIT[4]
1	0	1	INIT[5]
1	1	0	INIT[6]
1	1	1	INIT[7]
INIT = INIT 属性で指定された 16 進数値を 2 進数で表した値			

## デザインの入力方法

インスタンシエーション	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	8 ビット値	すべてゼロ	ルックアップ テーブルの初期値を指定

## VHDL 記述 (インスタンシエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT3: 3-input Look-Up Table with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT3_inst : LUT3
generic map (
  INIT => X"00")
port map (
  O => O,    -- LUT general output
  I0 => I0,  -- LUT input
  I1 => I1,  -- LUT input
  I2 => I2   -- LUT input
);

-- End of LUT3_inst instantiation
```

## Verilog 記述 (インスタンシエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT3: 3-input Look-Up Table with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT3_inst : LUT3
generic map (
  INIT => X"00")
port map (
  O => O,    -- LUT general output
  I0 => I0,  -- LUT input
  I1 => I1,  -- LUT input
  I2 => I2   -- LUT input
);

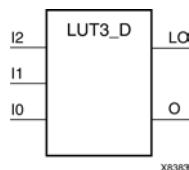
-- End of LUT3_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## LUT3\_D

: 3-Bit Look-Up Table with Dual Output



### 概要

このデザイン エLEMENTは 3 ビットのルックアップ テーブル (LUT) で、同じ機能を持つ O および LO という 2 つの出力があります。

出力 O は一般的なインターコネクトです。LO 出力は同じ CLB スライス内の別の出力、または高速バッファに接続します。LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。

FPGA LUT プリミティブでは、INIT パラメータで論理値が設定されます。デフォルトは 0 で、入力値にかかわらず出力を 0 に駆動します (グラウンドとして機能)。ただし多くの場合、LUT プリミティブのロジック ファンクションを指定するために、新しい INIT の値を決定する必要があります。LUT の値を指定する方法には、次の 2 つがあります。

**論理表を使用する方法：**LUT の INIT 値を決定する一般的な方法。バイナリの論理表にすべての入力をリストして出力のロジック値を指定し、これらの出力値から、初期値を作成します。

**論理式を使用する方法：**リストされた論理表の値に対応する LUT の各入力にパラメータを定義し、パラメータを元にロジックの論理式を生成します。概念を理解してしまえばこの方法の方が簡単で、前出の方法のようにパラメータの指定にコードを使用する必要がありません。

### 論理表

入力			出力	
I2	I1	I0	O	LO
0	0	0	INIT[0]	INIT[0]
0	0	1	INIT[1]	INIT[1]
0	1	0	INIT[2]	INIT[2]
0	1	1	INIT[3]	INIT[3]
1	0	0	INIT[4]	INIT[4]
1	0	1	INIT[5]	INIT[5]
1	1	0	INIT[6]	INIT[6]
1	1	1	INIT[7]	INIT[7]
INIT = INIT 属性で指定された 16 進数値を 2 進数で表した値				

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	8 ビット値	すべてゼロ	ルックアップ テーブルの初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT3_D: 3-input Look-Up Table with general and local outputs
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT3_D_inst : LUT3_D
generic map (
  INIT => X"00")
port map (
  LO => LO, -- LUT local output
  O  => O,  -- LUT general output
  I0 => I0, -- LUT input
  I1 => I1, -- LUT input
  I2 => I2  -- LUT input
);

-- End of LUT3_D_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT3_D: 3-input Look-Up Table with general and local outputs
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT3_D_inst : LUT3_D
generic map (
  INIT => X"00")
port map (
  LO => LO, -- LUT local output
  O  => O,  -- LUT general output
  I0 => I0, -- LUT input
  I1 => I1, -- LUT input
  I2 => I2  -- LUT input
);

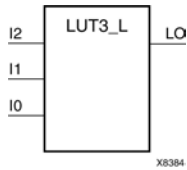
-- End of LUT3_D_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## LUT3\_L

： 3-Bit Look-Up Table with Local Output



### 概要

このデザイン エLEMENT は、3 ビットのルックアップ テーブル (LUT) で、同じ CLB スライス内にある別の出力および高速バッファへの接続に使用するローカル出力 (LO) があります。このELEMENTはバッファまたはインバータの機能を果たします。

LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。

FPGA LUT プリミティブでは、INIT パラメータで論理値が設定されます。デフォルトは 0 で、入力値にかかわらず出力を 0 に駆動します (グラウンドとして機能)。ただし多くの場合、LUT プリミティブのロジック ファンクションを指定するために、新しい INIT の値を決定する必要があります。LUT の値を指定する方法には、次の 2 つがあります。

**論理表を使用する方法：**LUT の INIT 値を決定する一般的な方法。バイナリの論理表にすべての入力をリストして出力のロジック値を指定し、これらの出力値から、初期値を作成します。

**論理式を使用する方法：**リストされた論理表の値に対応する LUT の各入力にパラメータを定義し、パラメータを元にロジックの論理式を生成します。概念を理解してしまえばこの方法の方が簡単で、前出の方法のようにパラメータの指定にコードを使用する必要がありません。

### 論理表

入力			出力
I2	I1	I0	LO
0	0	0	INIT[0]
0	0	1	INIT[1]
0	1	0	INIT[2]
0	1	1	INIT[3]
1	0	0	INIT[4]
1	0	1	INIT[5]
1	1	0	INIT[6]
1	1	1	INIT[7]
INIT = INIT 属性で指定された 16 進数値を 2 進数で表した値			

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	8 ビット値	すべてゼロ	ルックアップ テーブルの初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT3_L: 3-input Look-Up Table with local output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT3_L_inst : LUT3_L
generic map (
  INIT => X"00")
port map (
  LO => LO,    -- LUT local output
  I0 => I0,    -- LUT input
  I1 => I1,    -- LUT input
  I2 => I2     -- LUT input
);

-- End of LUT3_L_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT3_L: 3-input Look-Up Table with local output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT3_L_inst : LUT3_L
generic map (
  INIT => X"00")
port map (
  LO => LO,    -- LUT local output
  I0 => I0,    -- LUT input
  I1 => I1,    -- LUT input
  I2 => I2     -- LUT input
);

-- End of LUT3_L_inst instantiation
```

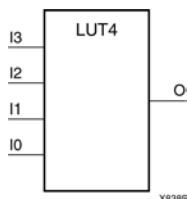


## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## LUT4

: 4-Bit Look-Up-Table with General Output



### 概要

このデザイン エLEMENTは一般出力 (O) を持つ 4 ビットのルックアップ テーブル (LUT) です。

LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。このELEMENTは、バッファまたはインバータの機能を果たします。これらのELEMENTは基本ブロックで、各 CLB スライスに 2 つ、各 CLB に 4 つずつあります。LUT には複数のバリエーションがあり、異なるタイミング モデルでレイアウト前のタイミング予測をより正確に行う必要がある場合に使用できます。

FPGA LUT プリミティブでは、INIT パラメータで論理値が設定されます。デフォルトは 0 で、入力値にかかわらず出力を 0 に駆動します (グラウンドとして機能)。ただし多くの場合、LUT プリミティブのロジック ファンクションを指定するために、新しい INIT の値を決定する必要があります。LUT の値を指定する方法には、次の 2 つがあります。

**論理表を使用する方法：**LUT の INIT 値を決定する一般的な方法。バイナリの論理表にすべての入力をリストして出力のロジック値を指定し、これらの出力値から、初期値を作成します。

**論理式を使用する方法：**リストされた論理表の値に対応する LUT の各入力にパラメータを定義し、パラメータを元にロジックの論理式を生成します。概念を理解してしまえばこの方法の方が簡単で、前出の方法のようにパラメータの指定にコードを使用する必要がありません。

## 論理表

入力				出力
I3	I2	I1	I0	O
0	0	0	0	INIT[0]
0	0	0	1	INIT[1]
0	0	1	0	INIT[2]
0	0	1	1	INIT[3]
0	1	0	0	INIT[4]
0	1	0	1	INIT[5]
0	1	1	0	INIT[6]
0	1	1	1	INIT[7]
1	0	0	0	INIT[8]
1	0	0	1	INIT[9]
1	0	1	0	INIT[10]
1	0	1	1	INIT[11]
1	1	0	0	INIT[12]
1	1	0	1	INIT[13]
1	1	1	0	INIT[14]
1	1	1	1	INIT[15]
INIT = INIT 属性で指定された 16 進数値を 2 進数で表した値				

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	ルックアップ テーブルの初期値を指定

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT4: 4-input Look-Up Table with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT4_inst : LUT4
generic map (
  INIT => X"0000")
port map (
  O => O,    -- LUT general output
  I0 => I0,  -- LUT input
  I1 => I1,  -- LUT input
  I2 => I2,  -- LUT input
  I3 => I3   -- LUT input
);

-- End of LUT4_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT4: 4-input Look-Up Table with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT4_inst : LUT4
generic map (
  INIT => X"0000")
port map (
  O => O,    -- LUT general output
  I0 => I0,  -- LUT input
  I1 => I1,  -- LUT input
  I2 => I2,  -- LUT input
  I3 => I3   -- LUT input
);

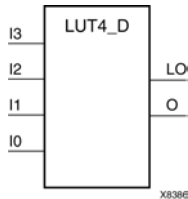
-- End of LUT4_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## LUT4\_D

: 4-Bit Look-Up Table with Dual Output



### 概要

このデザイン エLEMENTは 4 ビットのルックアップ テーブル (LUT) で、同じ機能を持つ O および LO という 2 つの出力があります。

出力 O は一般的なインターコネクトです。LO 出力は同じ CLB スライス内の別の出力、または高速バッファに接続します。LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。

FPGA LUT プリミティブでは、INIT パラメータで論理値が設定されます。デフォルトは 0 で、入力値にかかわらず出力を 0 に駆動します (グラウンドとして機能)。ただし多くの場合、LUT プリミティブのロジック ファンクションを指定するために、新しい INIT の値を決定する必要があります。LUT の値を指定する方法には、次の 2 つがあります。

**論理表を使用する方法：**LUT の INIT 値を決定する一般的な方法。バイナリの論理表にすべての入力をリストして出力のロジック値を指定し、これらの出力値から、初期値を作成します。

**論理式を使用する方法：**リストされた論理表の値に対応する LUT の各入力にパラメータを定義し、パラメータを元にロジックの論理式を生成します。概念を理解してしまえばこの方法の方が簡単で、前出の方法のようにパラメータの指定にコードを使用する必要がありません。

## 論理表

入力				出力	
I3	I2	I1	I0	O	LO
0	0	0	0	INIT[0]	INIT[0]
0	0	0	1	INIT[1]	INIT[1]
0	0	1	0	INIT[2]	INIT[2]
0	0	1	1	INIT[3]	INIT[3]
0	1	0	0	INIT[4]	INIT[4]
0	1	0	1	INIT[5]	INIT[5]
0	1	1	0	INIT[6]	INIT[6]
0	1	1	1	INIT[7]	INIT[7]
1	0	0	0	INIT[8]	INIT[8]
1	0	0	1	INIT[9]	INIT[9]
1	0	1	0	INIT[10]	INIT[10]
1	0	1	1	INIT[11]	INIT[11]
1	1	0	0	INIT[12]	INIT[12]
1	1	0	1	INIT[13]	INIT[13]
1	1	1	0	INIT[14]	INIT[14]
1	1	1	1	INIT[15]	INIT[15]

INIT = INIT 属性で指定された 16 進数値を 2 進数で表した値

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	ルックアップ テーブルの初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT4_D: 4-input Look-Up Table with general and local outputs
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT4_D_inst : LUT4_D
generic map (
  INIT => X"0000")
port map (
  LO => LO, -- LUT local output
  O  => O,  -- LUT general output
  I0 => I0, -- LUT input
  I1 => I1, -- LUT input
  I2 => I2, -- LUT input
  I3 => I3  -- LUT input
);

-- End of LUT4_D_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT4_D: 4-input Look-Up Table with general and local outputs
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT4_D_inst : LUT4_D
generic map (
  INIT => X"0000")
port map (
  LO => LO, -- LUT local output
  O  => O,  -- LUT general output
  I0 => I0, -- LUT input
  I1 => I1, -- LUT input
  I2 => I2, -- LUT input
  I3 => I3  -- LUT input
);

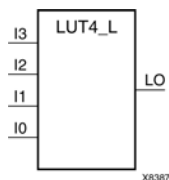
-- End of LUT4_D_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## LUT4\_L

: 4-Bit Look-Up Table with Local Output



### 概要

このデザイン エLEMENT は、4 ビットのルックアップ テーブル (LUT) で、同じ CLB スライス内にある別の出力および高速バッファへの接続に使用するローカル出力 (LO) があります。このELEMENTはバッファまたはインバータの機能を果たします。

LUT のファンクションを設定するため、INIT 属性を使用して、各入力値に対する出力値を 16 進数で指定する必要があります。

FPGA LUT プリミティブでは、INIT パラメータで論理値が設定されます。デフォルトは 0 で、入力値にかかわらず出力を 0 に駆動します (グラウンドとして機能)。ただし多くの場合、LUT プリミティブのロジック ファンクションを指定するために、新しい INIT の値を決定する必要があります。LUT の値を指定する方法には、次の 2 つがあります。

**論理表を使用する方法：**LUT の INIT 値を決定する一般的な方法。バイナリの論理表にすべての入力をリストして出力のロジック値を指定し、これらの出力値から、初期値を作成します。

**論理式を使用する方法：**リストされた論理表の値に対応する LUT の各入力にパラメータを定義し、パラメータを元にロジックの論理式を生成します。概念を理解してしまえばこの方法の方が簡単で、前出の方法のようにパラメータの指定にコードを使用する必要がありません。



## 論理表

入力				出力
I3	I2	I1	I0	LO
0	0	0	0	INIT[0]
0	0	0	1	INIT[1]
0	0	1	0	INIT[2]
0	0	1	1	INIT[3]
0	1	0	0	INIT[4]
0	1	0	1	INIT[5]
0	1	1	0	INIT[6]
0	1	1	1	INIT[7]
1	0	0	0	INIT[8]
1	0	0	1	INIT[9]
1	0	1	0	INIT[10]
1	0	1	1	INIT[11]
1	1	0	0	INIT[12]
1	1	0	1	INIT[13]
1	1	1	0	INIT[14]
1	1	1	1	INIT[15]

INIT = INIT 属性で指定された 16 進数値を 2 進数で表した値

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	ルックアップ テーブルの初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT4_L: 4-input Look-Up Table with local output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT4_L_inst : LUT4_L
generic map (
  INIT => X"0000")
port map (
  LO => LO, -- LUT local output
  I0 => I0, -- LUT input
  I1 => I1, -- LUT input
  I2 => I2, -- LUT input
  I3 => I3  -- LUT input
);

-- End of LUT4_L_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- LUT4_L: 4-input Look-Up Table with local output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

LUT4_L_inst : LUT4_L
generic map (
  INIT => X"0000")
port map (
  LO => LO, -- LUT local output
  I0 => I0, -- LUT input
  I1 => I1, -- LUT input
  I2 => I2, -- LUT input
  I3 => I3  -- LUT input
);

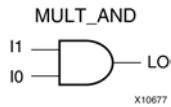
-- End of LUT4_L_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## MULT\_AND

: Fast Multiplier AND



### 概要

このデザイン エLEMENTはスライス内にある AND コンポーネントです。このスライスでは 2 つの入力が 4 入力 LUT と共有され、出力がキャリー ロジックに駆動しています。この追加のロジックはその他の目的にも使用できますが、高速で小型の乗算器の作成に特に有用です。I1 および I0 入力は、対応する LUT の I1 および I0 入力に接続する必要があります。LO 出力は、対応する MUXCY、MUXCY\_D、または MUXCY\_L の DI 入力に接続する必要があります。

### 論理表

入力		出力
I1	I0	LO
0	0	0
0	1	0
1	0	0
1	1	1

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MULT_AND: 2-input AND gate connected to Carry chain
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MULT_AND_inst : MULT_AND
port map (
  LO => LO,    -- MULT_AND output (connect to MUXCY DI)
  I0 => I0,    -- MULT_AND data[0] input
  I1 => I1     -- MULT_AND data[1] input
);

-- End of MULT_AND_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MULT_AND: 2-input AND gate connected to Carry chain
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MULT_AND_inst : MULT_AND
port map (
  LO => LO,    -- MULT_AND output (connect to MUXCY DI)
  I0 => I0,    -- MULT_AND data[0] input
  I1 => I1     -- MULT_AND data[1] input
);

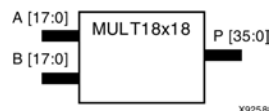
-- End of MULT_AND_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## MULT18X18

: 18 x 18 Signed Multiplier



### 概要

MULT18X18 は、組み合わせ符号付き 18 X 18 ビット乗算器です。18 ビット入力 A の値に 18 ビット入力 B の値を掛け合わせた積が、出力 P に 36 ビットで出力されます。

### 論理表

入力		出力
A	B	P
A	B	A X B
A、B、および P は 2 の補数です。		

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- MULT18X18: 18 x 18 signed asynchronous multiplier
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MULT18X18_inst : MULT18X18
port map (
  P => P,      -- 36-bit multiplier output
  A => A,      -- 18-bit multiplier input
  B => B       -- 18-bit multiplier input
);

-- End of MULT18X18_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MULT18X18: 18 x 18 signed asynchronous multiplier
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MULT18X18_inst : MULT18X18
port map (
  P => P,      -- 36-bit multiplier output
  A => A,      -- 18-bit multiplier input
  B => B       -- 18-bit multiplier input
);

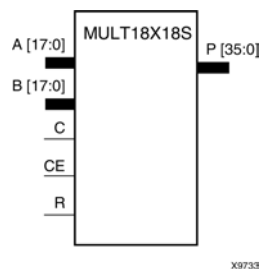
-- End of MULT18X18_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## MULT18X18S

: 18 x 18 Signed Multiplier – Registered Version



### 概要

MULT18X18S は、符号付き 18 X 18 乗算器 (MULT18X18) にレジスタを追加したもので、出力 (P)、データ入力 (A、B、C)、クロック イネーブル入力 (CE)、および同期リセット入力 (R) があります。レジスタは、GSR パルス後 0 に初期化されます。

18 ビット入力 A の値に 18 ビット入力 B の値を掛け合わせた積が、出力 P に 36 ビットで出力されます。

### 論理表

入力					出力
C	CE	Am	Bn	R	P
↑	X	X	X	1	0
↑	1	Am	Bn	0	A X B
X	0	X	X	0	変化なし

A、B、および P は 2 の補数です。

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MULT18X18S: 18 x 18 signed synchronous multiplier
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MULT18X18S_inst : MULT18X18S
port map (
  P => P,      -- 36-bit multiplier output
  A => A,      -- 18-bit multiplier input
  B => B,      -- 18-bit multiplier input
  C => C,      -- Clock input
  CE => CE,    -- Clock enable input
  R => R       -- Synchronous reset input
);

-- End of MULT18X18S_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MULT18X18S: 18 x 18 signed synchronous multiplier
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MULT18X18S_inst : MULT18X18S
port map (
  P => P,      -- 36-bit multiplier output
  A => A,      -- 18-bit multiplier input
  B => B,      -- 18-bit multiplier input
  C => C,      -- Clock input
  CE => CE,    -- Clock enable input
  R => R       -- Synchronous reset input
);

-- End of MULT18X18S_inst instantiation
```

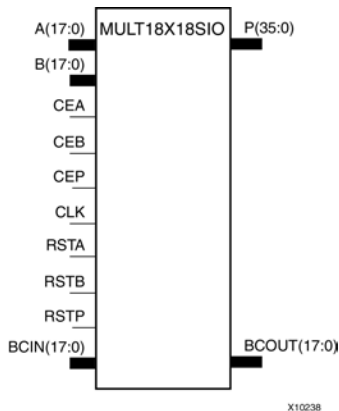
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## MULT18X18SIO

: 18 x 18 Cascadable Signed Multiplier with Optional Input and Output Registers, Clock Enable, and Synchronous Reset



### 概要

このデザイン エLEMENTは、36 ビット出力、18 X 18 ビット入力の専用の符号付き乗算器です。AREG、BREG、PREG 属性をすべて 0 に設定すると、非同期の乗算が実行されます。逆に属性をすべて 1 に設定すると、異なるレイテンシとパフォーマンス特性で同期の乗算が実行されます。同期乗算器を使用する場合、乗算器のレジスタ バンクの各セットに対してアクティブ High のクロック イネーブル (CEA、CEB、CEP) と同期リセット (RSTA、RSTB、RSTP) が使用されます。BCIN ポートと BCOUT ポートに B\_INPUT 属性を使用して MULT18X18SIO をカスケード接続すると、より大型の乗算ファンクションを作成できます。

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### 使用可能な属性

属性	タイプ	値	デフォルト	説明
AREG	整数	0, 1	1	A ポートで入力レジスタを使用するかどうかを指定します。0 の場合はレジスタが使用されず、1 の場合はレジスタが使用されます。
BREG	整数	0, 1	1	B ポートで入力レジスタを使用するかどうかを指定します。0 の場合はレジスタが使用されず、1 の場合はレジスタが使用されます。
B_INPUT	文字列	DIRECT、CASCADE	DIRECT	B ポートが FPGA に接続されている (DIRECT) か、別の MULT18X18SIO の BCOUT ポートに接続されているかを指定します。
PREG	整数	0, 1	1	乗算器の出力レジスタを使用するかどうかを指定します。0 の場合はレジスタが使用されず、1 の場合はレジスタが使用されます。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MULT18X18SIO: 18 x 18 cascaded, signed synchronous/asynchronous multiplier
--               Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MULT18X18SIO_inst : MULT18X18SIO
generic map (
  AREG => 1, -- Enable the input registers on the A port (1=on, 0=off)
  BREG => 1, -- Enable the input registers on the B port (1=on, 0=off)
  B_INPUT => "DIRECT", -- B cascade input "DIRECT" or "CASCADE"
  PREG => 1) -- Enable the input registers on the P port (1=on, 0=off)
port map (
  BCOUT => BCOUT, -- 18-bit cascade output
  P => P, -- 36-bit multiplier output
  A => A, -- 18-bit multiplier input
  B => B, -- 18-bit multiplier input
  BCIN => BCIN, -- 18-bit cascade input
  CEA => CEA, -- Clock enable input for the A port
  CEB => CEB, -- Clock enable input for the B port
  CEP => CEP, -- Clock enable input for the P port
  CLK => CLK, -- Clock input
  RSTA => RSTA, -- Synchronous reset input for the A port
  RSTB => RSTB, -- Synchronous reset input for the B port
  RSTP => RSTP, -- Synchronous reset input for the P port
);

-- End of MULT18X18SIO_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MULT18X18SIO: 18 x 18 cascaded, signed synchronous/asynchronous multiplier
--               Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MULT18X18SIO_inst : MULT18X18SIO
generic map (
  AREG => 1, -- Enable the input registers on the A port (1=on, 0=off)
  BREG => 1, -- Enable the input registers on the B port (1=on, 0=off)
  B_INPUT => "DIRECT", -- B cascade input "DIRECT" or "CASCADE"
  PREG => 1) -- Enable the input registers on the P port (1=on, 0=off)
port map (
  BCOUT => BCOUT, -- 18-bit cascade output
  P => P, -- 36-bit multiplier output
  A => A, -- 18-bit multiplier input
  B => B, -- 18-bit multiplier input
  BCIN => BCIN, -- 18-bit cascade input
  CEA => CEA, -- Clock enable input for the A port
  CEB => CEB, -- Clock enable input for the B port
  CEP => CEP, -- Clock enable input for the P port
  CLK => CLK, -- Clock input
  RSTA => RSTA, -- Synchronous reset input for the A port
  RSTB => RSTB, -- Synchronous reset input for the B port
  RSTP => RSTP, -- Synchronous reset input for the P port
);

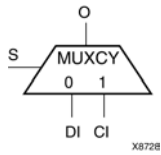
-- End of MULT18X18SIO_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## MUXCY

: 2-to-1 Multiplexer for Carry Logic with General Output



### 概要

スライスの直接入力 (DI) は、MUXCY の DI 入力に接続します。LC のキャリー入力 (CI) は、MUXCY の CI 入力に接続します。セレクト入力 (S) は、ルックアップ テーブル (LUT) の出力で駆動し、MUX ファンクションとしてコンフィギュレーションします。キャリー出力 (O) には選択された入力の値が出力され、各 LC のキャリー出力ファンクションをインプリメントします。S が Low の場合は DI が選択され、High の場合は CI が選択されます。

このほか、ローカル出力を持つ MUXCY\_D および MUXCY\_L があり、異なるタイミング モデルでレイアウト前のタイミング予測をより正確に行う必要がある場合に使用できます。

### 論理表

入力			出力
S	DI	CI	O
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXCY: Carry-Chain MUX with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXCY_inst : MUXCY
port map (
    O => O,    -- Carry output signal
    CI => CI,  -- Carry input signal
    DI => DI,  -- Data input signal
    S => S     -- MUX select, tie to '1' or LUT4 out
);

-- End of MUXCY_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXCY: Carry-Chain MUX with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXCY_inst : MUXCY
port map (
    O => O,    -- Carry output signal
    CI => CI,  -- Carry input signal
    DI => DI,  -- Data input signal
    S => S     -- MUX select, tie to '1' or LUT4 out
);

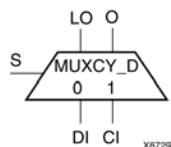
-- End of MUXCY_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## MUXCY\_D

： 2-to-1 Multiplexer for Carry Logic with Dual Output



### 概要

このデザイン エレメントは、1 ビットの高速キャリー伝搬ファンクションをインプリメントするために使用します。このようなファンクションは、1 つのロジック セル (LC) に 1 つずつインプリメントできるので、1 つの CLB に合計 4 ビットをインプリメントできます。LC の直接入力 (DI) は MUXCY\_D の DI 入力に接続し、LC のキャリー入力 (CI) は MUXCY\_D の CI 入力に接続します。セレクト入力 (S) は、ルックアップ テーブル (LUT) の出力で駆動し、XOR ファンクションとしてコンフィギュレーションします。キャリー出力 (O と LO) には選択された入力の値が出力され、各 LC のキャリー出力ファンクションをインプリメントします。S が Low の場合は DI が選択され、High の場合は CI が選択されます。

出力 O と LO は、機能的に同じです。出力 O は一般的なインターコネクトです。「MUXCY」および「MUXCY\_L」も参照してください。

### 論理表

入力			出力	
S	DI	CI	O	LO
0	1	X	1	1
0	0	X	0	0
1	X	1	1	1
1	X	0	0	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXCY_D: Carry-Chain MUX with general and local outputs
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXCY_D_inst : MUXCY_D
port map (
  LO => LO, -- Carry local output signal
  O  => O,  -- Carry general output signal
  CI => CI, -- Carry input signal
  DI => DI, -- Data input signal
  S  => S   -- MUX select, tie to '1' or LUT4 out
);

-- End of MUXCY_D_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXCY_D: Carry-Chain MUX with general and local outputs
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXCY_D_inst : MUXCY_D
port map (
  LO => LO, -- Carry local output signal
  O  => O,  -- Carry general output signal
  CI => CI, -- Carry input signal
  DI => DI, -- Data input signal
  S  => S   -- MUX select, tie to '1' or LUT4 out
);

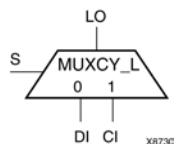
-- End of MUXCY_D_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## MUXCY\_L

: 2-to-1 Multiplexer for Carry Logic with Local Output



### 概要

このデザイン エレメントは、1 ビットの高速キャリー伝搬ファンクションをインプリメントするために使用します。このようなファンクションは、1 つのロジック セル (LC) に 1 つずつインプリメントできるので、1 つの CLB に合計 4 ビットをインプリメントできます。LC の直接入力 (DI) は MUXCY\_L の DI 入力に接続し、LC のキャリー入力 (CI) は MUXCY\_L の CI 入力に接続します。セレクト入力 (S) は、ルックアップ テーブル (LUT) の出力で駆動し、XOR ファンクションとしてコンフィギュレーションします。キャリー出力 (LO) には選択された入力の値が出力され、各 LC のキャリー出力ファンクションをインプリメントします。S が Low の場合は DI が選択され、High の場合は CI が選択されます。

「MUXCY」および「MUXCY\_D」も参照してください。

### 論理表

入力			出力
S	DI	CI	LO
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可



## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXCY_L: Carry-Chain MUX with local output
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXCY_L_inst : MUXCY_L
port map (
    LO => LO, -- Carry local output signal
    CI => CI, -- Carry input signal
    DI => DI, -- Data input signal
    S => S    -- MUX select, tie to '1' or LUT4 out
);

-- End of MUXCY_L_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXCY_L: Carry-Chain MUX with local output
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXCY_L_inst : MUXCY_L
port map (
    LO => LO, -- Carry local output signal
    CI => CI, -- Carry input signal
    DI => DI, -- Data input signal
    S => S    -- MUX select, tie to '1' or LUT4 out
);

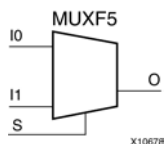
-- End of MUXCY_L_inst instantiation
```

## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)

## MUXF5

: 2-to-1 Look-Up Table Multiplexer with General Output



### 概要

このデザイン エLEMENTは、ルックアップ テーブルと組み合わせて、5 ファンクションのルックアップ テーブルまたは 4:1 マルチプレクサを作成するためのマルチプレクサ ファンクションを、CLB スライス を 1 つ使用してインプリメントします。I0 および I1 入力には、2 つのルックアップ テーブルのローカル出力 (LO) を接続します。セレクト入力 (S) は、どの内部ネットでも駆動できます。S が Low の場合は I0 が選択され、High の場合は I1 が選択されます。

このほか、ローカル出力を持つ MUXF5\_D および MUXF5\_L があり、異なるタイミング モデルでレイアウト前のタイミング予測をより正確に行う必要がある場合に使用できます。

### 論理表

入力			出力
S	I0	I1	O
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF5: Slice MUX to tie two LUT4's together with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF5_inst : MUXF5
port map (
    O => O,    -- Output of MUX to general routing
    I0 => I0,  -- Input (tie directly to the output of LUT4)
    I1 => I1,  -- Input (tie directly to the output of LUT4)
    S => S     -- Input select to MUX
);

-- End of MUXF5_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF5: Slice MUX to tie two LUT4's together with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF5_inst : MUXF5
port map (
    O => O,    -- Output of MUX to general routing
    I0 => I0,  -- Input (tie directly to the output of LUT4)
    I1 => I1,  -- Input (tie directly to the output of LUT4)
    S => S     -- Input select to MUX
);

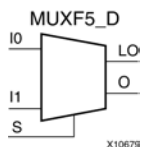
-- End of MUXF5_inst instantiation
```

## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)

## MUXF5\_D

: 2-to-1 Look-Up Table Multiplexer with Dual Output



### 概要

このデザイン エLEMENTは、ルックアップ テーブルと組み合わせて、5 ファンクションのルックアップ テーブルまたは 4:1 マルチプレクサを作成するためのマルチプレクサ ファンクションを、CLB スライス を 1 つ使用してインプリメントします。I0 および I1 入力には、2 つのルックアップ テーブルのローカル出力 (LO) を接続します。セレクト入力 (S) は、どの内部ネットでも駆動できます。S が Low の場合は I0 が選択され、High の場合は I1 が選択されます。

出力 O と LO は、機能的に同じです。出力 O は一般的なインターコネクトです。LO 出力は、同じ CLB スライス内にある別の入力との接続に使用します。「MUXF5」および「MUXF5\_L」も参照してください。

### 論理表

入力			出力	
S	I0	I1	O	LO
0	1	X	1	1
0	0	X	0	0
1	X	1	1	1
1	X	0	0	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF5_D: Slice MUX to tie two LUT4's together with general and local outputs
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF5_D_inst : MUXF5_D
port map (
  LO => LO,  -- Ouput of MUX to local routing
  O => O,    -- Output of MUX to general routing
  IO => IO,  -- Input (tie directly to the output of LUT4)
  I1 => I1,  -- Input (tie directoy to the output of LUT4)
  S => S     -- Input select to MUX
);

-- End of MUXF5_D_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF5_D: Slice MUX to tie two LUT4's together with general and local outputs
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF5_D_inst : MUXF5_D
port map (
  LO => LO,  -- Ouput of MUX to local routing
  O => O,    -- Output of MUX to general routing
  IO => IO,  -- Input (tie directly to the output of LUT4)
  I1 => I1,  -- Input (tie directoy to the output of LUT4)
  S => S     -- Input select to MUX
);

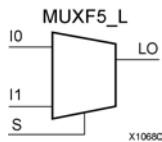
-- End of MUXF5_D_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## MUXF5\_L

: 2-to-1 Look-Up Table Multiplexer with Local Output



### 概要

このデザイン エLEMENTは、ルックアップ テーブルと組み合わせて、5 ファンクションのルックアップ テーブルまたは 4:1 マルチプレクサを作成するためのマルチプレクサ ファンクションを、CLB スライスを 1 つ使用してインプリメントします。I0 および I1 入力には、2 つのルックアップ テーブルのローカル出力 (LO) を接続します。セレクト入力 (S) は、どの内部ネットでも駆動できます。S が Low の場合は I0 が選択され、High の場合は I1 が選択されます。

LO 出力は、同じ CLB スライス内にある別の入力との接続に使用します。

「MUXF5」および「MUXF5\_D」も参照してください。

### 論理表

入力			出力
S	I0	I1	LO
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF5_L: Slice MUX to tie two LUT4's together with local output
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF5_L_inst : MUXF5_L
port map (
    LO => LO,  -- Output of MUX to local routing
    IO => IO,  -- Input (tie directly to the output of LUT4)
    I1 => I1,  -- Input (tie directly to the output of LUT4)
    S => S    -- Input select to MUX
);

-- End of MUXF5_L_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF5_L: Slice MUX to tie two LUT4's together with local output
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF5_L_inst : MUXF5_L
port map (
    LO => LO,  -- Output of MUX to local routing
    IO => IO,  -- Input (tie directly to the output of LUT4)
    I1 => I1,  -- Input (tie directly to the output of LUT4)
    S => S    -- Input select to MUX
);

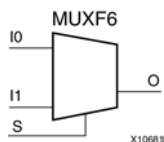
-- End of MUXF5_L_inst instantiation
```

## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)

## MUXF6

： 2-to-1 Look-Up Table Multiplexer with General Output



### 概要

このデザイン エLEMENTは、対応する 4 つのルックアップ テーブルと 2 つの MUXF5 を組み合わせて、6 ファンクションのルックアップ テーブルまたは 8:1 マルチプレクサを作成するためのマルチプレクサ ファンクションを、スライス 2 つにインプリメントします。I0 および I1 入力には、同じ CLB 内にある 2 つの MUXF5 のローカル出力 (LO) を接続します。セレクト入力 (S) は、どの内部ネットでも駆動できます。S が Low の場合は I0 が選択され、High の場合は I1 が選択されます。

このほか、ローカル出力を持つ MUXF6\_D および MUXF6\_L があり、異なるタイミング モデルでレイアウト前のタイミング予測をより正確に行う必要がある場合に使用できます。

### 論理表

入力			出力
S	I0	I1	O
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可



## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF6: CLB MUX to tie two MUXF5's together with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF6_inst : MUXF6
port map (
    O => O,      -- Output of MUX to general routing
    I0 => I0,     -- Input (tie to MUXF5 LO out)
    I1 => I1,     -- Input (tie to MUXF5 LO out)
    S => S       -- Input select to MUX
);

-- End of MUXF6_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF6: CLB MUX to tie two MUXF5's together with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF6_inst : MUXF6
port map (
    O => O,      -- Output of MUX to general routing
    I0 => I0,     -- Input (tie to MUXF5 LO out)
    I1 => I1,     -- Input (tie to MUXF5 LO out)
    S => S       -- Input select to MUX
);

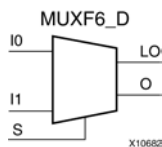
-- End of MUXF6_inst instantiation
```

## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)

## MUXF6\_D

: 2-to-1 Look-Up Table Multiplexer with Dual Output



### 概要

このデザイン エLEMENTは、対応する 4 つのルックアップ テーブルと 2 つの MUXF5 を組み合わせて、6 ファンクションのルックアップ テーブルまたは 8:1 マルチプレクサを作成するためのマルチプレクサ ファンクションを、スライス 2 つにインプリメントします。I0 および I1 入力には、同じ CLB 内にある 2 つの MUXF5 のローカル出力 (LO) を接続します。セレクト入力 (S) は、どの内部ネットでも駆動できます。S が Low の場合は I0 が選択され、High の場合は I1 が選択されます。

出力 O と LO は、機能的に同じです。出力 O は一般的なインターコネクトです。LO 出力は、同じ CLB スライス内にある別の入力との接続に使用します。

### 論理表

入力			出力	
S	I0	I1	O	LO
0	1	X	1	1
0	0	X	0	0
1	X	1	1	1
1	X	0	0	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF6_D: CLB MUX to tie two MUXF5's together with general and local outputs
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF6_D_inst : MUXF6_D
port map (
  LO => LO,  -- Output of MUX to local routing
  O  => O,   -- Output of MUX to general routing
  I0 => I0,  -- Input (tie to MUXF5 LO out)
  I1 => I1,  -- Input (tie to MUXF5 LO out)
  S  => S    -- Input select to MUX
);

-- End of MUXF6_D_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF6_D: CLB MUX to tie two MUXF5's together with general and local outputs
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF6_D_inst : MUXF6_D
port map (
  LO => LO,  -- Output of MUX to local routing
  O  => O,   -- Output of MUX to general routing
  I0 => I0,  -- Input (tie to MUXF5 LO out)
  I1 => I1,  -- Input (tie to MUXF5 LO out)
  S  => S    -- Input select to MUX
);

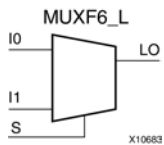
-- End of MUXF6_D_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## MUXF6\_L

: 2-to-1 Look-Up Table Multiplexer with Local Output



### 概要

このデザイン エLEMENTは、対応する 4 つのルックアップ テーブルと 2 つの MUXF5 を組み合わせて、6 ファンクションのルックアップ テーブルまたは 8:1 マルチプレクサを作成するためのマルチプレクサ ファンクションをインプリメントします。I0 および I1 入力には、同じ CLB 内にある 2 つの MUXF5 のローカル出力 (LO) を接続します。セレクト入力 (S) は、どの内部ネットでも駆動できます。S が Low の場合は I0 が選択され、High の場合は I1 が選択されます。

LO 出力は、同じ CLB スライス内にある別の入力との接続に使用します。

### 論理表

入力			出力
S	I0	I1	LO
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF6_L: CLB MUX to tie two MUXF5's together with local output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF6_L_inst : MUXF6_L
port map (
    LO => LO,  -- Output of MUX to local routing
    I0 => I0,  -- Input (tie to MUXF5 LO out)
    I1 => I1,  -- Input (tie to MUXF5 LO out)
    S => S     -- Input select to MUX
);

-- End of MUXF6_L_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF6_L: CLB MUX to tie two MUXF5's together with local output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF6_L_inst : MUXF6_L
port map (
    LO => LO,  -- Output of MUX to local routing
    I0 => I0,  -- Input (tie to MUXF5 LO out)
    I1 => I1,  -- Input (tie to MUXF5 LO out)
    S => S     -- Input select to MUX
);

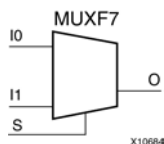
-- End of MUXF6_L_inst instantiation
```

## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)

## MUXF7

: 2-to-1 Look-Up Table Multiplexer with General Output



### 概要

このデザイン エレメントは、対応するルックアップ テーブルと組み合わせて、7 ファンクションのルックアップ テーブルまたは 8:1 マルチプレクサを作成するためのマルチプレクサ ファンクションをインプリメントします。I0 および I1 入力には、MUXF6 のローカル出力 (LO) を接続します。セレクト入力 (S) は、どの内部ネットでも駆動できます。S が Low の場合は I0 が選択され、High の場合は I1 が選択されます。

このほか、ローカル出力を持つ MUXF7\_D および MUXF7\_L があり、異なるタイミング モデルでレイアウト前のタイミング予測をより正確に行う必要がある場合に使用できます。

### 論理表

入力			出力
S	I0	I1	O
0	I0	X	I0
1	X	I1	I1
X	0	0	0
X	1	1	1

### ポートの説明

ポート名	方向	幅	機能
O	出力	1	汎用配線への MUX の出力
I0	入力	1	入力 (MUXF6 LO 出力に接続)
I1	入力	1	入力 (MUXF6 LO 出力に接続)
S	入力	1	MUX への入力セレクト

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF7: CLB MUX to tie two MUXF6's together with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF7_inst : MUXF7
port map (
    O => O,    -- Output of MUX to general routing
    I0 => I0,   -- Input (tie to MUXF6 LO out or LUT6 O6 pin)
    I1 => I1,   -- Input (tie to MUXF6 LO out or LUT6 O6 pin)
    S => S     -- Input select to MUX
);

-- End of MUXF7_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF7: CLB MUX to tie two MUXF6's together with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF7_inst : MUXF7
port map (
    O => O,    -- Output of MUX to general routing
    I0 => I0,   -- Input (tie to MUXF6 LO out or LUT6 O6 pin)
    I1 => I1,   -- Input (tie to MUXF6 LO out or LUT6 O6 pin)
    S => S     -- Input select to MUX
);

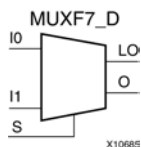
-- End of MUXF7_inst instantiation
```

## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)

## MUXF7\_D

: 2-to-1 Look-Up Table Multiplexer with Dual Output



### 概要

このデザイン エレメントは、対応するルックアップ テーブルと組み合わせて、7 ファンクションのルックアップ テーブルまたは 16:1 マルチプレクサを作成するためのマルチプレクサ ファンクションをインプリメントします。I0 および I1 入力には、MUXF6 のローカル出力 (LO) を接続します。セレクト入力 (S) は、どの内部ネットでも駆動できます。S が Low の場合は I0 が選択され、High の場合は I1 が選択されます。

出力 O と LO は、機能的に同じです。出力 O は一般的なインターコネクトです。LO 出力は、同じ CLB スライス内にある別の入力との接続に使用します。

### 論理表

入力			出力	
S	I0	I1	O	LO
0	I0	X	I0	I0
1	X	I1	I1	I1
X	0	0	0	0
X	1	1	1	1

### ポートの説明

ポート名	方向	幅	機能
O	出力	1	汎用配線への MUX の出力
LO	出力	1	ローカル配線への MUX の出力
I0	入力	1	入力 (MUXF6 LO 出力に接続)
I1	入力	1	入力 (MUXF6 LO 出力に接続)
S	入力	1	MUX への入力セレクト

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可



## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF7_D: CLB MUX to tie two MUXF6's together with general and local outputs
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF7_D_inst : MUXF7_D
port map (
    LO => LO,  -- Ouput of MUX to local routing
    O => O,    -- Output of MUX to general routing
    I0 => I0,  -- Input (tie to MUXF6 LO out or LUT6 O6 pin)
    I1 => I1,  -- Input (tie to MUXF6 LO out or LUT6 O6 pin)
    S => S     -- Input select to MUX
);

-- End of MUXF7_D_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF7_D: CLB MUX to tie two MUXF6's together with general and local outputs
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF7_D_inst : MUXF7_D
port map (
    LO => LO,  -- Ouput of MUX to local routing
    O => O,    -- Output of MUX to general routing
    I0 => I0,  -- Input (tie to MUXF6 LO out or LUT6 O6 pin)
    I1 => I1,  -- Input (tie to MUXF6 LO out or LUT6 O6 pin)
    S => S     -- Input select to MUX
);

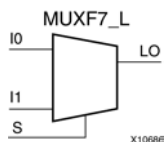
-- End of MUXF7_D_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## MUXF7\_L

: 2-to-1 look-up table Multiplexer with Local Output



### 概要

このデザイン エレメントは、対応するルックアップ テーブルと組み合わせて、7 ファンクションのルックアップ テーブルまたは 16:1 マルチプレクサを作成するためのマルチプレクサ ファンクションをインプリメントします。I0 および I1 入力には、MUXF6 のローカル出力 (LO) を接続します。セレクト入力 (S) は、どの内部ネットでも駆動できます。S が Low の場合は I0 が選択され、High の場合は I1 が選択されます。

LO 出力は、同じ CLB スライス内にある別の入力との接続に使用します。

### 論理表

入力			出力
S	I0	I1	LO
0	I0	X	I0
1	X	I1	I1
X	0	0	0
X	1	1	1

### ポートの説明

ポート名	方向	幅	機能
LO	出力	1	ローカル配線への MUX の出力
I0	入力	1	入力
I1	入力	1	入力
S	入力	1	MUX への入力セレクト

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF7_L: CLB MUX to tie two MUXF6's together with local output
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF7_L_inst : MUXF7_L
port map (
    LO => LO, -- Output of MUX to local routing
    IO => IO, -- Input (tie to MUXF6 LO out or LUT6 O6 pin)
    I1 => I1, -- Input (tie to MUXF6 LO out or LUT6 O6 pin)
    S => S    -- Input select to MUX
);

-- End of MUXF7_L_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF7_L: CLB MUX to tie two MUXF6's together with local output
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF7_L_inst : MUXF7_L
port map (
    LO => LO, -- Output of MUX to local routing
    IO => IO, -- Input (tie to MUXF6 LO out or LUT6 O6 pin)
    I1 => I1, -- Input (tie to MUXF6 LO out or LUT6 O6 pin)
    S => S    -- Input select to MUX
);

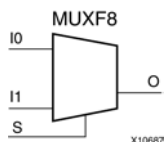
-- End of MUXF7_L_inst instantiation
```

## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)

## MUXF8

: 2-to-1 Look-Up Table Multiplexer with General Output



### 概要

このデザイン エLEMENTは、対応するルックアップ テーブルと MUXF5、MUXF6、および MUXF7 を組み合わせて、8 ファンクションのルックアップ テーブルまたは 16:1 マルチプレクサを作成するためのマルチプレクサ ファンクションを、CLB 2 個 (スライス 8 個) にインプリメントします。I0 および I1 入力には、MUXF7 のローカル出力 (LO) を接続します。セレクト入力 (S) は、どの内部ネットでも駆動できます。S が Low の場合は I0 が選択され、High の場合は I1 が選択されます。

### 論理表

入力			出力
S	I0	I1	O
0	I0	X	I0
1	X	I1	I1
X	0	0	0
X	1	1	1

### ポートの説明

ポート名	方向	幅	機能
O	出力	1	汎用配線への MUX の出力
I0	入力	1	入力 (MUXF7 LO 出力に接続)
I1	入力	1	入力 (MUXF7 LO 出力に接続)
S	入力	1	MUX への入力セレクト

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF8: CLB MUX to tie two MUXF7's together with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF8_inst : MUXF8
port map (
    O => O,      -- Output of MUX to general routing
    I0 => I0,     -- Input (tie to MUXF7 LO out)
    I1 => I1,     -- Input (tie to MUXF7 LO out)
    S => S       -- Input select to MUX
);

-- End of MUXF8_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF8: CLB MUX to tie two MUXF7's together with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF8_inst : MUXF8
port map (
    O => O,      -- Output of MUX to general routing
    I0 => I0,     -- Input (tie to MUXF7 LO out)
    I1 => I1,     -- Input (tie to MUXF7 LO out)
    S => S       -- Input select to MUX
);

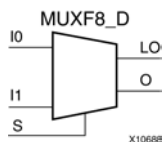
-- End of MUXF8_inst instantiation
```

## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)

## MUXF8\_D

: 2-to-1 Look-Up Table Multiplexer with Dual Output



### 概要

このデザイン エLEMENTは、対応するルックアップ テーブル 4 つと MUXF8 を 2 つ組み合わせて、8 ファンクションのルックアップ テーブルまたは 32:1 マルチプレクサを作成するためのマルチプレクサ ファンクションを、CLB 2 個 (スライス 8 個) にインプリメントします。I0 および I1 入力には、MUXF7 のローカル出力 (LO) を接続します。セレクト入力 (S) は、どの内部ネットでも駆動できます。S が Low の場合は I0 が選択され、High の場合は I1 が選択されます。

出力 O と LO は、機能的に同じです。出力 O は一般的なインターコネクトです。LO 出力は、同じ CLB スライス内にある別の入力との接続に使用します。

### 論理表

入力			出力	
S	I0	I1	O	LO
0	I0	X	I0	I0
1	X	I1	I1	I1
X	0	0	0	0
X	1	1	1	1

### ポートの説明

ポート名	方向	幅	機能
O	出力	1	汎用配線への MUX の出力
LO	出力	1	ローカル配線への MUX の出力
I0	入力	1	入力 (MUXF7 LO 出力に接続)
I1	入力	1	入力 (MUXF7 LO 出力に接続)
S	入力	1	MUX への入力セレクト

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF8_D: CLB MUX to tie two MUXF7's together with general and local outputs
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF8_D_inst : MUXF8_D
port map (
    LO => LO,  -- Ouput of MUX to local routing
    O  => O,   -- Output of MUX to general routing
    IO => IO,  -- Input (tie to MUXF7 LO out)
    I1 => I1,  -- Input (tie to MUXF7 LO out)
    S  => S    -- Input select to MUX
);

-- End of MUXF8_D_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF8_D: CLB MUX to tie two MUXF7's together with general and local outputs
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF8_D_inst : MUXF8_D
port map (
    LO => LO,  -- Ouput of MUX to local routing
    O  => O,   -- Output of MUX to general routing
    IO => IO,  -- Input (tie to MUXF7 LO out)
    I1 => I1,  -- Input (tie to MUXF7 LO out)
    S  => S    -- Input select to MUX
);

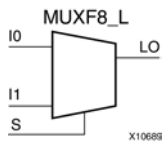
-- End of MUXF8_D_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## MUXF8\_L

： 2-to-1 Look-Up Table Multiplexer with Local Output



### 概要

このデザイン エレメントは、対応するルックアップ テーブル 4 つと MUXF8 を 2 つ組み合わせて、8 ファンクションのルックアップ テーブルまたは 32:1 マルチプレクサを作成するためのマルチプレクサ ファンクションを、CLB 2 個（スライス 8 個）にインプリメントします。I0 および I1 入力には、MUXF7 のローカル出力 (LO) を接続します。セレクト入力 (S) は、どの内部ネットでも駆動できます。S が Low の場合は I0 が選択され、High の場合は I1 が選択されます。

LO 出力は、同じ CLB スライス内にある別の入力との接続に使用します。

### 論理表

入力			出力
S	I0	I1	LO
0	I0	X	I0
1	X	I1	I1
X	0	0	0
X	1	1	1

### ポートの説明

ポート名	方向	幅	機能
LO	出力	1	ローカル配線への MUX の出力
I0	入力	1	入力 (MUXF7 LO 出力に接続)
I1	入力	1	入力 (MUXF7 LO 出力に接続)
S	入力	1	MUX への入力セレクト

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可



## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF8_L: CLB MUX to tie two MUXF7's together with local output
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF8_L_inst : MUXF8_L
port map (
    LO => LO,  -- Output of MUX to local routing
    I0 => I0,  -- Input (tie to MUXF7 LO out)
    I1 => I1,  -- Input (tie to MUXF7 LO out)
    S => S     -- Input select to MUX
);

-- End of MUXF8_L_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- MUXF8_L: CLB MUX to tie two MUXF7's together with local output
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

MUXF8_L_inst : MUXF8_L
port map (
    LO => LO,  -- Output of MUX to local routing
    I0 => I0,  -- Input (tie to MUXF7 LO out)
    I1 => I1,  -- Input (tie to MUXF7 LO out)
    S => S     -- Input select to MUX
);

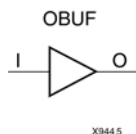
-- End of MUXF8_L_inst instantiation
```

## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)

## OBUF

: Output Buffer



### 概要

このデザイン エLEMENTは単純な出力バッファで、出力信号を、トリステートでない FPGA デバイス ピンに駆動するために使用します。デザインのすべての出力ポートに OBUF、OBUFT、OBUFDS、OBUFTDS のいずれかを接続する必要があります。

このELEMENTは内部回路を外部から分離し、チップから出力する信号の駆動電流を供給します。I/O ブロック (IOB) 内にあります。出力 (O) は、OPAD または IOPAD に接続されます。このELEMENTでは、LVTTL 規格が使用され、DRIVE 制約と SLOW または FAST 制約を使用して駆動電流とスルー レートを選択できます。デフォルトでは、DRIVE=12mA、スルー レートは SLOW に設定されています。

### ポートの説明

ポート名	方向	幅	機能
O	出力	1	最上位出力ポートに直接接続される OBUF の出力
I	入力	1	OBUF の入力。出力ポートを駆動するロジックに接続

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### 使用可能な属性

属性	タイプ	値	デフォルト	説明
DRIVE	整数	2、4、6、8、12、16、24	12	出力の駆動電流を指定します。許容範囲で最も低い値を使用してください。
IOSTANDARD	文字列	データシートを参照	DEFAULT	ELEMENTに I/O 規格を割り当てます。
SLEW	文字列	SLOW、FAST	SLOW	出力ドライバのスルー レートを指定します。この属性の最適な設定方法は、データシートを参照してください。

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- OBUF: Single-ended Output Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

OBUF_inst : OBUF
generic map (
    DRIVE => 12,
    IOSTANDARD => "DEFAULT",
    SLEW => "SLOW")
port map (
    O => O,      -- Buffer output (connect directly to top-level port)
    I => I       -- Buffer input
);

-- End of OBUF_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- OBUF: Single-ended Output Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

OBUF_inst : OBUF
generic map (
    DRIVE => 12,
    IOSTANDARD => "DEFAULT",
    SLEW => "SLOW")
port map (
    O => O,      -- Buffer output (connect directly to top-level port)
    I => I       -- Buffer input
);

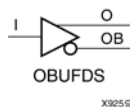
-- End of OBUF_inst instantiation
```

## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)

## OBUFDS

: Differential Signaling Output Buffer



### 概要

このデザイン エレメントは、低電圧の差動信号 (1.8V CMOS) をサポートする単一の出力バッファです。内部回路を外部から分離し、チップから出力する信号の駆動電流を供給します。出力には 2 つの異なるポート (O および OB) があり、これらのポートをそれぞれ「マスタ」、「スレーブ」と呼びます。マスタとスレーブは MYNET と MYNETB のように、同じ論理信号の反対の状態を示します。

### 論理表

入力	出力	
I	O	OB
0	0	1
1	1	0

### ポートの説明

ポート名	方向	幅	機能
O	出力	1	Diff_p 出力 (最上位ポートに直接接続)
OB	出力	1	Diff_n 出力 (最上位ポートに直接接続)
I	入力	1	バッファの入力

### デザインの入力方法

インスタンス化	推奨
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### 使用可能な属性

属性	タイプ	値	デフォルト	説明
IOSTANDARD	文字列	データシートを参照	DEFAULT	エレメントに I/O 規格を割り当てます。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- OBUFDS: Differential Output Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

OBUFDS_inst : OBUFDS
generic map (
    IOSTANDARD => "DEFAULT")
port map (
    O => O,      -- Diff_p output (connect directly to top-level port)
    OB => OB,    -- Diff_n output (connect directly to top-level port)
    I => I       -- Buffer input
);

-- End of OBUFDS_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- OBUFDS: Differential Output Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

OBUFDS_inst : OBUFDS
generic map (
    IOSTANDARD => "DEFAULT")
port map (
    O => O,      -- Diff_p output (connect directly to top-level port)
    OB => OB,    -- Diff_n output (connect directly to top-level port)
    I => I       -- Buffer input
);

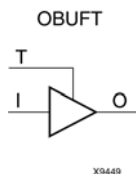
-- End of OBUFDS_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## OBUFT

: 3-State Output Buffer with Active Low Output Enable



### 概要

このデザイン エレメントは、入力 (I)、出力 (O)、アクティブ Low 出力イネーブル (T) を持つ単一のトリステート出力バッファです。このエレメントでは、LVTTL 規格が使用され、DRIVE 制約と SLOW または FAST 制約を使用して駆動電流とスルー レートを選択できます。デフォルトでは、DRIVE=12mA、スルー レートは SLOW に設定されています。

T が Low の場合、バッファに入力された値が対応する出力に送られます。T が High の場合は、出力がハイ インピーダンス (オフまたは Z ステート) になります。OBUFT は、双方向 I/O を作成するなど、トリステート機能にシングルエンド出力を使用する必要がある場合に使用します。

### 論理表

入力		出力
T	I	O
1	X	Z
0	1	1
0	0	0

### ポートの説明

ポート名	方向	幅	機能
O	出力	1	バッファ出力 (最上位ポートに直接接続)
I	入力	1	バッファの入力
T	入力	1	トリステート イネーブル入力

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
DRIVE	整数	2、4、6、8、12、16、24	12	出力の駆動電流を指定します。許容範囲で最も低い値を使用してください。
IOSTANDARD	文字列	データシートを参照	DEFAULT	エレメントに I/O 規格を割り当てます。
SLEW	文字列	SLOW、FAST	SLOW	出力ドライバのスループートを指定。この属性の最適な設定方法は、データシートを参照してください。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- OBUFT: Single-ended 3-state Output Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

OBUFT_inst : OBUFT
generic map (
    DRIVE => 12,
    IOSTANDARD => "DEFAULT",
    SLEW => "SLOW")
port map (
    O => O,      -- Buffer output (connect directly to top-level port)
    I => I,      -- Buffer input
    T => T       -- 3-state enable input
);

-- End of OBUFT_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- OBUFT: Single-ended 3-state Output Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

OBUFT_inst : OBUFT
generic map (
    DRIVE => 12,
    IOSTANDARD => "DEFAULT",
    SLEW => "SLOW")
port map (
    O => O,      -- Buffer output (connect directly to top-level port)
    I => I,      -- Buffer input
    T => T       -- 3-state enable input
);

-- End of OBUFT_inst instantiation
```

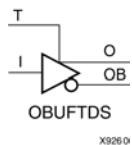
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリ データシート](#)



## OBUFTDS

： 3-State Output Buffer with Differential Signaling, Active-Low Output Enable



### 概要

このデザイン エレメントは、低電圧差動信号を使用する出力バッファです。OBUFTDS では、デザイン レベルのインターフェイス信号は、一方が「マスタ」で、もう一方が「スレーブ」となる 2 つの異なるポート (O、OB) で表されます。マスタとスレーブは MYNET\_P と MYNET\_N のように、同じ論理信号の反対の状態を示します。

### 論理表

入力		出力	
I	T	O	OB
X	1	Z	Z
0	0	0	1
1	0	1	0

### ポートの説明

ポート名	方向	幅	機能
O	出力	1	Diff_p 出力 (最上位ポートに直接接続)
OB	出力	1	Diff_n 出力 (最上位ポートに直接接続)
I	入力	1	バッファの入力
T	入力	1	トライステート イネーブル入力

### デザインの入力方法

インスタンスエーション	推奨
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### 使用可能な属性

属性	タイプ	値	デフォルト	説明
IOSTANDARD	文字列	データシートを参照	DEFAULT	I/O 規格をエレメントに割り当て

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- OBUFTDS: Differential 3-state Output Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

OBUFTDS_inst : OBUFTDS
generic map (
  IOSTANDARD => "DEFAULT")
port map (
  O => O,      -- Diff_p output (connect directly to top-level port)
  OB => OB,    -- Diff_n output (connect directly to top-level port)
  I => I,      -- Buffer input
  T => T       -- 3-state enable input
);

-- End of OBUFTDS_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- OBUFTDS: Differential 3-state Output Buffer
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

OBUFTDS_inst : OBUFTDS
generic map (
  IOSTANDARD => "DEFAULT")
port map (
  O => O,      -- Diff_p output (connect directly to top-level port)
  OB => OB,    -- Diff_n output (connect directly to top-level port)
  I => I,      -- Buffer input
  T => T       -- 3-state enable input
);

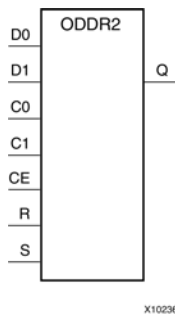
-- End of OBUFTDS_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## ODDR2

: Dual Data Rate Output D Flip-Flop with Optional Data Alignment, Clock Enable and Programmable Synchronous or Asynchronous Set/Reset



### 概要

このデザイン エLEMENTは、出力ダブル データ レート (DDR) レジスタで、FPGA から出力されるダブル データ レート 信号を生成するために使用します。ODDR2 は、C0 と C1 の 2 つのクロックを使用してコンポーネントに接続されるので、C0 および C1 の両方の立ち上がりエッジでデータが出力されます。ODDR2 は、レジスタの動作を停止するために使用できるアクティブ High のクロック イネーブル (CE) ポート、対応するクロックに同期または非同期になるよう設定できるセット/リセット ポートを備えています。ODDR2 には、1 クロックで取り込まれたデータを 2 クロックで出力するオプションの調整機能があります。

### 論理表

入力							出力
S	R	CE	D0	D1	C0	C1	O
1	X	X	X	X	X	X	1
0	1	X	X	X	X	X	not INIT
0	0	0	X	X	X	X	変化なし
0	0	1	D0	X	↑	X	D0
0	0	1	X	D1	X	↑	D1

セット/リセットは SRTYPE 値で同期に設定可能

### デザインの入力方法

インスタンシエーション	推奨
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
DDR_ALIGNMENT	文字列	NONE、C0、C1	NONE	DDR レジスタの入力キャプチャ ビヘイビアを設定します。NONE に設定すると、C0 クロックが Low から High に切り替わる時は D0 入力に、C1 クロックが Low から High に切り替わる時は D1 に データを入力します。C0 では、D0 と D1 両方への入力が C0 クロックの立ち上がりエッジに同期します。C1 では、D0 と D1 両方への入力が C1 クロックの立ち上がりエッジに同期します。
INIT	整数	0、1	0	Q0 出力の初期値を 0 または 1 に設定
SRTYPE	文字列	SYNC、ASYNC	SYNC	セット/リセットを SYNC または ASYNC に設定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- ODDR2: Output Double Data Rate Output Register with Set, Reset
--      and Clock Enable.
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

ODDR2_inst : ODDR2
generic map(
  DDR_ALIGNMENT => "NONE", -- Sets output alignment to "NONE", "C0", "C1"
  INIT => '0', -- Sets initial state of the Q output to '0' or '1'
  SRTYPE => "SYNC") -- Specifies "SYNC" or "ASYNC" set/reset
port map (
  Q => Q, -- 1-bit output data
  C0 => C0, -- 1-bit clock input
  C1 => C1, -- 1-bit clock input
  CE => CE, -- 1-bit clock enable input
  D0 => D0, -- 1-bit data input (associated with C0)
  D1 => D1, -- 1-bit data input (associated with C1)
  R => R, -- 1-bit reset input
  S => S -- 1-bit set input
);

-- End of ODDR2_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- ODDR2: Output Double Data Rate Output Register with Set, Reset
--      and Clock Enable.
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

ODDR2_inst : ODDR2
generic map(
  DDR_ALIGNMENT => "NONE", -- Sets output alignment to "NONE", "C0", "C1"
  INIT => '0', -- Sets initial state of the Q output to '0' or '1'
  SRTYPE => "SYNC") -- Specifies "SYNC" or "ASYNC" set/reset
port map (
```

```
Q => Q, -- 1-bit output data
C0 => C0, -- 1-bit clock input
C1 => C1, -- 1-bit clock input
CE => CE, -- 1-bit clock enable input
D0 => D0, -- 1-bit data input (associated with C0)
D1 => D1, -- 1-bit data input (associated with C1)
R => R, -- 1-bit reset input
S => S -- 1-bit set input
);

-- End of ODDR2_inst instantiation
```

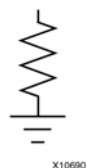
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## PULLDOWN

: Resistor to GND for Input Pads, Open-Drain, and 3-State Outputs

PULLDOWN



### 概要

この抵抗エレメントは、入力、出力、双方向のパッドに接続し、フロートする可能性のあるノードのロジックレベルを Low にします。

### ポートの説明

ポート名	方向	幅	機能
O	出力	1	プルダウン出力 (最上位ポートに直接接続)

### デザインの入力方法

インスタンシエーション	可
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンシエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- PULLDOWN: I/O Buffer Weak Pull-down
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

PULLDOWN_inst : PULLDOWN
port map (
  O => O      -- Pulldown output (connect directly to top-level port)
);

-- End of PULLDOWN_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- PULLDOWN: I/O Buffer Weak Pull-down
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

PULLDOWN_inst : PULLDOWN
port map (
  O => O      -- Pulldown output (connect directly to top-level port)
);

-- End of PULLDOWN_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## PULLUP

: Resistor to VCC for Input PADs, Open-Drain, and 3-State Outputs



### 概要

このデザイン エレメントは、1 つの入力、トライステート出力、または双方向ポートが内部または外部ソースで駆動されないときに、値、weak High で駆動できます。このエレメントは、すべてのドライバが使用されていないときにオープンドレイン エレメントおよびマクロのロジック レベルを 1 (High) にします。

### ポートの説明

ポート名	方向	幅	機能
O	出力	1	プルアップ出力 (最上位ポートに直接接続)

### デザインの入力方法

インスタンス化	可
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- PULLUP: I/O Buffer Weak Pull-up
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

PULLUP_inst : PULLUP
port map (
  O => O      -- Pullup output (connect directly to top-level port)
);

-- End of PULLUP_inst instantiation
```



## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

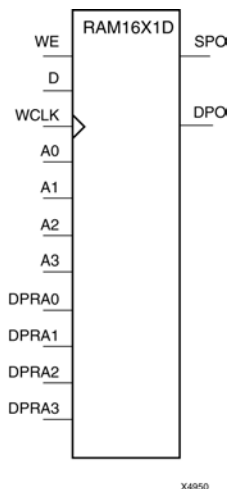
```
Library UNISIM;  
use UNISIM.vcomponents.all;  
  
-- PULLUP: I/O Buffer Weak Pull-up  
--      Spartan-3E  
-- Xilinx HDL Libraries Guide, version 12.2  
  
PULLUP_inst : PULLUP  
port map (  
    O => O      -- Pullup output (connect directly to top-level port)  
);  
  
-- End of PULLUP_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM16X1D

: 16-Deep by 1-Wide Static Dual Port Synchronous RAM



### 概要

このエレメントは 16 ワード X 1 ビットのデュアル ポート SRAM で、同期書き込み機能を備えています。デバイスには、読み出しアドレス (DPRA3 ~ DPRA0) と書き込みアドレス (A3 ~ A0) の 2 種類のアドレス ポートがあります。この 2 種類のアドレス ポートは非同期です。読み出しアドレスによって出力ピン (DPO) に出力される値が指定され、書き込みアドレスによって書き込みを行う位置が指定されます。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。

WE が High の場合、WCLK が Low から High に切り替わるときに、データ入力 (D) の値が 4 ビットの書き込みアドレスで選択されたワードにロードされます。書き込みを正しく行うには、WCLK が Low から High に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ High ですが、インバータを使用してアクティブ Low にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

SPO 出力には、A3 ~ A0 で指定されたメモリ セルの値が出力されます。DPO 出力には、DPRA3 ~ DPRA0 で指定されたメモリ セルの値が出力されます。

**メモ：** 書き込み処理は、読み出しアドレス ポートのアドレスには影響されません。

INIT 属性を使用すると、RAM を直接初期化できます。値は、INIT=ABAC のように、16 進数で指定してください。INIT 属性を指定しない場合は、RAM は 0 に初期化されます。

### 論理表

モード選択を次の論理表に示します。

入力			出力	
WE (モード)	WCLK	D	SPO	DPO
0 (読み出し)	X	X	data_a	data_d
1 (読み出し)	0	X	data_a	data_d
1 (読み出し)	1	X	data_a	data_d
1 (書き込み)	↑	D	D	data_d
1 (読み出し)	↓	X	data_a	data_d
data_a = A3 ~ A0 で指定されたワード				
data_d = DPRA3 ~ DPRA0 で指定されたワード				

## デザインの入力方法

インスタンシエーション	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	RAM、レジスタ、LUT の初期値を指定

## VHDL 記述 (インスタンシエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM16X1D: 16 x 1 positive edge write, asynchronous read dual-port distributed RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2
```

```
RAM16X1D_inst : RAM16X1D
generic map (
  INIT => X"0000")
port map (
  DPO => DPO,      -- Read-only 1-bit data output for DPRA
  SPO => SPO,      -- R/W 1-bit data output for A0-A3
  A0 => A0,        -- R/W address[0] input bit
  A1 => A1,        -- R/W address[1] input bit
  A2 => A2,        -- R/W address[2] input bit
  A3 => A3,        -- R/W address[3] input bit
  D => D,          -- Write 1-bit data input
  DPRA0 => DPRA0,  -- Read-only address[0] input bit
  DPRA1 => DPRA1,  -- Read-only address[1] input bit
  DPRA2 => DPRA2,  -- Read-only address[2] input bit
  DPRA3 => DPRA3,  -- Read-only address[3] input bit
  WCLK => WCLK,    -- Write clock input
  WE => WE         -- Write enable input
);

-- End of RAM16X1D_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM16X1D: 16 x 1 positive edge write, asynchronous read dual-port distributed RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM16X1D_inst : RAM16X1D
generic map (
  INIT => X"0000")
port map (
  DPO => DPO,      -- Read-only 1-bit data output for DPRA
  SPO => SPO,      -- R/W 1-bit data output for A0-A3
  A0 => A0,         -- R/W address[0] input bit
  A1 => A1,         -- R/W address[1] input bit
  A2 => A2,         -- R/W address[2] input bit
  A3 => A3,         -- R/W address[3] input bit
  D => D,           -- Write 1-bit data input
  DPRA0 => DPRA0,  -- Read-only address[0] input bit
  DPRA1 => DPRA1,  -- Read-only address[1] input bit
  DPRA2 => DPRA2,  -- Read-only address[2] input bit
  DPRA3 => DPRA3,  -- Read-only address[3] input bit
  WCLK => WCLK,    -- Write clock input
  WE => WE         -- Write enable input
);

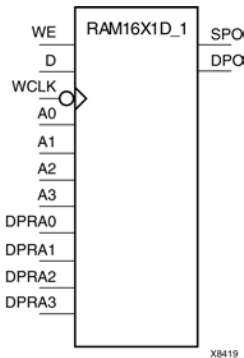
-- End of RAM16X1D_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM16X1D\_1

: 16-Deep by 1-Wide Static Dual Port Synchronous RAM with Negative-Edge Clock



### 概要

このエレメントは、クロックのネガティブ エッジで動作する 16 ワード X 1 ビットのデュアル ポート SRAM で、同期書き込み機能を備えています。デバイスには、読み出しアドレス (DPRA3 ~ DPRA0) と書き込みアドレス (A3 ~ A0) の独立した 2 種類のアドレス ポートがあります。この 2 種類のアドレス ポートは非同期です。読み出しアドレスによって出力ピン (DPO) に出力される値が指定され、書き込みアドレスによって書き込みを行う位置が指定されます。

ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High の場合、WCLK が High から Low に切り替わるときに、4 ビットの書き込みアドレスで選択されているワードにデータ入力 (D) の値がロードされます。書き込みを正しく行うには、WCLK が High から Low に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ High ですが、インバータを使用してアクティブ Low にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

INIT 属性を使用すると、コンフィギュレーション中に RAM16X1D\_1 を初期化できます。

SPO 出力には、A3 ~ A0 で指定されたメモリ セルの値が出力されます。DPO 出力には、DPRA3 ~ DPRA0 で指定されたメモリ セルの値が出力されます。

**メモ：** 書き込み処理は、読み出しアドレス ポートのアドレスには影響されません。

### 論理表

モード選択を次の論理表に示します。

入力			出力	
WE (モード)	WCLK	D	SPO	DPO
0 (読み出し)	X	X	data_a	data_d
1 (読み出し)	0	X	data_a	data_d
1 (読み出し)	1	X	data_a	data_d
1 (書き込み)	↓	D	D	data_d
1 (読み出し)	↑	X	data_a	data_d
data_a = A3 ~ A0 で指定されたワード				
data_d = DPRA3 ~ DPRA0 で指定されたワード				

## ポートの説明

ポート名	方向	幅	機能
DPO	出力	1	読み出し専用 1 ビット データ出力
SPO	出力	1	読み出し/書き込み 1 ビット データ出力
A0	入力	1	読み出し/書き込み address[0] 入力
A1	入力	1	読み出し/書き込み address[1] 入力
A2	入力	1	読み出し/書き込み address[2] 入力
A3	入力	1	読み出し/書き込み address[3] 入力
D	入力	1	書き込み 1 ビット データ入力
DPRA0	入力	1	読み出し専用 address[0] 入力
DPRA1	入力	1	読み出し専用 address[1] 入力
DPRA2	入力	1	読み出し専用 address[2] 入力
DPRA3	入力	1	読み出し専用 address[3] 入力
WCLK	入力	1	書き込みクロック入力
WE	入力	1	書き込みイネーブル入力

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	RAM、レジスタ、LUT の初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM16X1D_1: 16 x 1 negative edge write, asynchronous read dual-port distributed RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM16X1D_1_inst : RAM16X1D_1
generic map (
    INIT => X"0000")
port map (
    DPO => DPO,      -- Read-only 1-bit data output for DPRA
    SPO => SPO,      -- R/W 1-bit data output for A0-A3
    A0 => A0,         -- R/W address[0] input bit
    A1 => A1,         -- R/W address[1] input bit
    A2 => A2,         -- R/W address[2] input bit
    A3 => A3,         -- R/W address[3] input bit
    D => D,           -- Write 1-bit data input
    DPRA0 => DPRA0,  -- Read-only address[0] input bit
    DPRA1 => DPRA1,  -- Read-only address[1] input bit
    DPRA2 => DPRA2,  -- Read-only address[2] input bit
    DPRA3 => DPRA3,  -- Read-only address[3] input bit
    WCLK => WCLK,     -- Write clock input
    WE => WE          -- Write enable input
);

-- End of RAM16X1D_1_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM16X1D_1: 16 x 1 negative edge write, asynchronous read dual-port distributed RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM16X1D_1_inst : RAM16X1D_1
generic map (
    INIT => X"0000")
port map (
    DPO => DPO,      -- Read-only 1-bit data output for DPRA
    SPO => SPO,      -- R/W 1-bit data output for A0-A3
    A0 => A0,         -- R/W address[0] input bit
    A1 => A1,         -- R/W address[1] input bit
    A2 => A2,         -- R/W address[2] input bit
    A3 => A3,         -- R/W address[3] input bit
    D => D,           -- Write 1-bit data input
    DPRA0 => DPRA0,  -- Read-only address[0] input bit
    DPRA1 => DPRA1,  -- Read-only address[1] input bit
    DPRA2 => DPRA2,  -- Read-only address[2] input bit
    DPRA3 => DPRA3,  -- Read-only address[3] input bit
    WCLK => WCLK,     -- Write clock input
    WE => WE          -- Write enable input
);

-- End of RAM16X1D_1_inst instantiation
```

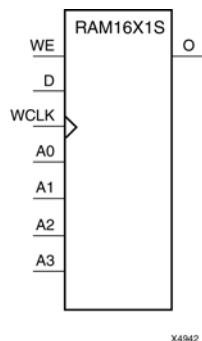
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリ データシート](#)



## RAM16X1S

: 16-Deep by 1-Wide Static Synchronous RAM



### 概要

このエレメントは 16 ワード X 1 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High になると、WCLK が Low から High に切り替わるときに、データ入力 (D) の値が 4 ビットのアドレス (A3 ~ A0) で選択されたワードにロードされます。WCLK はデフォルトではアクティブ High ですが、インバータを使用してアクティブ Low にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

出力ピン (O) に出力される値は、アドレス ピンで指定された RAM 内の位置に格納されている値です。INIT 属性を使用すると、コンフィギュレーション中に RAM16X1S を初期化できます。

### 論理表

入力			出力
WE (モード)	WCLK	D	O
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↑	D	D
1 (読み出し)	↓	X	データ
データ = A3 ~ A0 で指定されたワード			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	RAM の初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM16X1S: 16 x 1 posedge write distributed (LUT) RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM16X1S_inst : RAM16X1S
generic map (
  INIT => X"0000")
port map (
  O => O,          -- RAM output
  A0 => A0,         -- RAM address[0] input
  A1 => A1,         -- RAM address[1] input
  A2 => A2,         -- RAM address[2] input
  A3 => A3,         -- RAM address[3] input
  D => D,           -- RAM data input
  WCLK => WCLK,     -- Write clock input
  WE => WE          -- Write enable input
);

-- End of RAM16X1S_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM16X1S: 16 x 1 posedge write distributed (LUT) RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM16X1S_inst : RAM16X1S
generic map (
  INIT => X"0000")
port map (
  O => O,          -- RAM output
  A0 => A0,         -- RAM address[0] input
  A1 => A1,         -- RAM address[1] input
  A2 => A2,         -- RAM address[2] input
  A3 => A3,         -- RAM address[3] input
  D => D,           -- RAM data input
  WCLK => WCLK,     -- Write clock input
  WE => WE          -- Write enable input
);

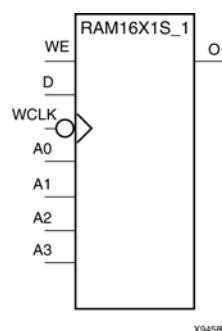
-- End of RAM16X1S_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM16X1S\_1

: 16-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock



### 概要

このエレメントはクロックの立ち下がりエッジで動作する 16 ワード X 1 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High の場合、WCLK が High から Low に切り替わるときに、データ入力 (D) の値が 4 ビットのアドレス (A3 ~ A0) で選択されたワードにロードされます。書き込みを正しく行うには、WCLK が High から Low に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ Low ですが、インバータを使用してアクティブ High にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

出力ピン (O) に出力される値は、アドレス ピンで指定された RAM 内の位置に格納されている値です。

INIT 属性を使用すると、コンフィギュレーション中にこのエレメントを初期化できます。

### 論理表

入力			出力
WE (モード)	WCLK	D	O
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↓	D	D
1 (読み出し)	↑	X	データ
データ = A3 ~ A0 で指定されたワード			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	RAM の初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM16X1S_1: 16 x 1 negedge write distributed (LUT) RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM16X1S_1_inst : RAM16X1S_1
generic map (
  INIT => X"0000")
port map (
  O => O,          -- RAM output
  A0 => A0,         -- RAM address[0] input
  A1 => A1,         -- RAM address[1] input
  A2 => A2,         -- RAM address[2] input
  A3 => A3,         -- RAM address[3] input
  D => D,          -- RAM data input
  WCLK => WCLK,     -- Write clock input
  WE => WE         -- Write enable input
);

-- End of RAM16X1S_1_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM16X1S_1: 16 x 1 negedge write distributed (LUT) RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM16X1S_1_inst : RAM16X1S_1
generic map (
  INIT => X"0000")
port map (
  O => O,          -- RAM output
  A0 => A0,         -- RAM address[0] input
  A1 => A1,         -- RAM address[1] input
  A2 => A2,         -- RAM address[2] input
  A3 => A3,         -- RAM address[3] input
  D => D,          -- RAM data input
  WCLK => WCLK,     -- Write clock input
  WE => WE         -- Write enable input
);

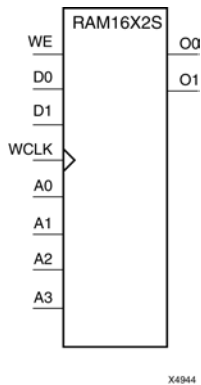
-- End of RAM16X1S_1_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM16X2S

: 16-Deep by 2-Wide Static Synchronous RAM



### 概要

このエレメントは 16 ワード X 2 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High の場合、WCLK が Low から High に切り替わる時に、入力 (D1 ~ D0) の値が 4 ビットのアドレス (A3 ~ A0) で選択されたワードにロードされます。書き込みを正しく行うには、WCLK が Low から High に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ High ですが、インバータを使用してアクティブ Low にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

出力ピン (O1 ~ O0) に出力される値は、アドレスピンで指定された RAM 内の位置に格納されている値です。

INIT\_xx 属性を使用すると、RAM の初期値を指定できます。INIT\_00 は出力 (O0) に対応する RAM のセルを初期化し、INIT\_01 は出力 (O1) に対応するセルを初期化します。たとえば、RAM16X2S インスタンスは、INIT\_00 および INIT\_01 にそれぞれ 4 個の 16 進数値を指定して初期化します。RAM16X8S インスタンスは、INIT\_00 ~ INIT\_07 の 8 個の属性にそれぞれ 4 個の 16 進数値を指定して初期化します。RAM64X2S インスタンスは、INIT\_00 および INIT\_01 にそれぞれ 16 個の 16 進数値を指定して初期化します。

Virtex-4 デバイス以外では、このエレメントの初期値を直接指定することはできません。

### 論理表

入力			出力
WE (モード)	WCLK	D1 ~ D0	O1 ~ O0
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↑	D1 ~ D0	D1 ~ D0
1 (読み出し)	↓	X	データ
データ = A3 ~ A0 で指定されたワード			

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_01	16 進数	16 ビット値	すべてゼロ	RAM、レジスタ、LUT の初期値を指定

## VHDL 記述（インスタンス化）

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM16X2S: 16 x 2 posedge write distributed (LUT) RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM16X2S_inst : RAM16X2S
generic map (
  INIT_00 => X"0000", -- INIT for bit 0 of RAM
  INIT_01 => X"0000") -- INIT for bit 1 of RAM
port map (
  O0 => O0,      -- RAM data[0] output
  O1 => O1,      -- RAM data[1] output
  A0 => A0,      -- RAM address[0] input
  A1 => A1,      -- RAM address[1] input
  A2 => A2,      -- RAM address[2] input
  A3 => A3,      -- RAM address[3] input
  D0 => D0,      -- RAM data[0] input
  D1 => D1,      -- RAM data[1] input
  WCLK => WCLK,  -- Write clock input
  WE => WE       -- Write enable input
);

-- End of RAM16X2S_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM16X2S: 16 x 2 posedge write distributed (LUT) RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM16X2S_inst : RAM16X2S
generic map (
    INIT_00 => X"0000", -- INIT for bit 0 of RAM
    INIT_01 => X"0000") -- INIT for bit 1 of RAM
port map (
    O0 => O0,          -- RAM data[0] output
    O1 => O1,          -- RAM data[1] output
    A0 => A0,          -- RAM address[0] input
    A1 => A1,          -- RAM address[1] input
    A2 => A2,          -- RAM address[2] input
    A3 => A3,          -- RAM address[3] input
    D0 => D0,          -- RAM data[0] input
    D1 => D1,          -- RAM data[1] input
    WCLK => WCLK,      -- Write clock input
    WE => WE           -- Write enable input
);

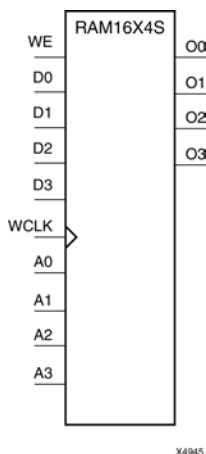
-- End of RAM16X2S_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM16X4S

: 16-Deep by 4-Wide Static Synchronous RAM



### 概要

このエレメントは 16 ワード X 4 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High になると、WCLK が Low から High に切り替わるときに、入力 (D3 ~ D0) の値が 4 ビットのアドレス (A3 ~ A0) で選択されたワードにロードされます。書き込みを正しく行うには、WCLK が Low から High に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ High ですが、インバータを使用してアクティブ Low にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

出力ピン (O3 ~ O0) に出力される値は、アドレスピンで指定された RAM 内の位置に格納されている値です。

### 論理表

入力			出力
WE (モード)	WCLK	D3:D0	O3:O0
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↑	D3:D0	D3:D0
1 (読み出し)	↓	X	データ
データ = A3 ~ A0 で指定されたワード			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可



## 使用可能な属性

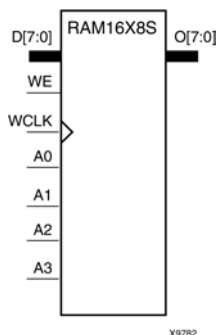
属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_03	16 進数	16 ビット値	すべてゼロ	RAM の初期値を指定

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM16X8S

: 16-Deep by 8-Wide Static Synchronous RAM



### 概要

このエレメントは 16 ワード X 8 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High になると、WCLK が Low から High に切り替わるときに、入力 (D7 ~ D0) の値が 4 ビットのアドレス (A3 ~ A0) で選択されたワードにロードされます。書き込みを正しく行うには、WCLK が Low から High に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ High ですが、インバータを使用してアクティブ Low にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

出力ピン (O7 ~ O0) に出力される値は、アドレス ピンで指定された RAM 内の位置に格納されている値です。

### 論理表

入力			出力
WE (モード)	WCLK	D7:D0	O7:O0
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↑	D7:D0	D7:D0
1 (読み出し)	↓	X	データ
データ = A3 ~ A0 で指定されたワード			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### 使用可能な属性

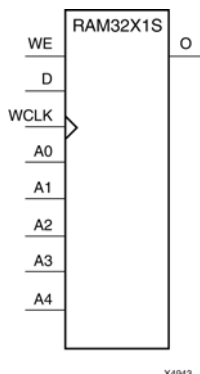
属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_07	16 進数	16 ビット値	すべてゼロ	RAM、レジスタ、LUT の初期値を指定

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM32X1S

: 32-Deep by 1-Wide Static Synchronous RAM



### 概要

このデザイン エLEMENTは 32 ワード X 1 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High になると、WCLK が Low から High に切り替わるときに、データ入力 (D) の値が 5 ビットのアドレス (A4 ~ A0) で選択されたワードにロードされます。書き込みを正しく行うには、WCLK が Low から High に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ High ですが、インバータを使用してアクティブ Low にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

出力ピン (O) に出力される値は、アドレス ピンで指定された RAM 内の位置に格納されている値です。INIT 属性を使用すると、コンフィギュレーション中に RAM32X1S を初期化できます。

### 論理表

入力			出力
WE (モード)	WCLK	D	O
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↓	D	D
1 (読み出し)	↑	X	データ

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	32 ビット値	すべてゼロ	RAM の初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM32X1S: 32 x 1 posedge write distributed (LUT) RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM32X1S_inst : RAM32X1S
generic map (
  INIT => X"00000000")
port map (
  O => O,          -- RAM output
  A0 => A0,         -- RAM address[0] input
  A1 => A1,         -- RAM address[1] input
  A2 => A2,         -- RAM address[2] input
  A3 => A3,         -- RAM address[3] input
  A4 => A4,         -- RAM address[4] input
  D => D,          -- RAM data input
  WCLK => WCLK,     -- Write clock input
  WE => WE         -- Write enable input
);

-- End of RAM32X1S_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM32X1S: 32 x 1 posedge write distributed (LUT) RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM32X1S_inst : RAM32X1S
generic map (
  INIT => X"00000000")
port map (
  O => O,          -- RAM output
  A0 => A0,         -- RAM address[0] input
  A1 => A1,         -- RAM address[1] input
  A2 => A2,         -- RAM address[2] input
  A3 => A3,         -- RAM address[3] input
  A4 => A4,         -- RAM address[4] input
  D => D,          -- RAM data input
  WCLK => WCLK,     -- Write clock input
  WE => WE         -- Write enable input
);

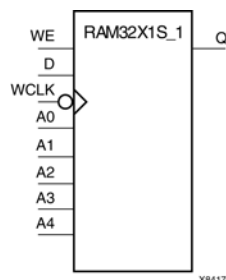
-- End of RAM32X1S_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM32X1S\_1

: 32-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock



### 概要

このデザイン エLEMENTは 32 ワード X 1 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High になると、WCLK が High から Low に切り替わるときに、データ入力 (D) の値が 5 ビットのアドレス (A4 ~ A0) で選択されたワードにロードされます。書き込みを正しく行うには、WCLK が High から Low に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ Low ですが、インバータを使用してアクティブ High にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

出力ピン (O) に出力される値は、アドレス ピンで指定された RAM 内の位置に格納されている値です。INIT 属性を使用すると、コンフィギュレーション中に RAM32X1S\_1 を初期化できます。

### 論理表

入力			出力
WE (モード)	WCLK	D	O
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↓	D	D
1 (読み出し)	↑	X	データ
データ = A4 ~ A0 で指定されたワード			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	32 ビット値	0	RAM、レジスタ、LUT の初期値を指定

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM32X1S_1: 32 x 1 negedge write distributed (LUT) RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM32X1S_1_inst : RAM32X1S_1
generic map (
  INIT => X"00000000")
port map (
  O => O,          -- RAM output
  A0 => A0,         -- RAM address[0] input
  A1 => A1,         -- RAM address[1] input
  A2 => A2,         -- RAM address[2] input
  A3 => A3,         -- RAM address[3] input
  A4 => A4,         -- RAM address[4] input
  D => D,          -- RAM data input
  WCLK => WCLK,     -- Write clock input
  WE => WE         -- Write enable input
);

-- End of RAM32X1S_1_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM32X1S_1: 32 x 1 negedge write distributed (LUT) RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM32X1S_1_inst : RAM32X1S_1
generic map (
  INIT => X"00000000")
port map (
  O => O,          -- RAM output
  A0 => A0,         -- RAM address[0] input
  A1 => A1,         -- RAM address[1] input
  A2 => A2,         -- RAM address[2] input
  A3 => A3,         -- RAM address[3] input
  A4 => A4,         -- RAM address[4] input
  D => D,          -- RAM data input
  WCLK => WCLK,     -- Write clock input
  WE => WE         -- Write enable input
);

-- End of RAM32X1S_1_inst instantiation
```

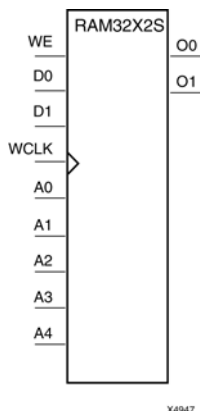


## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM32X2S

: 32-Deep by 2-Wide Static Synchronous RAM



### 概要

このデザイン エLEMENTは 32 ワード X 2 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High になると、WCLK が Low から High に切り替わるときに、入力 (D1 ~ D0) の値が 5 ビットのアドレス (A4 ~ A0) で選択されたワードにロードされます。書き込みを正しく行うには、WCLK が Low から High に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ High ですが、インバータを使用してアクティブ Low にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。出力ピン (O1 ~ O0) に出力される値は、アドレスピンで指定された RAM 内の位置に格納されている値です。

INIT\_00 および INIT\_01 属性を使用して RAM32X2S の初期値を指定できます。

### 論理表

入力			出力
WE (モード)	WCLK	D	O0 ~ O1
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↑	D1 ~ D0	D1 ~ D0
1 (読み出し)	↓	X	データ
データ = A4 ~ A0 で指定されたワード			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00	16 進数	32 ビット値	すべてゼロ	RAM のビット 0 の初期値を指定
INIT_01	16 進数	32 ビット値	すべてゼロ	RAM のビット 1 の初期値を指定

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM32X2S: 32 x 2 posedge write distributed (LUT) RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM32X2S_inst : RAM32X2S
generic map (
    INIT_00 => X"00000000", -- INIT for bit 0 of RAM
    INIT_01 => X"00000000") -- INIT for bit 1 of RAM
port map (
    O0 => O0,      -- RAM data[0] output
    O1 => O1,      -- RAM data[1] output
    A0 => A0,      -- RAM address[0] input
    A1 => A1,      -- RAM address[1] input
    A2 => A2,      -- RAM address[2] input
    A3 => A3,      -- RAM address[3] input
    A4 => A4,      -- RAM address[4] input
    D0 => D0,      -- RAM data[0] input
    D1 => D1,      -- RAM data[1] input
    WCLK => WCLK,  -- Write clock input
    WE => WE       -- Write enable input
);

-- End of RAM32X2S_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM32X2S: 32 x 2 posedge write distributed (LUT) RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM32X2S_inst : RAM32X2S
generic map (
  INIT_00 => X"00000000", -- INIT for bit 0 of RAM
  INIT_01 => X"00000000") -- INIT for bit 1 of RAM
port map (
  O0 => O0,      -- RAM data[0] output
  O1 => O1,      -- RAM data[1] output
  A0 => A0,      -- RAM address[0] input
  A1 => A1,      -- RAM address[1] input
  A2 => A2,      -- RAM address[2] input
  A3 => A3,      -- RAM address[3] input
  A4 => A4,      -- RAM address[4] input
  D0 => D0,      -- RAM data[0] input
  D1 => D1,      -- RAM data[1] input
  WCLK => WCLK,  -- Write clock input
  WE => WE       -- Write enable input
);

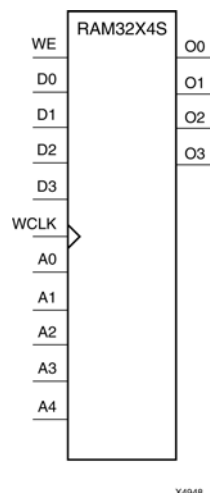
-- End of RAM32X2S_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM32X4S

: 32-Deep by 4-Wide Static Synchronous RAM



### 概要

このデザイン エLEMENTは 32 ワード X 4 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High になると、WCLK が Low から High に切り替わるときに、入力 (D3 ~ D0) の値が 5 ビットのアドレス (A4 ~ A0) で選択されたワードにロードされます。書き込みを正しく行うには、WCLK が Low から High に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ High ですが、インバータを使用してアクティブ Low にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

出力ピン (O3 ~ O0) に出力される値は、アドレスピンで指定された RAM 内の位置に格納されている値です。

### 論理表

入力			出力
WE	WCLK	D3 ~ D0	O3 ~ O0
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↑	D3:D0	D3:D0
1 (読み出し)	↓	X	データ
データ = A4 ~ A0 で指定されたワード			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

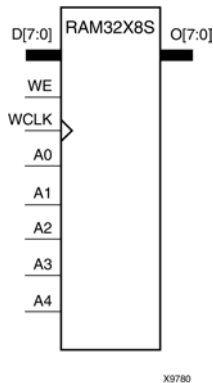
属性	タイプ	値	デフォルト	説明
INIT_00	16 進数	32 ビット値	すべてゼロ	RAM のビット 0 の初期値を指定
INIT_01	16 進数	32 ビット値	すべてゼロ	RAM のビット 1 の初期値を指定
INIT_02	16 進数	32 ビット値	すべてゼロ	RAM のビット 2 の初期値を指定
INIT_03	16 進数	32 ビット値	すべてゼロ	RAM のビット 3 の初期値を指定

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM32X8S

: 32-Deep by 8-Wide Static Synchronous RAM



### 概要

このデザイン エLEMENTは 32 ワード X 8 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High になると、WCLK が Low から High に切り替わるときに、入力 (D7 ~ D0) の値が 5 ビットのアドレス (A4 ~ A0) で選択されたワードにロードされます。書き込みを正しく行うには、WCLK が Low から High に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ High ですが、インバータを使用してアクティブ Low にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

出力ピン (O7 ~ O0) に出力される値は、アドレス ピンで指定された RAM 内の位置に格納されている値です。

### 論理表

入力			出力
WE (モード)	WCLK	D7:D0	O7:O0
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↑	D7:D0	D7:D0
1 (読み出し)	↓	X	データ
データ = A4 ~ A0 で指定されたワード			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00	16 進数	32 ビット値	すべてゼロ	RAM のビット 0 の初期値を指定
INIT_01	16 進数	32 ビット値	すべてゼロ	RAM のビット 1 の初期値を指定
INIT_02	16 進数	32 ビット値	すべてゼロ	RAM のビット 2 の初期値を指定
INIT_03	16 進数	32 ビット値	すべてゼロ	RAM のビット 3 の初期値を指定
INIT_04	16 進数	32 ビット値	すべてゼロ	RAM のビット 4 の初期値を指定
INIT_05	16 進数	32 ビット値	すべてゼロ	RAM のビット 5 の初期値を指定
INIT_06	16 進数	32 ビット値	すべてゼロ	RAM のビット 6 の初期値を指定
INIT_07	16 進数	32 ビット値	すべてゼロ	RAM のビット 7 の初期値を指定

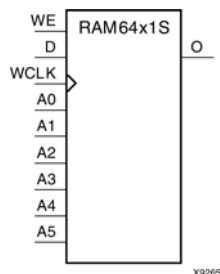
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## RAM64X1S

: 64-Deep by 1-Wide Static Synchronous RAM



### 概要

このデザイン エLEMENTは 64 ワード X 1 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High になると、WCLK が Low から High に切り替わるときに、データ入力 (D) の値が 6 ビットのアドレス (A5 ～ A0) で選択されたワードにロードされます。WCLK はデフォルトではアクティブ High ですが、インバータを使用してアクティブ Low にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

出力ピン (O) に出力される値は、アドレス ピンで指定された RAM 内の位置に格納されている値です。

INIT 属性を使用すると、コンフィギュレーション中にこのELEMENTを初期化できます。

### 論理表

モード選択を次の論理表に示します。

入力			出力
WE (モード)	WCLK	D	O
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↑	D	D
1 (読み出し)	↓	X	データ
データ = A5 ～ A0 で指定されたワード			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	64 ビット値	すべてゼロ	ROM、RAM、レジスタ、LUT の初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM64X1S: 64 x 1 positive edge write, asynchronous read single-port distributed RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM64X1S_inst : RAM64X1S
generic map (
  INIT => X"0000000000000000")
port map (
  O => O,           -- 1-bit data output
  A0 => A0,          -- Address[0] input bit
  A1 => A1,          -- Address[1] input bit
  A2 => A2,          -- Address[2] input bit
  A3 => A3,          -- Address[3] input bit
  A4 => A4,          -- Address[4] input bit
  A5 => A5,          -- Address[5] input bit
  D => D,            -- 1-bit data input
  WCLK => WCLK,      -- Write clock input
  WE => WE           -- Write enable input
);

-- End of RAM64X1S_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM64X1S: 64 x 1 positive edge write, asynchronous read single-port distributed RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM64X1S_inst : RAM64X1S
generic map (
  INIT => X"0000000000000000")
port map (
  O => O,           -- 1-bit data output
  A0 => A0,          -- Address[0] input bit
  A1 => A1,          -- Address[1] input bit
  A2 => A2,          -- Address[2] input bit
  A3 => A3,          -- Address[3] input bit
  A4 => A4,          -- Address[4] input bit
  A5 => A5,          -- Address[5] input bit
  D => D,            -- 1-bit data input
  WCLK => WCLK,      -- Write clock input
  WE => WE           -- Write enable input
);

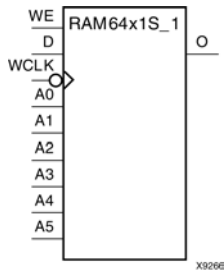
-- End of RAM64X1S_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM64X1S\_1

: 64-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock



### 概要

このデザイン エLEMENTは 64 ワード X 1 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High になると、WCLK が High から Low に切り替わるときに、データ入力 (D) の値が 6 ビットのアドレス (A5 ~ A0) で選択されたワードにロードされます。書き込みを正しく行うには、WCLK が High から Low に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ Low ですが、インバータを使用してアクティブ High にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

出力ピン (O) に出力される値は、アドレス ピンで指定された RAM 内の位置に格納されている値です。

INIT 属性を使用すると、コンフィギュレーション中にこのELEMENTを初期化できます。

### 論理表

入力			出力
WE (モード)	WCLK	D	O
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↓	D	D
1 (読み出し)	↑	X	データ
データ = A5 ~ A0 で指定されたワード			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	64 ビット値	すべてゼロ	ROM、RAM、レジスタ、LUT の初期値を指定

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM64X1S_1: 64 x 1 negative edge write, asynchronous read single-port distributed RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM64X1S_1_inst : RAM64X1S_1
generic map (
  INIT => X"0000000000000000")
port map (
  O => O,          -- 1-bit data output
  A0 => A0,         -- Address[0] input bit
  A1 => A1,         -- Address[1] input bit
  A2 => A2,         -- Address[2] input bit
  A3 => A3,         -- Address[3] input bit
  A4 => A4,         -- Address[4] input bit
  A5 => A5,         -- Address[5] input bit
  D => D,          -- 1-bit data input
  WCLK => WCLK,     -- Write clock input
  WE => WE         -- Write enable input
);

-- End of RAM64X1S_1_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAM64X1S_1: 64 x 1 negative edge write, asynchronous read single-port distributed RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAM64X1S_1_inst : RAM64X1S_1
generic map (
  INIT => X"0000000000000000")
port map (
  O => O,          -- 1-bit data output
  A0 => A0,         -- Address[0] input bit
  A1 => A1,         -- Address[1] input bit
  A2 => A2,         -- Address[2] input bit
  A3 => A3,         -- Address[3] input bit
  A4 => A4,         -- Address[4] input bit
  A5 => A5,         -- Address[5] input bit
  D => D,          -- 1-bit data input
  WCLK => WCLK,     -- Write clock input
  WE => WE         -- Write enable input
);

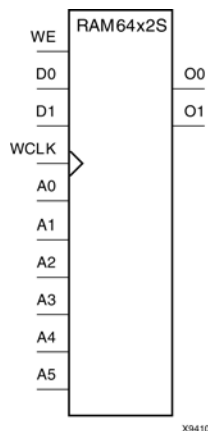
-- End of RAM64X1S_1_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAM64X2S

: 64-Deep by 2-Wide Static Synchronous RAM



### 概要

このデザイン エLEMENTは 64 ワード X 2 ビットの SRAM で、同期書き込み機能を備えています。ライト イネーブル (WE) が Low の場合、ライト クロック (WCLK) の遷移は無視され、RAM に格納されている値は変化しません。WE が High になると、WCLK が Low から High に切り替わるときに、入力 (D1 ~ D0) の値が 6 ビットのアドレス (A5 ~ A0) で選択されたワードにロードされます。書き込みを正しく行うには、WCLK が Low から High に切り替わる前に、書き込みアドレスとデータ入力の値を安定させる必要があります。WCLK はデフォルトではアクティブ High ですが、インバータを使用してアクティブ Low にすることもできます。WCLK の入力ネットに配置されたインバータは、RAM ブロック内に組み込まれます。

出力ピン (O1 ~ O0) に出力される値は、アドレス ピンで指定された RAM 内の位置に格納されている値です。INIT\_00 および INIT\_01 属性を使用してこのデザイン エLEMENTの初期値を指定できます。

### 論理表

入力			出力
WE (モード)	WCLK	D0:D1	O0:O1
0 (読み出し)	X	X	データ
1 (読み出し)	0	X	データ
1 (読み出し)	1	X	データ
1 (書き込み)	↑	D1 ~ D0	D1 ~ D0
1 (読み出し)	↓	X	データ
データ = A5 ~ A0 で指定されたワード			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

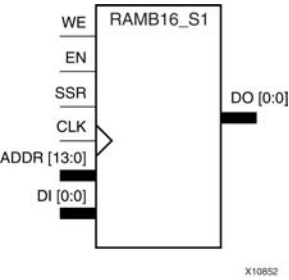
属性	タイプ	値	デフォルト	説明
INIT_00	16 進数	64 ビット値	すべてゼロ	RAM、レジスタ、LUT の初期値を指定
INIT_01	16 進数	64 ビット値	すべてゼロ	RAM、レジスタ、LUT の初期値を指定

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリ データシート](#)

# RAMB16\_S1

: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 1-bit Port



## 概要

このデザイン エLEMENTは、同期書き込み機能を備えた専用 RAM ブロックです。ブロック RAM のポートには 16384 ビットのデータ メモリがあります。セルの構成を、次の表に示します。

データ セル		パリティ セル				
ワード数	幅	ワード数	幅	アドレス バス	データ バス	パリティ バス
16384	1	–	–	(13:0)	(0:0)	–

イネーブル ピン EN は、ポートの読み出し、書き込み、リセットを制御します。EN が Low の場合、データは書き込まれず、出力 (DO および DOP) は変化しません。EN とリセット (SSR) が High の場合、クロック (CLK) が Low から High に切り替わるときに DO および DOP が SRVAL にセットされます。ライト イネーブル (WE) も High の場合は、DI および DIP の値が RAM に書き込まれます。EN が High で SSR と WE が Low の場合、クロックが Low から High に切り替わるときに、RAM アドレス (ADDR) に格納されているデータが読み出されます。SSR が Low で EN と WE が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDR) で選択されているワードにデータ入力 (DI および DIP) の値が読み込まれます。データ出力 (DO および DOP) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

前述の説明では、制御ピン (EN、WE、SSR、CLK) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。



## 論理表

入力								出力			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT	INIT	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR=グローバル セットリセット信号

INIT = データ メモリ用に INIT 属性で設定された値。デフォルトはすべて 0 です。

SRVAL = SRVAL 属性で設定された SSR のアサート後の値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE=NO\_CHANGE

<sup>2</sup> WRITE\_MODE=READ\_FIRST

<sup>3</sup> WRITE\_MODE=WRITE\_FIRST

## 初期化

## メモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16 の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## 出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。シングルポート RAMB16 の出力レジスタの初期化に使用する属性には、INIT と SRVAL の 2 種類があります。INIT 属性は、電源投入時の出力レジスタの値を指定します。SRVAL 属性は、SSR (セット/リセット) 入力をアサートしたときの値を指定します。

INIT および SRVAL 属性は、初期化値を、出力ポートの各ビットを 1 ビット含む 16 進数文字列として指定します。たとえば、ポート幅が 1 の RAMB16\_S1 の場合、出力レジスタは 1 ビットなので、INIT および SRVAL には 1 または 0 を指定します。ポート幅が 4 の RAMB16\_S4 の場合は、出力レジスタが 4 ビットなので、0 ~ F の 16 進数値を指定します。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT または SRVAL の値の上位ビットで指定します。

#### 書き込みモードの選択

WRITE\_MODE 属性は、RAMB16 メモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE は WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。WRITE\_MODE を READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。WRITE\_MODE を NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DO 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定
SRVAL	2 進数/ 16 進数	任意の値	すべてゼロ	SSR ピンをアサートすると、DO 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
WRITE_MODE	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DO ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S1: Single-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_inst : RAMB16_S1
generic map (
  INIT => X"0", -- Value of output RAM registers at startup
  SRVAL => X"0", -- Output value upon SSR assertion
  WRITE_MODE => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 4095
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 4096 to 8191
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 8192 to 12287
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 12288 to 16383
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000"
```

```

INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DO => DO,          -- 1-bit Data Output
  ADDR => ADDR,       -- 14-bit Address Input
  CLK => CLK,         -- Clock
  DI => DI,          -- 1-bit Data Input
  EN => EN,           -- RAM Enable Input
  SSR => SSR,         -- Synchronous Set/Reset Input
  WE => WE            -- Write Enable Input
);

-- End of RAMB16_S1_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S1: Single-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_inst : RAMB16_S1
generic map (
  INIT => X"0", -- Value of output RAM registers at startup
  SRVAL => X"0", -- Output value upon SSR assertion
  WRITE_MODE => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 4095
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 4096 to 8191
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",

```

```
INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 8192 to 12287
INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 12288 to 16383
INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DO => DO,      -- 1-bit Data Output
  ADDR => ADDR,   -- 14-bit Address Input
  CLK => CLK,     -- Clock
  DI => DI,       -- 1-bit Data Input
  EN => EN,       -- RAM Enable Input
  SSR => SSR,     -- Synchronous Set/Reset Input
  WE => WE        -- Write Enable Input
);

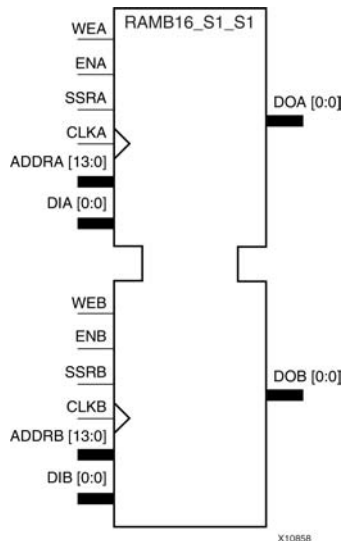
-- End of RAMB16_S1_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S1\_S1

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号

INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_A = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号

INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_B = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S1_S1	16384 x 1	—	(13:0)	(0:0)	—	16384 x 1	—	(13:0)	(0:0)	—

(a) ワード数 X 幅



各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるときに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるときに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるときに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるときに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マッピング

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																

パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリセルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロック SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリセルにアクセスできます。ただし、一方のポートがあるメモリセルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリセルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化 なし	X	変化 なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	変化 なし	変化 なし	変化 なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S1_S1: 16k x 1 Dual-Port RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_S1_inst : RAMB16_S1_S1
generic map (
  INIT_A => "0", -- Value of output RAM registers on Port A at startup
  INIT_B => "0", -- Value of output RAM registers on Port B at startup
  SRVAL_A => "0", -- Port A output value upon SSR assertion
  SRVAL_B => "0", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 4095
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 4096 to 8191
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 8192 to 12287
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 12288 to 16383
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
port map (
  DOA => DOA,      -- Port A 1-bit Data Output
  DOB => DOB,      -- Port B 1-bit Data Output
  ADDRA => ADDRA,   -- Port A 14-bit Address Input
  ADDR_B => ADDR_B, -- Port B 14-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,      -- Port A 1-bit Data Input
  DIB => DIB,      -- Port B 1-bit Data Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- Port B RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S1_S1_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S1_S1: 16k x 1 Dual-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_S1_inst : RAMB16_S1_S1
generic map (
  INIT_A => "0", -- Value of output RAM registers on Port A at startup
  INIT_B => "0", -- Value of output RAM registers on Port B at startup
  SRVAL_A => "0", -- Port A output value upon SSR assertion
  SRVAL_B => "0", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 4095
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 4096 to 8191
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

```

INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 8192 to 12287
INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 12288 to 16383
INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
port map (
  DOA => DOA,      -- Port A 1-bit Data Output
  DOB => DOB,      -- Port B 1-bit Data Output
  ADDRA => ADDRA,   -- Port A 14-bit Address Input
  ADDRb => ADDRb,   -- Port B 14-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,      -- Port A 1-bit Data Input
  DIB => DIB,      -- Port B 1-bit Data Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- PortB RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S1_S1_inst instantiation

```

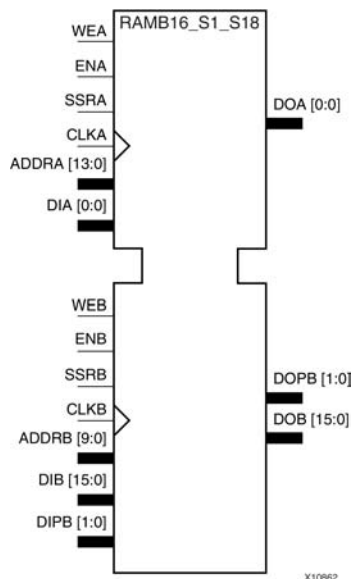
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## RAMB16\_S1\_S18

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 18-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号

INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_A = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、data <sup>3</sup>	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号  
INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。  
SRVAL\_B = レジスタの値  
addr = RAM アドレス  
RAM(addr) = アドレス ADDR の RAM の内容  
data = RAM の入力データ  
pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S1_S18	16384 x 1	—	(13:0)	(0:0)	—	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マッピング

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR\_port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリセルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロック SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリセルにアクセスできます。ただし、一方のポートがあるメモリセルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリセルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```



```
use UNISIM.vcomponents.all;

-- RAMB16_S1_S18: 16k/1k x 1/16 + 0/2 Parity bits Dual-Port RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_S18_inst : RAMB16_S1_S18
generic map (
  INIT_A => "0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"00000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => "0", -- Port A output value upon SSR assertion
  SRVAL_B => X"00000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 4095, Port B Address 0 to 255
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 4096 to 8191, Port B Address 256 to 511
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 8192 to 12287, Port B Address 512 to 767
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 12288 to 16383, Port B Address 768 to 1023
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Port B Address 0 to 255
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 256 to 511
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 512 to 767
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 768 to 1023
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 1-bit Data Output
  DOB => DOB,      -- Port B 16-bit Data Output
  DOPB => DOPB,    -- Port B 2-bit Parity Output
  ADDRA => ADDRA,  -- Port A 14-bit Address Input
  ADDRb => ADDRb,  -- Port B 10-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKB => CLKB,    -- Port B Clock
  DIA => DIA,      -- Port A 1-bit Data Input
  DIB => DIB,      -- Port B 16-bit Data Input
  DIPB => DIPB,    -- Port-B 2-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- PortB RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S1_S18_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S1_S18: 16k/1k x 1/16 + 0/2 Parity bits Dual-Port RAM
--               Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_S18_inst : RAMB16_S1_S18
generic map (
  INIT_A => "0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"00000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => "0", -- Port A output value upon SSR assertion
  SRVAL_B => X"00000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 4095, Port B Address 0 to 255
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

[illegible]

```
-- Port B Address 768 to 1023
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 1-bit Data Output
  DOB => DOB,      -- Port B 16-bit Data Output
  DOPB => DOPB,    -- Port B 2-bit Parity Output
  ADDRA => ADDRA,  -- Port A 14-bit Address Input
  ADDRb => ADDRb,  -- Port B 10-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKB => CLKB,    -- Port B Clock
  DIA => DIA,      -- Port A 1-bit Data Input
  DIB => DIB,      -- Port B 16-bit Data Input
  DIPB => DIPB,    -- Port-B 2-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- PortB RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

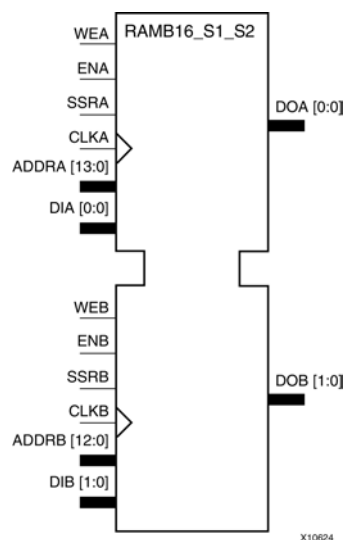
-- End of RAMB16_S1_S18_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S1\_S2

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 2-bit Ports



### 概要

このデザイン エレメントは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このエレメントのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号

INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_A = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、data <sup>3</sup>	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号  
INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。  
SRVAL\_B = レジスタの値  
addr = RAM アドレス  
RAM(addr) = アドレス ADDR の RAM の内容  
data = RAM の入力データ  
pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S1_S2	16384 x 1	—	(13:0)	(0:0)	—	8192 x 2	—	(12:0)	(1:0)	—

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マップ

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR\_port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																



## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<-----	3					2				1			0
2	1024	<-----	1									0			
4	512	<-----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan-3A の場合、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリセルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロックの SelectRAM は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリセルにアクセスできます。ただし、一方のポートがあるメモリセルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリセルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアル ポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化 なし	X	変化 なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	変化 なし	変化 なし	変化 なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット(0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット(0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S1_S2: 16k/8k x 1/2 Dual-Port RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_S2_inst : RAMB16_S1_S2
generic map (
  INIT_A => "0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"0", -- Value of output RAM registers on Port B at startup
  SRVAL_A => "0", -- Port A output value upon SSR assertion
  SRVAL_B => X"0", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 4095, Port B Address 0 to 2047
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 4096 to 8191, Port B Address 2048 to 4095
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  --Port A Address 8192 to 12287, Port B Address 4096 to 6143
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 12288 to 16383, Port B Address 6144 to 8091
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 1-bit Data Output
  DOB => DOB,      -- Port B 2-bit Data Output
  ADDRA => ADDRA,   -- Port A 14-bit Address Input
  ADDR_B => ADDR_B, -- Port B 13-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,      -- Port A 1-bit Data Input
  DIB => DIB,      -- Port B 2-bit Data Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- Port B RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S1_S2_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S1_S2: 16k/8k x 1/2 Dual-Port RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_S2_inst : RAMB16_S1_S2
generic map (
  INIT_A => "0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"0", -- Value of output RAM registers on Port B at startup
  SRVAL_A => "0", -- Port A output value upon SSR assertion
  SRVAL_B => X"0", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 4095, Port B Address 0 to 2047
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 4096 to 8191, Port B Address 2048 to 4095
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

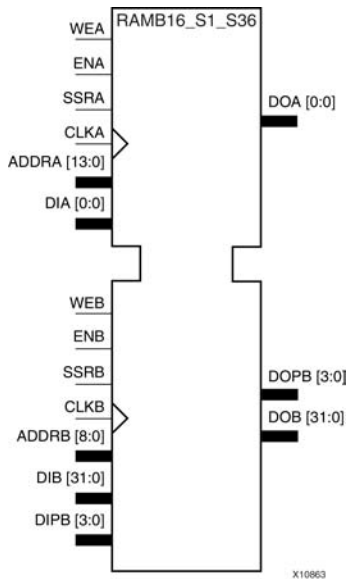
[illegible]

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S1\_S36

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 36-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。



## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号  
 INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。  
 SRVAL\_A = レジスタの値  
 addr = RAM アドレス  
 RAM(addr) = アドレス ADDR の RAM の内容  
 data = RAM の入力データ  
 pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号

INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_B = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレ ス バス	データ バス	パリティ バス
RAMB16_S1_S36	16384 x 1	—	(13:0)	(0:0)	—	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるときに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるときに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるときに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるときに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マッピング

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティ メモリ セルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<-----	3				2				1				0
2	1024	<-----	1								0				
4	512	<-----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリ セルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロック SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリ セルにアクセスできます。ただし、一方のポートがあるメモリ セルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリ セルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S1_S36: 16k/512 x 1/32 + 0/4 Parity bits Dual-Port RAM
--                Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_S36_inst : RAMB16_S1_S36
generic map (
  INIT_A => "0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"00000000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => "0", -- Port A output value upon SSR assertion
  SRVAL_B => X"00000000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 4095, Port B Address 0 to 127
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 4096 to 8191, Port B Address 128 to 255
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 8192 to 12287, Port B Address 256 to 383
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 12288 to 16383, Port B Address 384 to 512
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```



```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
--Port B Address 0 to 127
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 128 to 255
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 256 to 383
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 384 to 512
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 1-bit Data Output
  DOB => DOB,      -- Port B 32-bit Data Output
  DOPB => DOPB,    -- Port B 4-bit Parity Output
  ADDRA => ADDRA,   -- Port A 14-bit Address Input
  ADDRb => ADDRb,   -- Port B 9-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKb => CLKb,     -- Port B Clock
  DIA => DIA,      -- Port A 1-bit Data Input
  DIB => DIB,      -- Port B 32-bit Data Input
  DIPB => DIPB,    -- Port-B 4-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- PortB RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRb => SSRb,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S1_S36_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S1_S36: 16k/512 x 1/32 + 0/4 Parity bits Dual-Port RAM
--               Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_S36_inst : RAMB16_S1_S36
generic map (
  INIT_A => "0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"000000000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => "0", -- Port A output value upon SSR assertion
  SRVAL_B => X"000000000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 4095, Port B Address 0 to 127
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

Spartan-3E ライブラリ ガイド (HDL 用)  
UG617 (v12.2) 2010 年 7 月 23 日

```
-- Port B Address 384 to 512
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 1-bit Data Output
  DOB => DOB,      -- Port B 32-bit Data Output
  DOPB => DOPB,    -- Port B 4-bit Parity Output
  ADDRA => ADDRA,  -- Port A 14-bit Address Input
  ADDRb => ADDRb,  -- Port B 9-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKB => CLKB,    -- Port B Clock
  DIA => DIA,      -- Port A 1-bit Data Input
  DIB => DIB,      -- Port B 32-bit Data Input
  DIPB => DIPB,    -- Port-B 4-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- PortB RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

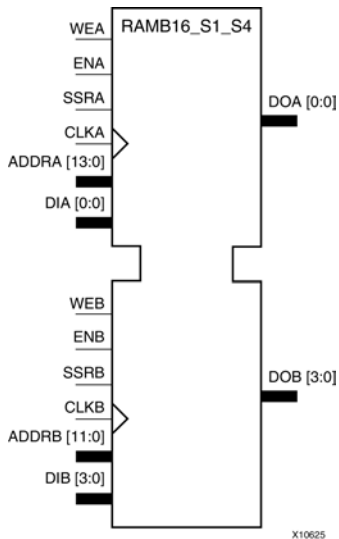
-- End of RAMB16_S1_S36_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S1\_S4

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 4-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号  
 INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。  
 SRVAL\_A = レジスタの値  
 addr = RAM アドレス  
 RAM(addr) = アドレス ADDR の RAM の内容  
 data = RAM の入力データ  
 pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号

INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_B = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S1_S4	16384 x 1	—	(13:0)	(0:0)	—	4096 x 4	—	(11:0)	(3:0)	—

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マップ

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																			
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0			
4	4096	<--	7				6				5				4				3				2				1				0					
8	2048	<--	3								2								1								0									
16	1024	<--	1																0																	
32	512	<--	0																																	

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリセルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロック SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリセルにアクセスできます。ただし、一方のポートがあるメモリセルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリセルに対して書き込みまたは読み出しを実行しないようにする必要があります。



次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、WARNING_ONLY、GENERATE_X_ONLY、NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、READ_FIRST、NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときのポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

属性	タイプ	値	デフォルト	説明
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_ FIRST	書き込みコマンドが DOA/DOB ポートで 実行されるときポートの動作を指定し ます。WRITE_FIRST に設定すると、値が ポートに書き込まれてから出力されます。 READ_FIRST に設定すると、新たに値が 書き込まれる前に、RAM の以前の値が 出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が 維持され、出力ポートの値が変化しま せん。RAM のポートから値を読み出さ ない場合は、このモードに設定すること をお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S1_S4: 16k/4k x 1/4 Dual-Port RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_S4_inst : RAMB16_S1_S4
generic map (
  INIT_A => "0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"0", -- Value of output RAM registers on Port B at startup
  SRVAL_A => "0", -- Port A output value upon SSR assertion
  SRVAL_B => X"0", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 4095, Port B Address 0 to 1023
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 4096 to 8191, Port B Address 1024 to 2047
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 8192 to 12287, Port B Address 2048 to 3071
INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 12288 to 16383, Port B Address 3072 to 4095
INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 1-bit Data Output
  DOB => DOB,      -- Port B 4-bit Data Output
  ADDRA => ADDRA,   -- Port A 14-bit Address Input
  ADDR_B => ADDR_B, -- Port B 12-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,      -- Port A 1-bit Data Input
  DIB => DIB,      -- Port B 4-bit Data Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- Port B RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S1_S4_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S1_S4: 16k/4k x 1/4 Dual-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_S4_inst : RAMB16_S1_S4
generic map (
  INIT_A => "0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"0", -- Value of output RAM registers on Port B at startup
  SRVAL_A => "0", -- Port A output value upon SSR assertion
  SRVAL_B => X"0", -- Port B output value upon SSR assertion

```

262

```
DOA => DOA,      -- Port A 1-bit Data Output
DOB => DOB,      -- Port B 4-bit Data Output
ADDRA => ADDRA,  -- Port A 14-bit Address Input
ADDRB => ADDRb,  -- Port B 12-bit Address Input
CLKA => CLKA,    -- Port A Clock
CLKB => CLKB,    -- Port B Clock
DIA => DIA,      -- Port A 1-bit Data Input
DIB => DIB,      -- Port B 4-bit Data Input
ENA => ENA,      -- Port A RAM Enable Input
ENB => ENB,      -- PortB RAM Enable Input
SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
WEA => WEA,      -- Port A Write Enable Input
WEB => WEB,      -- Port B Write Enable Input
);

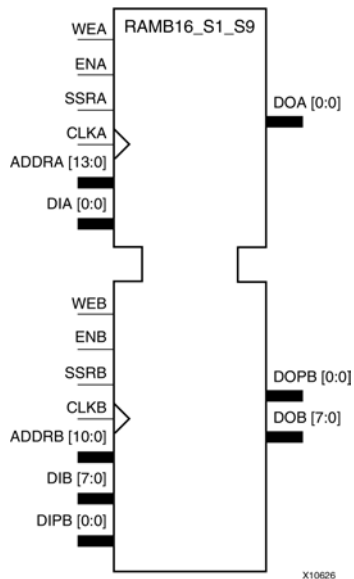
-- End of RAMB16_S1_S4_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S1\_S9

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 9-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。



## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号  
 INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。  
 SRVAL\_A = レジスタの値  
 addr = RAM アドレス  
 RAM(addr) = アドレス ADDR の RAM の内容  
 data = RAM の入力データ  
 pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号

INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_B = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S1_S9	16384 x 1	—	(13:0)	(0:0)	—	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マッピング

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A の場合、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力のアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力のアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリ セルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロックの SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリ セルにアクセスできます。ただし、一方のポートがあるメモリ セルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリ セルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S1_S9: 16k/2k x 1/8 + 0/1 Parity bit Dual-Port RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_S9_inst : RAMB16_S1_S9
generic map (
  INIT_A => "0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => "0", -- Port A output value upon SSR assertion
  SRVAL_B => X"000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 4095, Port B Address 0 to 511
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 4096 to 8191, Port B Address 512 to 1023
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 8192 to 12287, Port B Address 1024 to 1535
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 12288 to 16383, Port B Address 1536 to 2047
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```



```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Port B Address 0 to 511
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 512 to 1023
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 1024 to 1535
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 1535 to 2047
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 1-bit Data Output
  DOB => DOB,      -- Port B 8-bit Data Output
  DOPB => DOPB,    -- Port B 1-bit Parity Output
  ADDRA => ADDRA,  -- Port A 14-bit Address Input
  ADDRb => ADDRb,  -- Port B 11-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKb => CLKb,    -- Port B Clock
  DIA => DIA,      -- Port A 1-bit Data Input
  DIB => DIB,      -- Port B 8-bit Data Input
  DIPB => DIPB,    -- Port B 1-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- Port B RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRb => SSRb,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S1_S9_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S1_S9: 16k/2k x 1/8 + 0/1 Parity bit Dual-Port RAM
--              Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S1_S9_inst : RAMB16_S1_S9
generic map (
  INIT_A => "0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => "0", -- Port A output value upon SSR assertion
  SRVAL_B => X"000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 4095, Port B Address 0 to 511
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

Spartan-3E ライブラリ ガイド (HDL 用)  
UG617 (v12.2) 2010 年 7 月 23 日

```
-- Port B Address 1535 to 2047
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 1-bit Data Output
  DOB => DOB,      -- Port B 8-bit Data Output
  DOPB => DOPB,    -- Port B 1-bit Parity Output
  ADDRA => ADDRA,  -- Port A 14-bit Address Input
  ADDRb => ADDRb,  -- Port B 11-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKB => CLKB,    -- Port B Clock
  DIA => DIA,      -- Port A 1-bit Data Input
  DIB => DIB,      -- Port B 8-bit Data Input
  DIPB => DIPB,    -- Port-B 1-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- PortB RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

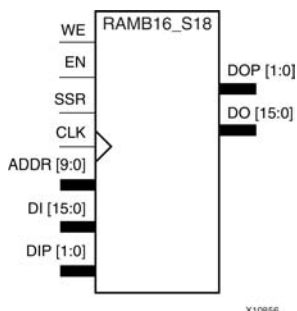
-- End of RAMB16_S1_S9_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S18

: 16K-bit Data + 2K-bit Parity Memory, Single-Port Synchronous Block RAM with 18-bit Port



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた専用 RAM ブロックです。ブロック RAM のポートには 16384 ビットのデータ メモリがあります。セルの構成を、次の表に示します。

データ セル		パリティ セル				
ワード数	幅	ワード数	幅	アドレス バス	データ バス	パリティ バス
1024	16	1024	2	(9:0)	(15:0)	(1:0)

イネーブル ピン EN は、ポートの読み出し、書き込み、リセットを制御します。EN が Low の場合、データは書き込まれず、出力 (DO および DOP) は変化しません。EN とリセット (SSR) が High の場合、クロック (CLK) が Low から High に切り替わるときに DO および DOP が SRVAL にセットされます。ライト イネーブル (WE) も High の場合は、DI および DIP の値が RAM に書き込まれます。EN が High で SSR と WE が Low の場合、クロックが Low から High に切り替わるときに、RAM アドレス (ADDR) に格納されているデータが読み出されます。SSR が Low で EN と WE が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDR) で選択されているワードにデータ入力 (DI および DIP) の値が読み込まれます。データ出力 (DO および DOP) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

前述の説明では、制御ピン (EN、WE、SSR、CLK) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

## 論理表

入力								出力			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT	INIT	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR=グローバル セットリセット信号

INIT = データ メモリ用に INIT 属性で設定された値。デフォルトはすべて 0 です。

SRVAL = SRVAL 属性で設定された SSR のアサート後の値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE=NO\_CHANGE

<sup>2</sup> WRITE\_MODE=READ\_FIRST

<sup>3</sup> WRITE\_MODE=WRITE\_FIRST

## 初期化

## メモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16 の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## 出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。シングルポート RAMB16 の出力レジスタの初期化に使用する属性には、INIT と SRVAL の 2 種類があります。INIT 属性は、電源投入時の出力レジスタの値を指定します。SRVAL 属性は、SSR (セット/リセット) 入力をアサートしたときの値を指定します。

INIT および SRVAL 属性は、初期化値を、出力ポートの各ビットを 1 ビット含む 16 進数文字列として指定します。たとえば、ポート幅が 1 の RAMB16\_S1 の場合、出力レジスタは 1 ビットなので、INIT および SRVAL には 1 または 0 を指定します。ポート幅が 4 の RAMB16\_S4 の場合は、出力レジスタが 4 ビットなので、0 ~ F の 16 進数値を指定します。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT または SRVAL の値の上位ビットで指定します。

#### 書き込みモードの選択

WRITE\_MODE 属性は、RAMB16 メモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE は WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。WRITE\_MODE を READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。WRITE\_MODE を NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DO 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定
SRVAL	2 進数/ 16 進数	任意の値	すべてゼロ	SSR ピンをアサートすると、DO 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
WRITE_MODE	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DO ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S18: 1k x 16 + 2 Parity bits Single-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S18_inst : RAMB16_S18
generic map (
  INIT => X"00000", -- Value of output RAM registers at startup
  SRVAL => X"00000", -- Output value upon SSR assertion
  WRITE_MODE => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 255
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 256 to 511
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 512 to 767
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 768 to 1023
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Address 0 to 255
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 256 to 511
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 512 to 767
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 768 to 1023
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DO => DO,      -- 16-bit Data Output
  DOP => DOP,     -- 2-bit parity Output
  ADDR => ADDR,   -- 10-bit Address Input
  CLK => CLK,     -- Clock
  DI => DI,       -- 16-bit Data Input
  DIP => DIP,     -- 2-bit parity Input
  EN => EN,       -- RAM Enable Input
  SSR => SSR,     -- Synchronous Set/Reset Input
  WE => WE        -- Write Enable Input
);

-- End of RAMB16_S18_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S18: 1k x 16 + 2 Parity bits Single-Port RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S18_inst : RAMB16_S18
generic map (
  INIT => X"00000", -- Value of output RAM registers at startup
  SRVAL => X"00000", -- Output value upon SSR assertion
  WRITE_MODE => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 255
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",

```



```
INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 256 to 511
INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 512 to 767
INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 768 to 1023
INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Address 0 to 255
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 256 to 511
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 512 to 767
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 768 to 1023
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
port map (
DO => DO,          -- 16-bit Data Output
DOP => DOP,         -- 2-bit parity Output
ADDR => ADDR,       -- 10-bit Address Input
CLK => CLK,         -- Clock
DI => DI,           -- 16-bit Data Input
```

```
DIP => DIP,      -- 2-bit parity Input
EN  => EN,       -- RAM Enable Input
SSR => SSR,      -- Synchronous Set/Reset Input
WE  => WE        -- Write Enable Input
);

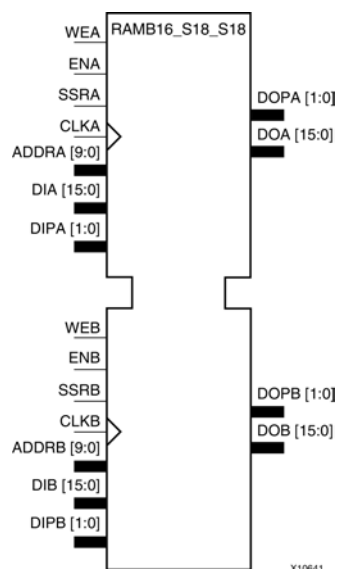
-- End of RAMB16_S18_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S18\_S18

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 18-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号

INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_A = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、data <sup>3</sup>	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号  
INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。  
SRVAL\_B = レジスタの値  
addr = RAM アドレス  
RAM(addr) = アドレス ADDR の RAM の内容  
data = RAM の入力データ  
pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレ ス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレ ス バス	データ バス	パリティ バス
RAMB16_S18_S18	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マッピング

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<-----	3				2				1				0
2	1024	<-----	1								0				
4	512	<-----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリセルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロック SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリセルにアクセスできます。ただし、一方のポートがあるメモリセルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリセルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB



WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット(0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット(0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S18_S18: 1k x 16 + 2 Parity bits Dual-Port RAM
--                      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S18_S18_inst : RAMB16_S18_S18
generic map (
  INIT_A => X"00000", -- Value of output RAM registers on Port A at startup
  INIT_B => X"00000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"00000", -- Port A output value upon SSR assertion
  SRVAL_B => X"00000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 255
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 256 to 511
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 512 to 767
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 768 to 1023
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Address 0 to 255
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 256 to 511
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 512 to 767
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 768 to 1023
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 16-bit Data Output
  DOB => DOB,      -- Port B 16-bit Data Output
  DOPA => DOPA,    -- Port A 2-bit Parity Output
  DOPB => DOPB,    -- Port B 2-bit Parity Output
  ADDRA => ADDRA,  -- Port A 10-bit Address Input
  ADDRb => ADDRb,  -- Port B 10-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKB => CLKB,    -- Port B Clock
  DIA => DIA,      -- Port A 16-bit Data Input
  DIB => DIB,      -- Port B 16-bit Data Input
  DIPa => DIPa,    -- Port A 2-bit parity Input
  DIPB => DIPB,    -- Port B 2-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- Port B RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S18_S18_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S18_S18: 1k x 16 + 2 Parity bits Dual-Port RAM
-- Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S18_S18_inst : RAMB16_S18_S18
generic map (
  INIT_A => X"00000", -- Value of output RAM registers on Port A at startup
  INIT_B => X"00000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"00000", -- Port A output value upon SSR assertion
  SRVAL_B => X"00000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 255
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

293

```

INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 768 to 1023
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 16-bit Data Output
  DOB => DOB,      -- Port B 16-bit Data Output
  DOPA => DOPA,     -- Port A 2-bit Parity Output
  DOPB => DOPB,     -- Port B 2-bit Parity Output
  ADDRA => ADDRA,   -- Port A 10-bit Address Input
  ADDRb => ADDRb,   -- Port B 10-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,       -- Port A 16-bit Data Input
  DIB => DIB,       -- Port B 16-bit Data Input
  DIPa => DIPa,     -- Port A 2-bit parity Input
  DIPB => DIPB,     -- Port-B 2-bit parity Input
  ENA => ENA,       -- Port A RAM Enable Input
  ENB => ENB,       -- PortB RAM Enable Input
  SSRA => SSRA,     -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,     -- Port B Synchronous Set/Reset Input
  WEA => WEA,       -- Port A Write Enable Input
  WEB => WEB        -- Port B Write Enable Input
);

-- End of RAMB16_S18_S18_inst instantiation

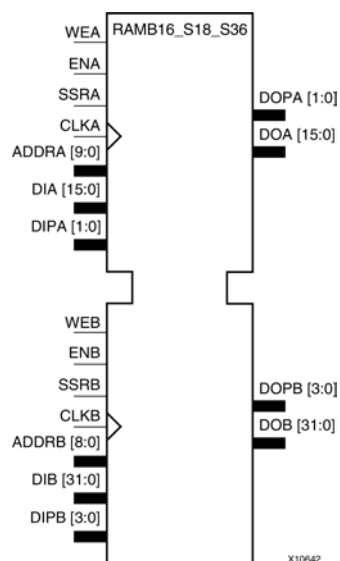
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S18\_S36

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 18-bit and 36-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号

INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_A = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST



論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、data <sup>3</sup>	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号  
INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。  
SRVAL\_B = レジスタの値  
addr = RAM アドレス  
RAM(addr) = アドレス ADDR の RAM の内容  
data = RAM の入力データ  
pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレ ス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレ ス バス	データ バス	パリティ バス
RAMB16_S18_S36	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるときに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるときに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるときに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるときに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マッピング

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティ メモリ セルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR\_port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<--	3				2				1				0
2	1024	<--	1								0				
4	512	<--	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリセルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロック SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリセルにアクセスできます。ただし、一方のポートがあるメモリセルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリセルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される時のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される時のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S18_S36: 1k/512 x 16/32 + 2/4 Parity bits Dual-Port RAM
-- Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S18_S36_inst : RAMB16_S18_S36
generic map (
  INIT_A => X"00000", -- Value of output RAM registers on Port A at startup
  INIT_B => X"000000000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"00000", -- Port A output value upon SSR assertion
  SRVAL_B => X"000000000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 255, Port B Address 0 to 127
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 256 to 511, Port B Address 128 to 255
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 512 to 767, Port B Address 256 to 383
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 768 to 1023, Port B Address 384 to 511
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

[illegible]

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

[illegible]



305

```

INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 768 to 1023, Port B Address 384 to 511
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 16-bit Data Output
  DOB => DOB,      -- Port B 32-bit Data Output
  DOPA => DOPA,     -- Port A 2-bit Parity Output
  DOPB => DOPB,     -- Port B 4-bit Parity Output
  ADDRA => ADDRA,   -- Port A 10-bit Address Input
  ADDRb => ADDRb,   -- Port B 9-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,       -- Port A 16-bit Data Input
  DIB => DIB,       -- Port B 32-bit Data Input
  DIPa => DIPa,     -- Port A 2-bit parity Input
  DIPB => DIPB,     -- Port-B 4-bit parity Input
  ENA => ENA,       -- Port A RAM Enable Input
  ENB => ENB,       -- PortB RAM Enable Input
  SSRA => SSRA,     -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,     -- Port B Synchronous Set/Reset Input
  WEA => WEA,       -- Port A Write Enable Input
  WEB => WEB        -- Port B Write Enable Input
);

-- End of RAMB16_S18_S36_inst instantiation

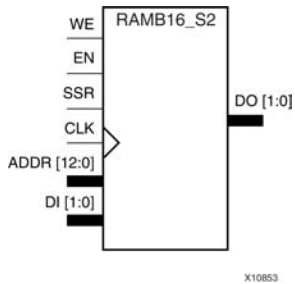
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S2

: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 2-bit Port



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた専用 RAM ブロックです。ブロック RAM のポートには 16384 ビットのデータ メモリがあります。セルの構成を、次の表に示します。

データ セル		パリティ セル				
ワード数	幅	ワード数	幅	アドレス バス	データ バス	パリティ バス
8192	2	–	–	(12:0)	(1:0)	–

イネーブル ピン EN は、ポートの読み出し、書き込み、リセットを制御します。EN が Low の場合、データは書き込まれず、出力 (DO および DOP) は変化しません。EN とリセット (SSR) が High の場合、クロック (CLK) が Low から High に切り替わるときに DO および DOP が SRVAL にセットされます。ライト イネーブル (WE) も High の場合は、DI および DIP の値が RAM に書き込まれます。EN が High で SSR と WE が Low の場合、クロックが Low から High に切り替わるときに、RAM アドレス (ADDR) に格納されているデータが読み出されます。SSR が Low で EN と WE が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDR) で選択されているワードにデータ入力 (DI および DIP) の値が読み込まれます。データ出力 (DO および DOP) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

前述の説明では、制御ピン (EN、WE、SSR、CLK) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

## 論理表

入力								出力			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT	INIT	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR=グローバル セットリセット信号

INIT = データ メモリ用に INIT 属性で設定された値。デフォルトはすべて 0 です。

SRVAL = SRVAL 属性で設定された SSR のアサート後の値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE=NO\_CHANGE

<sup>2</sup> WRITE\_MODE=READ\_FIRST

<sup>3</sup> WRITE\_MODE=WRITE\_FIRST

## 初期化

## メモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16 の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## 出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。シングルポート RAMB16 の出力レジスタの初期化に使用する属性には、INIT と SRVAL の 2 種類があります。INIT 属性は、電源投入時の出力レジスタの値を指定します。SRVAL 属性は、SSR (セット/リセット) 入力をアサートしたときの値を指定します。

INIT および SRVAL 属性は、初期化値を、出力ポートの各ビットを 1 ビット含む 16 進数文字列として指定します。たとえば、ポート幅が 1 の RAMB16\_S1 の場合、出力レジスタは 1 ビットなので、INIT および SRVAL には 1 または 0 を指定します。ポート幅が 4 の RAMB16\_S4 の場合は、出力レジスタが 4 ビットなので、0 ~ F の 16 進数値を指定します。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT または SRVAL の値の上位ビットで指定します。

#### 書き込みモードの選択

WRITE\_MODE 属性は、RAMB16 メモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE は WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。WRITE\_MODE を READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。WRITE\_MODE を NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DO 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定
SRVAL	2 進数/ 16 進数	任意の値	すべてゼロ	SSR ピンをアサートすると、DO 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
WRITE_MODE	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DO ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S2: 8k x 2 Single-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S2_inst : RAMB16_S2
generic map (
  INIT => X"0", -- Value of output RAM registers at startup
  SRVAL => X"0", -- Output value upon SSR assertion
  WRITE_MODE => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 2047
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 2048 to 4095
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 4096 to 6143
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 6143 to 8191
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
    DO => DO,          -- 2-bit Data Output
    ADDR => ADDR,      -- 13-bit Address Input
    CLK => CLK,         -- Clock
    DI => DI,          -- 2-bit Data Input
    EN => EN,          -- RAM Enable Input
    SSR => SSR,         -- Synchronous Set/Reset Input
    WE => WE           -- Write Enable Input
);

-- End of RAMB16_S2_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S2: 8k x 2 Single-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S2_inst : RAMB16_S2
generic map (
    INIT => X"0", -- Value of output RAM registers at startup
    SRVAL => X"0", -- Output value upon SSR assertion
    WRITE_MODE => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
    -- The following INIT_xx declarations specify the initial contents of the RAM
    -- Address 0 to 2047
    INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
    -- Address 2048 to 4095
    INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",

```

```
INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 4096 to 6143
INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 6143 to 8191
INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DO => DO,      -- 2-bit Data Output
  ADDR => ADDR,   -- 13-bit Address Input
  CLK => CLK,     -- Clock
  DI => DI,       -- 2-bit Data Input
  EN => EN,       -- RAM Enable Input
  SSR => SSR,     -- Synchronous Set/Reset Input
  WE => WE        -- Write Enable Input
);

-- End of RAMB16_S2_inst instantiation
```

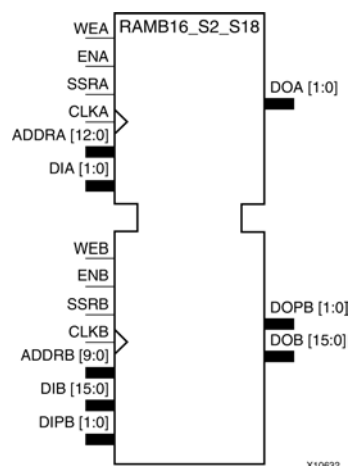
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## RAMB16\_S2\_S18

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 18-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号

INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_A = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、data <sup>3</sup>	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号  
INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。  
SRVAL\_B = レジスタの値  
addr = RAM アドレス  
RAM(addr) = アドレス ADDR の RAM の内容  
data = RAM の入力データ  
pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S2_S18	8192 x 2	—	(12:0)	(1:0)	—	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マップ

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																			
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0			
4	4096	<--	7				6				5				4				3				2				1				0					
8	2048	<--	3								2								1								0									
16	1024	<--	1																0																	
32	512	<--	0																																	

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリ セルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロックの SelectIO™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリ セルにアクセスできます。ただし、一方のポートがあるメモリ セルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリ セルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット(0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット(0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される時のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される時のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

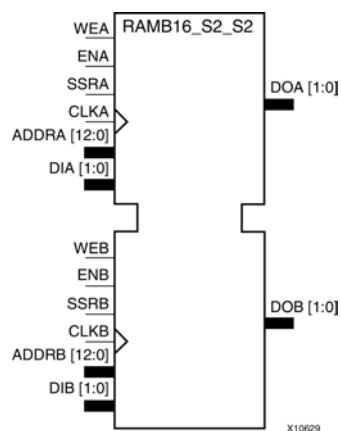
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## RAMB16\_S2\_S2

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号

INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_A = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、data <sup>3</sup>	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号  
INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。  
SRVAL\_B = レジスタの値  
addr = RAM アドレス  
RAM(addr) = アドレス ADDR の RAM の内容  
data = RAM の入力データ  
pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S2_S2	8192 x 2	—	(12:0)	(1:0)	—	8192 x 2	—	(12:0)	(1:0)	—

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マップ

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR\_port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A および上記のデバイスの場合、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリ セルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロックの SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリ セルにアクセスできます。ただし、一方のポートがあるメモリ セルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリ セルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、または NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ:</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときのポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。



属性	タイプ	値	デフォルト	説明
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S2_S2: 8k x 2 Dual-Port RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S2_S2_inst : RAMB16_S2_S2
generic map (
    INIT_A => X"0", -- Value of output RAM registers on Port A at startup
    INIT_B => X"0", -- Value of output RAM registers on Port B at startup
    SRVAL_A => X"0", -- Port A output value upon SSR assertion
    SRVAL_B => X"0", -- Port B output value upon SSR assertion
    WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
    WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
    SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
    -- The following INIT_xx declarations specify the initial contents of the RAM
    -- Address 0 to 2047
    INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
    -- Address 2048 to 4095
    INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 4096 to 6143
INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 6143 to 8191
INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 2-bit Data Output
  DOB => DOB,      -- Port B 2-bit Data Output
  ADDRA => ADDRA,   -- Port A 13-bit Address Input
  ADDR_B => ADDR_B, -- Port B 13-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,      -- Port A 2-bit Data Input
  DIB => DIB,      -- Port B 2-bit Data Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- Port B RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S2_S2_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S2_S2: 8k x 2 Dual-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S2_S2_inst : RAMB16_S2_S2
generic map (
  INIT_A => X"0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"0", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"0", -- Port A output value upon SSR assertion
  SRVAL_B => X"0", -- Port B output value upon SSR assertion

```

Spartan-3E ライブラリ ガイド (HDL 用)  
UG617 (v12.2) 2010 年 7 月 23 日

```
DOA => DOA,      -- Port A 2-bit Data Output
DOB => DOB,      -- Port B 2-bit Data Output
ADDRA => ADDRA,   -- Port A 13-bit Address Input
ADDRB => ADDRb,   -- Port B 13-bit Address Input
CLKA => CLKA,     -- Port A Clock
CLKB => CLKB,     -- Port B Clock
DIA => DIA,      -- Port A 2-bit Data Input
DIB => DIB,      -- Port B 2-bit Data Input
ENA => ENA,      -- Port A RAM Enable Input
ENB => ENB,      -- Port B RAM Enable Input
SSRA => SSRA,     -- Port A Synchronous Set/Reset Input
SSRB => SSRB,     -- Port B Synchronous Set/Reset Input
WEA => WEA,      -- Port A Write Enable Input
WEB => WEB,      -- Port B Write Enable Input
);

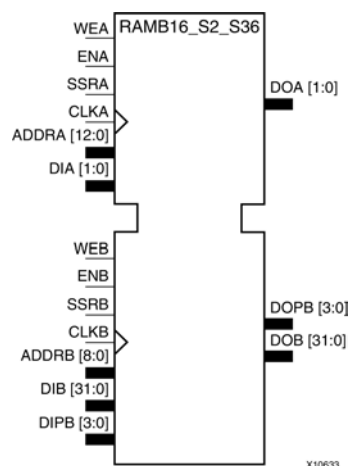
-- End of RAMB16_S2_S2_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S2\_S36

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 36-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号

INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_A = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、data <sup>3</sup>	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号  
INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。  
SRVAL\_B = レジスタの値  
addr = RAM アドレス  
RAM(addr) = アドレス ADDR の RAM の内容  
data = RAM の入力データ  
pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S2_S36	8192 x 2	—	(12:0)	(1:0)	—	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マップ

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR\_port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																



## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリ セルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロックの SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリ セルにアクセスできます。ただし、一方のポートがあるメモリ セルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリ セルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

341

```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Port B Address 0 to 127
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 128 to 255
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 256 to 383
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 384 to 511
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 2-bit Data Output
  DOB => DOB,      -- Port B 32-bit Data Output
  DOPB => DOPB,    -- Port B 4-bit Parity Output
  ADDRA => ADDRA,   -- Port A 13-bit Address Input
  ADDRb => ADDRb,  -- Port B 9-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKb => CLKb,    -- Port B Clock
  DIA => DIA,      -- Port A 2-bit Data Input
  DIB => DIB,      -- Port B 32-bit Data Input
  DIPB => DIPB,    -- Port-B 4-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- PortB RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRb => SSRb,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S2_S36_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S2_S36: 8k/512 x 2/32 + 0/4 Parity bits Dual-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S2_S36_inst : RAMB16_S2_S36
generic map (
  INIT_A => X"0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"000000000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"0", -- Port A output value upon SSR assertion
  SRVAL_B => X"000000000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 2047, Port B Address 0 to 127
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

343

```

-- Port B Address 384 to 511
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 2-bit Data Output
  DOB => DOB,      -- Port B 32-bit Data Output
  DOPB => DOPB,    -- Port B 4-bit Parity Output
  ADDRA => ADDRA,  -- Port A 13-bit Address Input
  ADDRb => ADDRb,  -- Port B 9-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKB => CLKB,    -- Port B Clock
  DIA => DIA,      -- Port A 2-bit Data Input
  DIB => DIB,      -- Port B 32-bit Data Input
  DIPB => DIPB,    -- Port-B 4-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- PortB RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S2_S36_inst instantiation

```

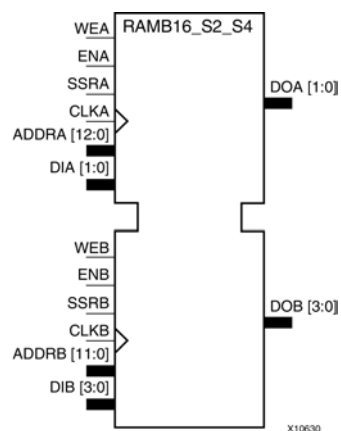
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## RAMB16\_S2\_S4

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 4-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号

INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_A = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、data <sup>3</sup>	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号  
INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。  
SRVAL\_B = レジスタの値  
addr = RAM アドレス  
RAM(addr) = アドレス ADDR の RAM の内容  
data = RAM の入力データ  
pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S2_S4	8192 x 2	—	(12:0)	(1:0)	—	4096 x 4	—	(11:0)	(3:0)	—

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マップ

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR\_port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリ セルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロックの SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリ セルにアクセスできます。ただし、一方のポートがあるメモリ セルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリ セルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、NONE、WARNING、GENERATE_X_ONLY	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>・ ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>・ WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>・ GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>・ NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、READ_FIRST、NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、READ_FIRST、NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。



```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S2_S4: 8k/4k x 2/4 Dual-Port RAM
-- Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S2_S4_inst : RAMB16_S2_S4
generic map (
  INIT_A => X"0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"0", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"0", -- Port A output value upon SSR assertion
  SRVAL_B => X"0", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 2047, Port B Address 0 to 1023
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 2048 to 4095, Port B Address 1024 to 2047
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 4096 to 6143, Port B Address 2048 to 3071
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 6144 to 8191, Port B Address 3072 to 4095
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 2-bit Data Output
  DOB => DOB,      -- Port B 4-bit Data Output
  ADDRA => ADDRA,   -- Port A 13-bit Address Input
  ADDR_B => ADDR_B, -- Port B 12-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,      -- Port A 2-bit Data Input
  DIB => DIB,      -- Port B 4-bit Data Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- Port B RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S2_S4_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S2_S4: 8k/4k x 2/4 Dual-Port RAM
-- Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S2_S4_inst : RAMB16_S2_S4
generic map (
  INIT_A => X"0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"0", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"0", -- Port A output value upon SSR assertion
  SRVAL_B => X"0", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 2047, Port B Address 0 to 1023
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 2048 to 4095, Port B Address 1024 to 2047
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

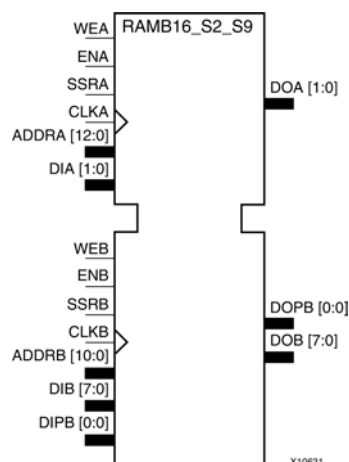
[illegible]

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリ データシート](#)

## RAMB16\_S2\_S9

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 9-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号

INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_A = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、data <sup>3</sup>	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号  
INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。  
SRVAL\_B = レジスタの値  
addr = RAM アドレス  
RAM(addr) = アドレス ADDR の RAM の内容  
data = RAM の入力データ  
pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S2_S9	8192 x 2	—	(12:0)	(1:0)	—	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マップ

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																



## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリ セルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan®-3A ブロックの SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリ セルにアクセスできます。ただし、一方のポートがあるメモリ セルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリ セルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、NONE、WARNING、GENERATE_X_ONLY	ALL	RAM の 2 つのポートから 1 つのアドレスに対して同時に書き込み/読み出しが実行されるというデータの競合が発生した場合の、シミュレーションの動作を指定します。デフォルトの ALL に設定すると、コンソールに警告メッセージが出力され、不定値を表す X が生成されます。可能な限りこの設定を使用してください。WARNING に設定すると、警告メッセージのみが出力されます。GENERATE_X_ONLY に設定すると、不定値を表す X が生成されますが、コンソールには何も出力されません。NONE に設定すると、エラーが無視されます。競合が発生したときのデータを確認する場合は、この属性のみを変更することをお勧めします。
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット(0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット(0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S2_S9: 8k/2k x 2/8 + 0/1 Parity bit Dual-Port RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S2_S9_inst : RAMB16_S2_S9
generic map (
```

```
INIT_A => X"0", -- Value of output RAM registers on Port A at startup
INIT_B => X"000", -- Value of output RAM registers on Port B at startup
SRVAL_A => X"0", -- Port A output value upon SSR assertion
SRVAL_B => X"000", -- Port B output value upon SSR assertion
WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
-- The following INIT_xx declarations specify the initial contents of the RAM
-- Port A Address 0 to 2047, Port B Address 0 to 511
INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 2048 to 4095, Port B Address 512 to 1023
INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 4096 to 6143, Port B Address 1024 to 1535
INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 6144 to 8191, Port B Address 1536 to 2047
INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Port B Address 0 to 511
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 512 to 1023
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 1024 to 1535
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 6144 to 8191, Port B Address 1536 to 2047
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 2-bit Data Output
  DOB => DOB,      -- Port B 8-bit Data Output
  DOPB => DOPB,    -- Port B 1-bit Parity Output
  ADDRA => ADDRA,  -- Port A 13-bit Address Input
  ADDRb => ADDRb,  -- Port B 11-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKB => CLKB,    -- Port B Clock
  DIA => DIA,      -- Port A 2-bit Data Input
  DIB => DIB,      -- Port B 8-bit Data Input
  DIPB => DIPB,    -- Port-B 1-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- Port B RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S2_S9_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S2_S9: 8k/2k x 2/8 + 0/1 Parity bit Dual-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S2_S9_inst : RAMB16_S2_S9
generic map (
  INIT_A => X"0",  -- Value of output RAM registers on Port A at startup
  INIT_B => X"000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"0",  -- Port A output value upon SSR assertion
  SRVAL_B => X"000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 2047, Port B Address 0 to 511
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",

```

```
INIT_OC => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_OD => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_OE => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_OF => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 2048 to 4095, Port B Address 512 to 1023
INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 4096 to 6143, Port B Address 1024 to 1535
INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 6144 to 8191, Port B Address 1536 to 2047
INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Port B Address 0 to 511
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 512 to 1023
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 1024 to 1535
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 6144 to 8191, Port B Address 1536 to 2047
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 2-bit Data Output
  DOB => DOB,      -- Port B 8-bit Data Output
  DOPB => DOPB,    -- Port B 1-bit Parity Output
  ADDRA => ADDRA,  -- Port A 13-bit Address Input
```

```
ADDRB => ADDRb,  -- Port B 11-bit Address Input
CLKA => CLKA,    -- Port A Clock
CLKB => CLKB,    -- Port B Clock
DIA => DIA,      -- Port A 2-bit Data Input
DIB => DIB,      -- Port B 8-bit Data Input
DIPB => DIPB,    -- Port-B 1-bit parity Input
ENA => ENA,      -- Port A RAM Enable Input
ENB => ENB,      -- Port B RAM Enable Input
SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
WEA => WEA,      -- Port A Write Enable Input
WEB => WEB,      -- Port B Write Enable Input
);

-- End of RAMB16_S2_S9_inst instantiation
```

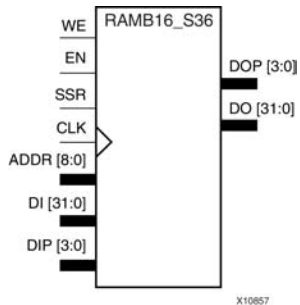
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## RAMB16\_S36

: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 36-bit Port



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた専用 RAM ブロックです。ブロック RAM のポートには 16384 ビットのデータ メモリがあります。セルの構成を、次の表に示します。

データ セル		パリティ セル				
ワード数	幅	ワード数	幅	アドレス バス	データ バス	パリティ バス
512	32	512	4	(8:0)	(31:0)	(3:0)

イネーブル ピン EN は、ポートの読み出し、書き込み、リセットを制御します。EN が Low の場合、データは書き込まれず、出力 (DO および DOP) は変化しません。EN とリセット (SSR) が High の場合、クロック (CLK) が Low から High に切り替わるときに DO および DOP が SRVAL にセットされます。ライト イネーブル (WE) も High の場合は、DI および DIP の値が RAM に書き込まれます。EN が High で SSR と WE が Low の場合、クロックが Low から High に切り替わるときに、RAM アドレス (ADDR) に格納されているデータが読み出されます。SSR が Low で EN と WE が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDR) で選択されているワードにデータ入力 (DI および DIP) の値が読み込まれます。データ出力 (DO および DOP) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

前述の説明では、制御ピン (EN、WE、SSR、CLK) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

## 論理表

入力								出力			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT	INIT	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR=グローバル セットリセット信号

INIT = データ メモリ用に INIT 属性で設定された値。デフォルトはすべて 0 です。

SRVAL = SRVAL 属性で設定された SSR のアサート後の値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE=NO\_CHANGE

<sup>2</sup> WRITE\_MODE=READ\_FIRST

<sup>3</sup> WRITE\_MODE=WRITE\_FIRST

## 初期化

## メモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16 の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## 出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。シングルポート RAMB16 の出力レジスタの初期化に使用する属性には、INIT と SRVAL の 2 種類があります。INIT 属性は、電源投入時の出力レジスタの値を指定します。SRVAL 属性は、SSR (セット/リセット) 入力をアサートしたときの値を指定します。

INIT および SRVAL 属性は、初期化値を、出力ポートの各ビットを 1 ビット含む 16 進数文字列として指定します。たとえば、ポート幅が 1 の RAMB16\_S1 の場合、出力レジスタは 1 ビットなので、INIT および SRVAL には 1 または 0 を指定します。ポート幅が 4 の RAMB16\_S4 の場合は、出力レジスタが 4 ビットなので、0 ~ F の 16 進数値を指定します。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT または SRVAL の値の上位ビットで指定します。

#### 書き込みモードの選択

WRITE\_MODE 属性は、RAMB16 メモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE は WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。WRITE\_MODE を READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。WRITE\_MODE を NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DO 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定
SRVAL	2 進数/ 16 進数	任意の値	すべてゼロ	SSR ピンをアサートすると、DO 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
WRITE_MODE	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DO ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。



```

INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Address 0 to 127
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 128 to 255
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 256 to 383
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 384 to 511
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
DO => DO,      -- 32-bit Data Output
DOP => DOP,     -- 4-bit parity Output
ADDR => ADDR,   -- 9-bit Address Input
CLK => CLK,     -- Clock
DI => DI,      -- 32-bit Data Input
DIP => DIP,     -- 4-bit parity Input
EN => EN,      -- RAM Enable Input
SSR => SSR,    -- Synchronous Set/Reset Input
WE => WE       -- Write Enable Input
);

-- End of RAMB16_S36_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S36: 512 x 32 + 4 Parity bits Single-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S36_inst : RAMB16_S36
generic map (
  INIT => X"0000000000", -- Value of output RAM registers at startup
  SRVAL => X"0000000000", -- Output value upon SSR assertion
  WRITE_MODE => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 127
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",

```

Spartan-3E ライブラリ ガイド (HDL 用)  
UG617 (v12.2) 2010 年 7 月 23 日

```
DIP => DIP,      -- 4-bit parity Input
EN  => EN,       -- RAM Enable Input
SSR => SSR,      -- Synchronous Set/Reset Input
WE  => WE        -- Write Enable Input
);

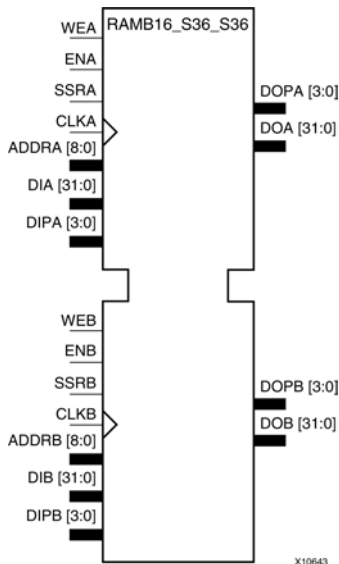
-- End of RAMB16_S36_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S36\_S36

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with Two 36-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。



## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号  
 INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。  
 SRVAL\_A = レジスタの値  
 addr = RAM アドレス  
 RAM(addr) = アドレス ADDR の RAM の内容  
 data = RAM の入力データ  
 pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号

INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_B = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレ ス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレ ス バス	データ バス	パリティ バス
RAMB16_S36_S36	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マッピング

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																			
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0			
4	4096	<--	7				6				5				4				3				2				1				0					
8	2048	<--	3								2								1								0									
16	1024	<--	1																0																	
32	512	<--	0																																	

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<---	3				2				1				0
2	1024	<---	1								0				
4	512	<---	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリセルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロック SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリセルにアクセスできます。ただし、一方のポートがあるメモリセルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリセルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S36_S36: 512 x 32 + 4 Parity bits Dual-Port RAM
--                Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S36_S36_inst : RAMB16_S36_S36
generic map (
  INIT_A => X"000000000", -- Value of output RAM registers on Port A at startup
  INIT_B => X"000000000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"000000000", -- Port A output value upon SSR assertion
  SRVAL_B => X"000000000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 127
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 128 to 255
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 256 to 383
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 384 to 511
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```



```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Address 0 to 127
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 128 to 255
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 256 to 383
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 384 to 511
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 32-bit Data Output
  DOB => DOB,      -- Port B 32-bit Data Output
  DOPA => DOPA,    -- Port A 4-bit Parity Output
  DOPB => DOPB,    -- Port B 4-bit Parity Output
  ADDRA => ADDRA,  -- Port A 9-bit Address Input
  ADDRb => ADDRb,  -- Port B 9-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKB => CLKB,    -- Port B Clock
  DIA => DIA,      -- Port A 32-bit Data Input
  DIB => DIB,      -- Port B 32-bit Data Input
  DIPA => DIPA,    -- Port A 4-bit parity Input
  DIPB => DIPB,    -- Port B 4-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- Port B RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S36_S36_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S36_S36: 512 x 32 + 4 Parity bits Dual-Port RAM
-- Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S36_S36_inst : RAMB16_S36_S36
generic map (
  INIT_A => X"000000000", -- Value of output RAM registers on Port A at startup
  INIT_B => X"000000000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"000000000", -- Port A output value upon SSR assertion
  SRVAL_B => X"000000000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 127
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

Spartan-3E ライブラリ ガイド (HDL 用)  
UG617 (v12.2) 2010 年 7 月 23 日

```

INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 384 to 511
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 32-bit Data Output
  DOB => DOB,      -- Port B 32-bit Data Output
  DOPA => DOPA,    -- Port A 4-bit Parity Output
  DOPB => DOPB,    -- Port B 4-bit Parity Output
  ADDRA => ADDRA,  -- Port A 9-bit Address Input
  ADDRb => ADDRb,  -- Port B 9-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKB => CLKB,    -- Port B Clock
  DIA => DIA,      -- Port A 32-bit Data Input
  DIB => DIB,      -- Port B 32-bit Data Input
  DIPa => DIPa,    -- Port A 4-bit parity Input
  DIPB => DIPB,    -- Port B 4-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- Port B RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S36_S36_inst instantiation

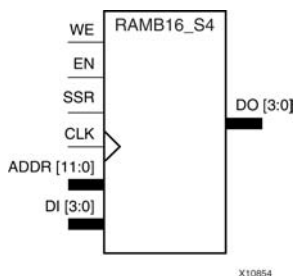
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S4

: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 4-bit Port



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた専用 RAM ブロックです。ブロック RAM のポートには 16384 ビットのデータ メモリがあります。セルの構成を、次の表に示します。

データ セル		パリティ セル				
ワード数	幅	ワード数	幅	アドレス バス	データ バス	パリティ バス
4096	4	–	–	(11:0)	(3:0)	–

イネーブル ピン EN は、ポートの読み出し、書き込み、リセットを制御します。EN が Low の場合、データは書き込まれず、出力 (DO および DOP) は変化しません。EN とリセット (SSR) が High の場合、クロック (CLK) が Low から High に切り替わるときに DO および DOP が SRVAL にセットされます。ライト イネーブル (WE) も High の場合は、DI および DIP の値が RAM に書き込まれます。EN が High で SSR と WE が Low の場合、クロックが Low から High に切り替わるときに、RAM アドレス (ADDR) に格納されているデータが読み出されます。SSR が Low で EN と WE が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDR) で選択されているワードにデータ入力 (DI および DIP) の値が読み込まれます。データ出力 (DO および DOP) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

前述の説明では、制御ピン (EN、WE、SSR、CLK) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

## 論理表

入力								出力			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT	INIT	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR=グローバル セットリセット信号

INIT = データ メモリ用に INIT 属性で設定された値。デフォルトはすべて 0 です。

SRVAL = SRVAL 属性で設定された SSR のアサート後の値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE=NO\_CHANGE

<sup>2</sup> WRITE\_MODE=READ\_FIRST

<sup>3</sup> WRITE\_MODE=WRITE\_FIRST

## 初期化

## メモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16 の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## 出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。シングルポート RAMB16 の出力レジスタの初期化に使用する属性には、INIT と SRVAL の 2 種類があります。INIT 属性は、電源投入時の出力レジスタの値を指定します。SRVAL 属性は、SSR (セット/リセット) 入力をアサートしたときの値を指定します。

INIT および SRVAL 属性は、初期化値を、出力ポートの各ビットを 1 ビット含む 16 進数文字列として指定します。たとえば、ポート幅が 1 の RAMB16\_S1 の場合、出力レジスタは 1 ビットなので、INIT および SRVAL には 1 または 0 を指定します。ポート幅が 4 の RAMB16\_S4 の場合は、出力レジスタが 4 ビットなので、0 ~ F の 16 進数値を指定します。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT または SRVAL の値の上位ビットで指定します。

#### 書き込みモードの選択

WRITE\_MODE 属性は、RAMB16 メモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE は WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。WRITE\_MODE を READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。WRITE\_MODE を NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DO 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定
SRVAL	2 進数/ 16 進数	任意の値	すべてゼロ	SSR ピンをアサートすると、DO 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
WRITE_MODE	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DO ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S4: 4k x 4 Single-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S4_inst : RAMB16_S4
generic map (
  INIT => X"0", -- Value of output RAM registers at startup
  SRVAL => X"0", -- Output value upon SSR assertion
  WRITE_MODE => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 1023
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 1024 to 2047
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 2048 to 3071
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 3072 to 4095
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000"
```

```

INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DO => DO,          -- 4-bit Data Output
  ADDR => ADDR,       -- 12-bit Address Input
  CLK => CLK,         -- Clock
  DI => DI,          -- 4-bit Data Input
  EN => EN,           -- RAM Enable Input
  SSR => SSR,         -- Synchronous Set/Reset Input
  WE => WE           -- Write Enable Input
);

-- End of RAMB16_S4_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S4: 4k x 4 Single-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S4_inst : RAMB16_S4
generic map (
  INIT => X"0", -- Value of output RAM registers at startup
  SRVAL => X"0", -- Output value upon SSR assertion
  WRITE_MODE => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 1023
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 1024 to 2047
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",

```



```
INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 2048 to 3071
INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 3072 to 4095
INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
DO => DO,      -- 4-bit Data Output
ADDR => ADDR,  -- 12-bit Address Input
CLK => CLK,    -- Clock
DI => DI,      -- 4-bit Data Input
EN => EN,      -- RAM Enable Input
SSR => SSR,    -- Synchronous Set/Reset Input
WE => WE       -- Write Enable Input
);

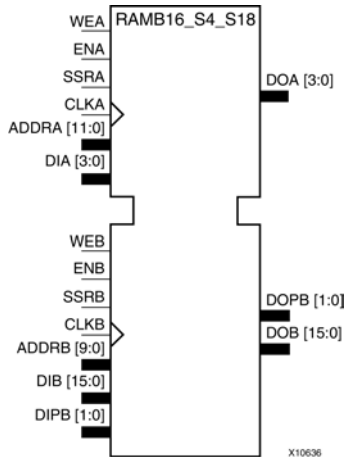
-- End of RAMB16_S4_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S4\_S18

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit and 18-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポート には、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号  
 INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。  
 SRVAL\_A = レジスタの値  
 addr = RAM アドレス  
 RAM(addr) = アドレス ADDR の RAM の内容  
 data = RAM の入力データ  
 pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号

INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_B = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S4_S18	4096 x 4	—	(11:0)	(3:0)	—	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIP A の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR A) で選択されているワードにデータ入力 (DIA および DIP A) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マッピング

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティ メモリ セルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7			6				5				4				3				2				1				0					
8	2048	<--	3							2								1								0									
16	1024	<--	1															0																	
32	512	<--	0																																

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリセルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロック SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリセルにアクセスできます。ただし、一方のポートがあるメモリセルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリセルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

次の表に、各ポート幅のアドレス マップを示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定



属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S4_S18: 4k/1k x 4/16 + 0/2 Parity bits Dual-Port RAM
--                Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S4_S18_inst : RAMB16_S4_S18
generic map (
  INIT_A => X"0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"0000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"0", -- Port A output value upon SSR assertion
  SRVAL_B => X"0000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 1023, Port B Address 0 to 255
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 1024 to 2047, Port B Address 256 to 511
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 2048 to 3071, Port B Address 512 to 767
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 3072 to 4095, Port B Address 768 to 1023
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Port A Address 0 to 1023, Port B Address 0 to 255
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 1024 to 2047, Port B Address 256 to 511
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 2048 to 3071, Port B Address 512 to 767
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 3072 to 4095, Port B Address 768 to 1023
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 4-bit Data Output
  DOB => DOB,      -- Port B 16-bit Data Output
  DOPB => DOPB,    -- Port B 2-bit Parity Output
  ADDRA => ADDRA,   -- Port A 12-bit Address Input
  ADDRb => ADDRb,   -- Port B 10-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,      -- Port A 4-bit Data Input
  DIB => DIB,      -- Port B 16-bit Data Input
  DIPB => DIPB,    -- Port-B 2-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- PortB RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S4_S18_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S4_S18: 4k/1k x 4/16 + 0/2 Parity bits Dual-Port RAM
-- Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S4_S18_inst : RAMB16_S4_S18
generic map (
  INIT_A => X"0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"00000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"0", -- Port A output value upon SSR assertion
  SRVAL_B => X"00000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 1023, Port B Address 0 to 255
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

Spartan-3E ライブラリ ガイド (HDL 用)  
UG617 (v12.2) 2010 年 7 月 23 日

```
-- Port A Address 3072 to 4095, Port B Address 768 to 1023
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 4-bit Data Output
  DOB => DOB,      -- Port B 16-bit Data Output
  DOPB => DOPB,    -- Port B 2-bit Parity Output
  ADDRA => ADDRA,  -- Port A 12-bit Address Input
  ADDRb => ADDRb,  -- Port B 10-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKB => CLKB,    -- Port B Clock
  DIA => DIA,      -- Port A 4-bit Data Input
  DIB => DIB,      -- Port B 16-bit Data Input
  DIPB => DIPB,    -- Port-B 2-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- PortB RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

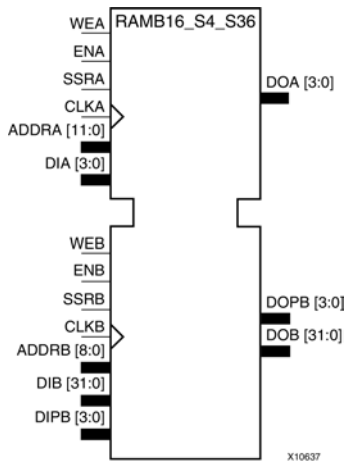
-- End of RAMB16_S4_S18_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S4\_S36

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit and 36-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号

INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_A = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号

INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_B = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S4_S36	4096 x 4	—	(11:0)	(3:0)	—	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)

(a) ワード数 X 幅



各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マップ

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																			
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0			
4	4096	<--	7				6				5				4				3				2				1				0					
8	2048	<--	3								2								1								0									
16	1024	<--	1															0																		
32	512	<--	0																																	

パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<-----	3				2				1				0
2	1024	<-----	1								0				
4	512	<-----	0												

デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリ セルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロックの SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリ セルにアクセスできます。ただし、一方のポートがあるメモリ セルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリ セルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S4_S36: 4k/512 x 4/32 + 0/4 Parity bits Dual-Port RAM
--                Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S4_S36_inst : RAMB16_S4_S36
generic map (
  INIT_A => X"0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"00000000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"0", -- Port A output value upon SSR assertion
  SRVAL_B => X"00000000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 1023, Port B Address 0 to 127
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 1024 to 2047, Port B Address 128 to 255
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 2048 to 3071, Port B Address 256 to 383
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 3072 to 4095, Port B Address 384 to 511
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Port A Address 0 to 1023, Port B Address 0 to 127
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 1024 to 2047, Port B Address 128 to 255
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 2048 to 3071, Port B Address 256 to 383
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 3072 to 4095, Port B Address 384 to 511
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
    DOA => DOA,      -- Port A 4-bit Data Output
    DOB => DOB,      -- Port B 32-bit Data Output
    DOPB => DOPB,    -- Port B 4-bit Parity Output
    ADDRA => ADDRA,   -- Port A 12-bit Address Input
    ADDRb => ADDRb,   -- Port B 9-bit Address Input
    CLKA => CLKA,     -- Port A Clock
    CLKb => CLKb,     -- Port B Clock
    DIA => DIA,      -- Port A 4-bit Data Input
    DIB => DIB,      -- Port B 32-bit Data Input
    DIPB => DIPB,    -- Port-B 4-bit parity Input
    ENA => ENA,      -- Port A RAM Enable Input
    ENB => ENB,      -- PortB RAM Enable Input
    SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
    SSRb => SSRb,    -- Port B Synchronous Set/Reset Input
    WEA => WEA,      -- Port A Write Enable Input
    WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S4_S36_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S4_S36: 4k/512 x 4/32 + 0/4 Parity bits Dual-Port RAM
--              Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S4_S36_inst : RAMB16_S4_S36
generic map (
    INIT_A => X"0", -- Value of output RAM registers on Port A at startup
    INIT_B => X"000000000", -- Value of output RAM registers on Port B at startup
    SRVAL_A => X"0", -- Port A output value upon SSR assertion
    SRVAL_B => X"000000000", -- Port B output value upon SSR assertion
    WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
    WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
    SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
    -- The following INIT_xx declarations specify the initial contents of the RAM
    -- Port A Address 0 to 1023, Port B Address 0 to 127
    INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
    INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",

```



Spartan-3E ライブラリ ガイド (HDL 用)  
UG617 (v12.2) 2010 年 7 月 23 日



```
-- Port A Address 3072 to 4095, Port B Address 384 to 511
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 4-bit Data Output
  DOB => DOB,      -- Port B 32-bit Data Output
  DOPB => DOPB,    -- Port B 4-bit Parity Output
  ADDRA => ADDRA,  -- Port A 12-bit Address Input
  ADDRb => ADDRb,  -- Port B 9-bit Address Input
  CLKA => CLKA,    -- Port A Clock
  CLKB => CLKB,    -- Port B Clock
  DIA => DIA,      -- Port A 4-bit Data Input
  DIB => DIB,      -- Port B 32-bit Data Input
  DIPB => DIPB,    -- Port-B 4-bit parity Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- PortB RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

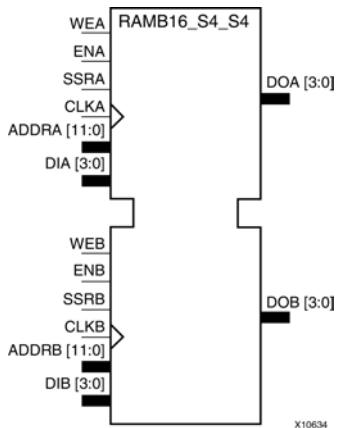
-- End of RAMB16_S4_S36_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S4\_S4

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit Ports



### 概要

このデザイン エレメントは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このエレメントのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号  
 INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。  
 SRVAL\_A = レジスタの値  
 addr = RAM アドレス  
 RAM(addr) = アドレス ADDR の RAM の内容  
 data = RAM の入力データ  
 pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号

INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_B = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S4_S4	4096 x 4	—	(11:0)	(3:0)	—	4096 x 4	—	(11:0)	(3:0)	—

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マップ

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR\_port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																			
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0			
4	4096	<--	7				6				5				4				3				2				1				0					
8	2048	<--	3								2								1								0									
16	1024	<--	1																0																	
32	512	<--	0																																	

パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリ セルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロックの SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリ セルにアクセスできます。ただし、一方のポートがあるメモリ セルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリ セルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定



属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S4_S4: 4k x 4 Dual-Port RAM
--                Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S4_S4_inst : RAMB16_S4_S4
generic map (
  INIT_A => X"0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"0", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"0", -- Port A output value upon SSR assertion
  SRVAL_B => X"0", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 1023
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 1024 to 2047
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 2048 to 3071
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 3072 to 4095
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 4-bit Data Output
  DOB => DOB,      -- Port B 4-bit Data Output
  ADDRA => ADDRA,   -- Port A 12-bit Address Input
  ADDR_B => ADDR_B, -- Port B 12-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,      -- Port A 4-bit Data Input
  DIB => DIB,      -- Port B 4-bit Data Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- Port B RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S4_S4_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S4_S4: 4k x 4 Dual-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S4_S4_inst : RAMB16_S4_S4
generic map (
  INIT_A => X"0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"0", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"0", -- Port A output value upon SSR assertion
  SRVAL_B => X"0", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 1023
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 1024 to 2047
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

```

INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 2048 to 3071
INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 3072 to 4095
INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
port map (
  DOA => DOA,      -- Port A 4-bit Data Output
  DOB => DOB,      -- Port B 4-bit Data Output
  ADDRA => ADDRA,   -- Port A 12-bit Address Input
  ADDRb => ADDRb,   -- Port B 12-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,      -- Port A 4-bit Data Input
  DIB => DIB,      -- Port B 4-bit Data Input
  ENA => ENA,      -- Port A RAM Enable Input
  ENB => ENB,      -- Port B RAM Enable Input
  SSRA => SSRA,    -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,    -- Port B Synchronous Set/Reset Input
  WEA => WEA,      -- Port A Write Enable Input
  WEB => WEB       -- Port B Write Enable Input
);

-- End of RAMB16_S4_S4_inst instantiation

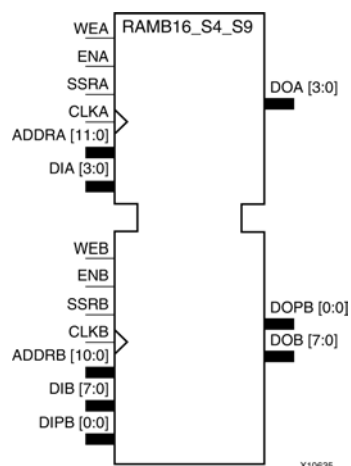
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S4\_S9

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit and 9-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号

INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_A = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、data <sup>3</sup>	変化なし <sup>1</sup> 、RAM(addr) <sup>2</sup> 、pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号  
INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。  
SRVAL\_B = レジスタの値  
addr = RAM アドレス  
RAM(addr) = アドレス ADDR の RAM の内容  
data = RAM の入力データ  
pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S4_S9	4096 x 4	—	(11:0)	(3:0)	—	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDRA) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRA) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マップ

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																			
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0			
4	4096	<--	7				6				5				4				3				2				1				0					
8	2048	<--	3								2								1								0									
16	1024	<--	1																0																	
32	512	<--	0																																	



## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリ セルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロックの SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリ セルにアクセスできます。ただし、一方のポートがあるメモリ セルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリ セルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット(0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット(0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S4_S9: 4k/2k x 4/8 + 0/1 Parity bit Dual-Port RAM
--               Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S4_S9_inst : RAMB16_S4_S9
generic map (
  INIT_A => X"0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"0", -- Port A output value upon SSR assertion
  SRVAL_B => X"000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 1023, Port B Address 0 to 511
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 1024 to 2047, Port B Address 512 to 1023
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 2048 to 3071, Port B Address 1024 to 1535
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 3072 to 4095, Port B Address 1536 to 2047
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Port B Address 0 to 511
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 512 to 1023
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 1024 to 1535
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 1536 to 2047
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA, -- Port A 4-bit Data Output
  DOB => DOB, -- Port B 8-bit Data Output
  DOPB => DOPB, -- Port B 1-bit Parity Output
  ADDRA => ADDRA, -- Port A 12-bit Address Input
  ADDRb => ADDRb, -- Port B 11-bit Address Input
  CLKA => CLKA, -- Port A Clock
  CLKb => CLKb, -- Port B Clock
  DIA => DIA, -- Port A 4-bit Data Input
  DIB => DIB, -- Port B 8-bit Data Input
  DIPB => DIPB, -- Port B 1-bit parity Input
  ENA => ENA, -- Port A RAM Enable Input
  ENB => ENB, -- Port B RAM Enable Input
  SSRA => SSRA, -- Port A Synchronous Set/Reset Input
  SSRb => SSRb, -- Port B Synchronous Set/Reset Input
  WEA => WEA, -- Port A Write Enable Input
  WEB => WEB -- Port B Write Enable Input
);

-- End of RAMB16_S4_S9_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S4_S9: 4k/2k x 4/8 + 0/1 Parity bit Dual-Port RAM
-- Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S4_S9_inst : RAMB16_S4_S9
generic map (
  INIT_A => X"0", -- Value of output RAM registers on Port A at startup
  INIT_B => X"000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"0", -- Port A output value upon SSR assertion
  SRVAL_B => X"000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 1023, Port B Address 0 to 511
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

439

```
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port B Address 1536 to 2047
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,    -- Port A 4-bit Data Output
  DOB => DOB,    -- Port B 8-bit Data Output
  DOPB => DOPB,  -- Port B 1-bit Parity Output
  ADDRA => ADDRA, -- Port A 12-bit Address Input
  ADDRb => ADDRb, -- Port B 11-bit Address Input
  CLKA => CLKA,   -- Port A Clock
  CLKB => CLKB,   -- Port B Clock
  DIA => DIA,     -- Port A 4-bit Data Input
  DIB => DIB,     -- Port B 8-bit Data Input
  DIPB => DIPB,   -- Port-B 1-bit parity Input
  ENA => ENA,     -- Port A RAM Enable Input
  ENB => ENB,     -- PortB RAM Enable Input
  SSRA => SSRA,   -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,   -- Port B Synchronous Set/Reset Input
  WEA => WEA,     -- Port A Write Enable Input
  WEB => WEB      -- Port B Write Enable Input
);

-- End of RAMB16_S4_S9_inst instantiation
```

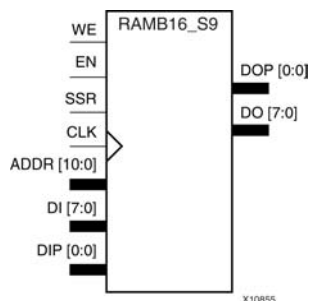
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## RAMB16\_S9

: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 9-bit Port



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた専用 RAM ブロックです。ブロック RAM のポートには 16384 ビットのデータ メモリがあります。セルの構成を、次の表に示します。

データ セル		パリティ セル				
ワード数	幅	ワード数	幅	アドレス バス	データ バス	パリティ バス
2048	8	2048	1	(10:0)	(7:0)	(0:0)

イネーブルピン EN は、ポートの読み出し、書き込み、リセットを制御します。EN が Low の場合、データは書き込まれず、出力 (DO および DOP) は変化しません。EN とリセット (SSR) が High の場合、クロック (CLK) が Low から High に切り替わるときに DO および DOP が SRVAL にセットされます。ライト イネーブル (WE) も High の場合は、DI および DIP の値が RAM に書き込まれます。EN が High で SSR と WE が Low の場合、クロックが Low から High に切り替わるときに、RAM アドレス (ADDR) に格納されているデータが読み出されます。SSR が Low で EN と WE が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDR) で選択されているワードにデータ入力 (DI および DIP) の値が読み込まれます。データ出力 (DO および DOP) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

前述の説明では、制御ピン (EN、WE、SSR、CLK) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

## 論理表

入力								出力			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT	INIT	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR=グローバル セットリセット信号

INIT = データ メモリ用に INIT 属性で設定された値。デフォルトはすべて 0 です。

SRVAL = SRVAL 属性で設定された SSR のアサート後の値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE=NO\_CHANGE

<sup>2</sup> WRITE\_MODE=READ\_FIRST

<sup>3</sup> WRITE\_MODE=WRITE\_FIRST

## 初期化

## メモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16 の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## 出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。シングルポート RAMB16 の出力レジスタの初期化に使用する属性には、INIT と SRVAL の 2 種類があります。INIT 属性は、電源投入時の出力レジスタの値を指定します。SRVAL 属性は、SSR (セット/リセット) 入力をアサートしたときの値を指定します。

INIT および SRVAL 属性は、初期化値を、出力ポートの各ビットを 1 ビット含む 16 進数文字列として指定します。たとえば、ポート幅が 1 の RAMB16\_S1 の場合、出力レジスタは 1 ビットなので、INIT および SRVAL には 1 または 0 を指定します。ポート幅が 4 の RAMB16\_S4 の場合は、出力レジスタが 4 ビットなので、0 ~ F の 16 進数値を指定します。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT または SRVAL の値の上位ビットで指定します。

#### 書き込みモードの選択

WRITE\_MODE 属性は、RAMB16 メモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE は WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。WRITE\_MODE を READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。WRITE\_MODE を NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	2 進数/ 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DO 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_00 ~ INIT_3F	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INITP_00 ~ INITP_07	2 進数/ 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定
SRVAL	2 進数/ 16 進数	任意の値	すべてゼロ	SSR ピンをアサートすると、DO 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
WRITE_MODE	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DO ポートで実行される時のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S9: 2k x 8 + 1 Parity bit Single-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S9_inst : RAMB16_S9
generic map (
  INIT => X"000", -- Value of output RAM registers at startup
  SRVAL => X"000", -- Output value upon SSR assertion
  WRITE_MODE => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 511
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 512 to 1023
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 1024 to 1535
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 1536 to 2047
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
```

```

INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Address 0 to 511
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 512 to 1023
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 1024 to 1535
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 1536 to 2047
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
DO => DO,      -- 8-bit Data Output
DOP => DOP,    -- 1-bit parity Output
ADDR => ADDR,  -- 11-bit Address Input
CLK => CLK,    -- Clock
DI => DI,      -- 8-bit Data Input
DIP => DIP,    -- 1-bit parity Input
EN => EN,      -- RAM Enable Input
SSR => SSR,    -- Synchronous Set/Reset Input
WE => WE       -- Write Enable Input
);

-- End of RAMB16_S9_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S9: 2k x 8 + 1 Parity bit Single-Port RAM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S9_inst : RAMB16_S9
generic map (
  INIT => X"000", -- Value of output RAM registers at startup
  SRVAL => X"000", -- Output value upon SSR assertion
  WRITE_MODE => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 511
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",

```

Spartan-3E ライブラリ ガイド (HDL 用)  
UG617 (v12.2) 2010 年 7 月 23 日

```
DIP => DIP,      -- 1-bit parity Input
EN  => EN,       -- RAM Enable Input
SSR => SSR,      -- Synchronous Set/Reset Input
WE  => WE        -- Write Enable Input
);

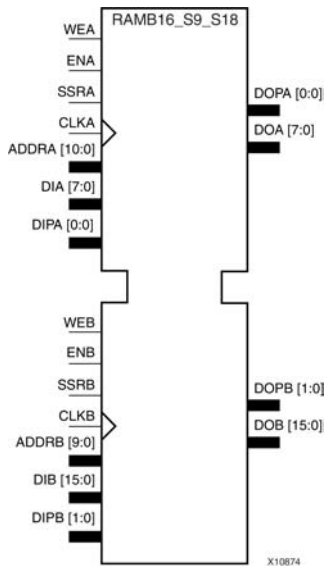
-- End of RAMB16_S9_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S9\_S18

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 9-bit and 18-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。



## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号  
 INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。  
 SRVAL\_A = レジスタの値  
 addr = RAM アドレス  
 RAM(addr) = アドレス ADDR の RAM の内容  
 data = RAM の入力データ  
 pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号

INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_B = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
コンポーネント	データセル (a)	パリティセル (a)	アドレスバス	データバス	パリティバス	データセル (a)	パリティセル (a)	アドレスバス	データバス	パリティバス
RAMB16_S9_S18	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マッピング

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティ メモリ セルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<-----	3				2				1				0
2	1024	<-----	1								0				
4	512	<-----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリセルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロック SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリセルにアクセスできます。ただし、一方のポートがあるメモリセルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリセルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S9_S18: 2k/1k x 8/16 + 1/2 Parity bits Parity bits Dual-Port RAM
--                Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S9_S18_inst : RAMB16_S9_S18
generic map (
  INIT_A => X"000", -- Value of output RAM registers on Port A at startup
  INIT_B => X"00000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"000", -- Port A output value upon SSR assertion
  SRVAL_B => X"00000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 511, Port B Address 0 to 255
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 512 to 1023, Port B Address 256 to 511
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 1024 to 1535, Port B Address 512 to 767
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Port A Address 1536 to 2047, Port B Address 768 to 1024
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```





Spartan-3E ライブラリ ガイド (HDL 用)  
UG617 (v12.2) 2010 年 7 月 23 日

```

INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 1536 to 2047, Port B Address 768 to 1024
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 8-bit Data Output
  DOB => DOB,      -- Port B 16-bit Data Output
  DOPA => DOPA,     -- Port A 1-bit Parity Output
  DOPB => DOPB,     -- Port B 2-bit Parity Output
  ADDRA => ADDRA,   -- Port A 11-bit Address Input
  ADDRb => ADDRb,   -- Port B 10-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,       -- Port A 8-bit Data Input
  DIB => DIB,       -- Port B 16-bit Data Input
  DIPa => DIPa,     -- Port A 1-bit parity Input
  DIPB => DIPB,     -- Port B 2-bit parity Input
  ENA => ENA,       -- Port A RAM Enable Input
  ENB => ENB,       -- Port B RAM Enable Input
  SSRA => SSRA,     -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,     -- Port B Synchronous Set/Reset Input
  WEA => WEA,       -- Port A Write Enable Input
  WEB => WEB        -- Port B Write Enable Input
);

-- End of RAMB16_S9_S18_inst instantiation

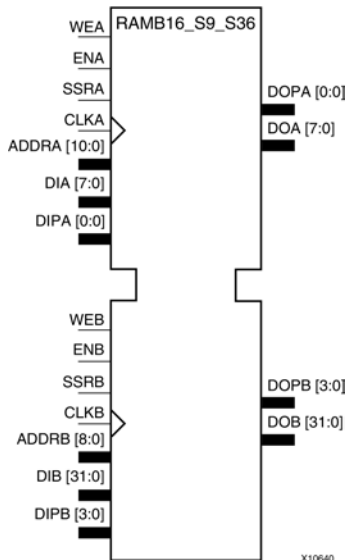
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S9\_S36

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 9-bit and 36-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。

## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号  
 INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。  
 SRVAL\_A = レジスタの値  
 addr = RAM アドレス  
 RAM(addr) = アドレス ADDR の RAM の内容  
 data = RAM の入力データ  
 pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号

INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_B = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S9_S36	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるときに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるときに、RAM アドレス (ADDRA) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDRA) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるときに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるときに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるときに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA、WEA、SSRA、CLKA、ENB、WEB、SSRB、CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マッピング

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																

パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<-----	3				2				1				0
2	1024	<-----	1								0				
4	512	<-----	0												

デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリ セルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロック SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリ セルにアクセスできます。ただし、一方のポートがあるメモリ セルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリ セルに対して書き込みまたは読み出しを実行しないようにする必要があります。



次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化なし	X	変化なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	変化なし	変化なし	変化なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行されるときポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

Spartan-3E ライブラリ ガイド (HDL 用)  
UG617 (v12.2) 2010 年 7 月 23 日

```

INIT_35 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_36 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_37 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_38 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_39 => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3A => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3B => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3C => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3D => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3E => X"0000000000000000000000000000000000000000000000000000000000000000",
INIT_3F => X"0000000000000000000000000000000000000000000000000000000000000000",
-- The next set of INITP_xx are for the parity bits
-- Port A Address 0 to 511, Port B Address 0 to 127
INITP_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 512 to 1023, Port B Address 128 to 255
INITP_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 1024 to 1535, Port B Address 256 to 383
INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 1536 to 2047, Port B Address 384 to 511
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 8-bit Data Output
  DOB => DOB,      -- Port B 32-bit Data Output
  DOPA => DOPA,     -- Port A 1-bit Parity Output
  DOPB => DOPB,     -- Port B 4-bit Parity Output
  ADDRA => ADDRA,   -- Port A 11-bit Address Input
  ADDR8 => ADDR8,   -- Port B 9-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,       -- Port A 8-bit Data Input
  DIB => DIB,       -- Port B 32-bit Data Input
  DIPA => DIPA,     -- Port A 1-bit parity Input
  DIPB => DIPB,     -- Port B 4-bit parity Input
  ENA => ENA,       -- Port A RAM Enable Input
  ENB => ENB,       -- Port B RAM Enable Input
  SSRA => SSRA,     -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,     -- Port B Synchronous Set/Reset Input
  WEA => WEA,       -- Port A Write Enable Input
  WEB => WEB        -- Port B Write Enable Input
);

-- End of RAMB16_S9_S36_inst instantiation

```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- RAMB16_S9_S36: 2k/512 x 8/32 + 1/4 Parity bits Parity bits Dual-Port RAM
-- Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S9_S36_inst : RAMB16_S9_S36
generic map (
  INIT_A => X"000", -- Value of output RAM registers on Port A at startup
  INIT_B => X"00000000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"000", -- Port A output value upon SSR assertion
  SRVAL_B => X"00000000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Port A Address 0 to 511, Port B Address 0 to 127
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",

```

Spartan-3E ライブラリ ガイド (HDL 用)  
UG617 (v12.2) 2010 年 7 月 23 日

```

INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Port A Address 1536 to 2047, Port B Address 384 to 511
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 8-bit Data Output
  DOB => DOB,      -- Port B 32-bit Data Output
  DOPA => DOPA,     -- Port A 1-bit Parity Output
  DOPB => DOPB,     -- Port B 4-bit Parity Output
  ADDRA => ADDRA,   -- Port A 11-bit Address Input
  ADDRb => ADDRb,   -- Port B 9-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,       -- Port A 8-bit Data Input
  DIB => DIB,       -- Port B 32-bit Data Input
  DIPa => DIPa,     -- Port A 1-bit parity Input
  DIPB => DIPB,     -- Port B 4-bit parity Input
  ENA => ENA,       -- Port A RAM Enable Input
  ENB => ENB,       -- Port B RAM Enable Input
  SSRA => SSRA,     -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,     -- Port B Synchronous Set/Reset Input
  WEA => WEA,       -- Port A Write Enable Input
  WEB => WEB        -- Port B Write Enable Input
);

-- End of RAMB16_S9_S36_inst instantiation

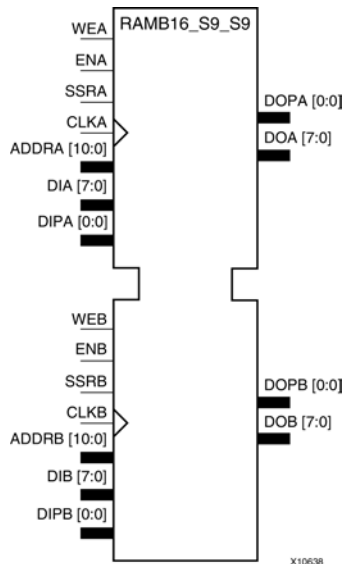
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## RAMB16\_S9\_S9

: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 9-bit Ports



### 概要

このデザイン エLEMENTは、同期書き込み機能を備えた、デュアル ポートの専用 RAM ブロックです。各ブロック RAM のポートには 16384 ビットのデータ メモリがあります。ポート幅が 9、18、36 ビットにコンフィギュレーションされたポートには、さらに 2048 ビットのパリティ メモリがあります。各ポートは、同じ 16384 個のデータ メモリ セルに独立してアクセスします。各ポートのデータ幅は、それぞれ個別に設定できます。このELEMENTのポートおよびセルの構成は「ポートの説明」の表に示されています。



## 論理表

論理表 A

入力								出力			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セットリセット信号  
 INIT\_A = 出力レジスタ用に INIT\_A 属性で設定された値。デフォルトはすべて 0 です。  
 SRVAL\_A = レジスタの値  
 addr = RAM アドレス  
 RAM(addr) = アドレス ADDR の RAM の内容  
 data = RAM の入力データ  
 pdata = RAM のパリティ データ  
<sup>1</sup> WRITE\_MODE\_A=NO\_CHANGE  
<sup>2</sup> WRITE\_MODE\_A=READ\_FIRST  
<sup>3</sup> WRITE\_MODE\_A=WRITE\_FIRST

論理表 B

入力								出力			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM の内容	
										データ RAM	パリティ RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	変化なし	変化なし
0	0	X	X	X	X	X	X	変化なし	変化なし	変化なし	変化なし
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	変化なし	変化なし
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	変化なし	変化なし
0	1	0	1	↑	addr	data	pdata	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 data <sup>3</sup>	変化なし <sup>1</sup> 、 RAM(addr) <sup>2</sup> 、 pdata <sup>3</sup>	RAM(addr) =>data	RAM(addr) =>pdata

GSR = グローバル セット リセット信号

INIT\_B = 出力レジスタ用に INIT\_B 属性で設定された値。デフォルトはすべて 0 です。

SRVAL\_B = レジスタの値

addr = RAM アドレス

RAM(addr) = アドレス ADDR の RAM の内容

data = RAM の入力データ

pdata = RAM のパリティ データ

<sup>1</sup> WRITE\_MODE\_B=NO\_CHANGE

<sup>2</sup> WRITE\_MODE\_B=READ\_FIRST

<sup>3</sup> WRITE\_MODE\_B=WRITE\_FIRST

## ポートの説明

ポート A						ポート B				
デザイン エレメント	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス	データ セル (a)	パリティ セル (a)	アドレス バス	データ バス	パリティ バス
RAMB16_S9_S9	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)

(a) ワード数 X 幅

各ポートは、それぞれのクロックに完全に同期します。ポート A の各データ入力ピン (DIA) のセットアップ タイム、およびデータ出力バス (DOA) の clock-to-out タイムは、CLKA を基準とします。ポート B の各データ入力ピン (DIB) のセットアップ タイム、およびデータ出力バス (DOB) の clock-to-out タイムは、CLKB を基準とします。イネーブル ピン ENA は、ポート A の読み出し、書き込み、リセットを制御します。ENA が Low の場合、データは書き込まれず、出力 (DOA および DOPA) は変化しません。ENA とリセット (SSRA) が High の場合、クロック (CLKA) が Low から High に切り替わるたびに DOA および DOPA が SRVAL\_A にセットされます。ライト イネーブル (WEA) も High の場合は、DIA および DIPA の値が RAM に書き込まれます。ENA が High で SSRA と WEA が Low の場合、クロックが Low から High に切り替わるたびに、RAM アドレス (ADDR\_A) に格納されているデータが読み出されます。SSRA が Low で ENA と WEA が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDR\_A) で選択されているワードにデータ入力 (DIA および DIPA) の値が読み込まれます。データ出力 (DOA および DOPA) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。

イネーブル ピン ENB は、ポート B の読み出し、書き込み、リセットを制御します。ENB が Low の場合、データは書き込まれず、出力 (DOB および DOPB) は変化しません。ENB とリセット (SSRB) が High の場合、クロック (CLKB) が Low から High に切り替わるたびに、DOB および DOPB が SRVAL\_B にセットされます。ライト イネーブル (WEB) も High の場合は、DIB および DIPB の値が RAM に書き込まれます。ENB が High で SSRB と WEB が Low の場合、クロックが Low から High に切り替わるたびに RAM アドレス (ADDRB) に格納されているデータが出力されます。SSRB が Low で ENB と WEB が High の場合、クロックが Low から High に切り替わるたびに、書き込みアドレス (ADDRB) で選択されているワードにデータ入力 (DIB および DIPB) の値が読み込まれます。データ出力 (DOB および DOPB) に出力される値は、書き込みモードによって異なります。書き込みモードは、デフォルトでは WRITE\_MODE=WRITE\_FIRST に設定されています。前述の説明では、制御ピン (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, CLKB) がアクティブ High であると想定していますが、ポートにインバータを配置してアクティブ Low にすることもできます。RAMB16 のポートに配置したインバータはブロック内に組み込まれるので、CLB リソースは使用されません。

#### アドレス マップ

各ポートは、ポートの幅によって異なるアドレス指定方法を使用して、同じ 18432 個のメモリ セルにアクセスします。「データ用のポート アドレス マップ」に示すように、すべてのポート幅で 16384 個のメモリ セルをデータの格納に使用できます。9、18、および 36 ビット幅のポートには、「パリティ用のポート アドレス マップ」に示すように、2408 個のパリティメモリセルもあります。特定のポート幅での物理的な RAM の位置は、次の式によって決定されます。

$$\text{Start} = ((\text{ADDR\_port} + 1) * (\text{Widthport})) - 1$$

$$\text{End} = (\text{ADDRport}) * (\text{Widthport})$$

次の表に、デュアル ポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

次の表に、各ポート幅のアドレス マップを示します。

データ用のポート アドレス マップ

データ 幅	ポートのデータ アドレス																																		
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0		
4	4096	<--	7				6				5				4				3				2				1				0				
8	2048	<--	3								2								1								0								
16	1024	<--	1																0																
32	512	<--	0																																

## パリティ用のポート アドレス マップ

パリティ幅	ポートのパリティ アドレス														
1	2048	<----	3				2				1				0
2	1024	<----	1								0				
4	512	<----	0												

## デュアル ポート RAMB16 のメモリ内容の初期化

INIT\_xx 属性を使用すると、デバイスのコンフィギュレーション中に RAMB16 のメモリの内容を初期化できます。各 RAMB16\_Sm\_Sn の初期値は、64 種類の初期化属性 (INIT\_00 ~ INIT\_3F) で指定します。各属性は、64 の 16 進数値から成り、合計 16384 ビットを指定できます。

INITP\_xx 属性を使用すると、デバイスのコンフィギュレーションまたはアサート中にパリティ メモリを初期化できます。9、18、36 ビット幅にコンフィギュレーションされたポートのパリティ メモリの初期値は、8 個の初期化属性 (INITP\_00 ~ INITP\_07) で指定します。各属性は 64 個の 16 進数値から成り、合計 2048 ビットを指定できます。

INIT\_xx または INITP\_xx 属性を指定しない場合は、そのアドレスの値は 0 に設定されます。属性を一部だけ指定すると、上位ビットが 0 になります。

## デュアル ポート RAMB16 の出力レジスタの初期化

Spartan®-3A 以降のデバイスでは、電源投入時 (GSR が High のとき) に出力レジスタの各ビットを 0 または 1 に初期化できます。また、セット/リセットをアサートした後の値を、電源投入時の初期値とは異なる値に設定できます。デュアル ポート RAMB16 の出力レジスタを初期化する属性には、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の 4 種類があります。INIT\_A 属性は電源投入時のポート A の出力レジスタの初期値を指定し、INIT\_B 属性は電源投入時のポート B の出力レジスタの初期値を指定します。SRVAL\_A 属性はポート A で SSRA (セット/リセット) 入力をアサートしたときの初期値を指定し、SRVAL\_B 属性はポート B で SSRB (セット/リセット) 入力をアサートしたときの初期値を指定します。

INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B 属性は、16 進数で指定します。この初期値はポート幅によって異なります。たとえば、ポート A の幅が 1、ポート B の幅が 4 の RAMB16\_S1\_S4 の場合、ポート A の出力レジスタは 1 ビットなので、INIT\_A または SRVAL\_A には 1 か 0 しか指定できません。ポート B の出力レジスタは 4 ビットなので、INIT\_B または SRVAL\_B に 0 ~ F の 16 進数値を指定できます。

パリティビットを含むポートでは、出力レジスタのパリティ部分は、INIT\_A、INIT\_B、SRVAL\_A、SRVAL\_B の値の上位ビットで指定します。

INIT および SRVAL 属性を指定しない場合は、デフォルトで 0 に初期化されます。

## 書き込みモードの選択

WRITE\_MODE\_A 属性は、デュアル ポート RAMB16 のポート A のメモリおよび出力の内容を制御し、WRITE\_MODE\_B 属性は、ポート B のメモリおよび出力の内容を制御します。デフォルトでは、WRITE\_MODE\_A と WRITE\_MODE\_B は両方とも WRITE\_FIRST に設定されています。この場合、入力値がメモリに書き込まれた後にその値が出力されます。READ\_FIRST に設定すると、メモリの内容が出力された後、入力値がメモリに書き込まれます。NO\_CHANGE に設定すると、入力値はメモリに書き込まれますが、出力の値は変化しません。ポート A とポート B で同じメモリ セルに読み出し/書き込みを行おうとした場合の競合の解決方法については、「ポート A とポート B が競合する場合の対処方法」を参照してください。

## ポート A とポート B が競合する場合の対処方法

Spartan-3A ブロック SelectRAM™ は、完全なデュアル ポート RAM で、2 つのポートが同時に同じメモリ セルにアクセスできます。ただし、一方のポートがあるメモリ セルに書き込みを行っている場合は、もう一方のポートで clock-to-clock セットアップ タイム内に、そのメモリ セルに対して書き込みまたは読み出しを実行しないようにする必要があります。

次の表に、デュアルポート RAMB16 で競合が発生した場合の動作を、WRITE\_MODE\_A と WRITE\_MODE\_B の設定別に示します。

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=NO\_CHANGE の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	変化 なし	X	変化 なし	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	変化 なし	変化 なし	変化 なし	X	X

WRITE\_MODE\_A=READ\_FIRST、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE\_MODE\_A=WRITE\_FIRST、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=READ\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化 なし	X	変化 なし	X	DIB	DIPB

WRITE\_MODE\_A=NO\_CHANGE、WRITE\_MODE\_B=WRITE\_FIRST の場合

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	変化なし	X	変化なし	X	X	X

WRITE\_MODE\_A=READ\_FIRST および WRITE\_MODE\_B=WRITE\_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	データ RAM	パリティ RAM
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	変化なし	変化なし
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT_00 ~ INIT_3F	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のデータ部分の初期値を指定
INIT_A	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INIT_B	2 進数 / 16 進数	任意の値	すべてゼロ	コンフィギュレーション後の DOA/DOB 出力ポートの初期値を指定。ビット幅は、RAM の A ポートまたは B ポートの幅によって決まります。
INITP_00 ~ INITP_07	2 進数 / 16 進数	任意の値	すべてゼロ	RAM 配列のパリティ部分の初期値を指定

属性	タイプ	値	デフォルト	説明
SIM_COLLISION_CHECK	文字列	ALL、 WARNING_ONLY、 GENERATE_X_ONLY、 NONE	ALL	<p>メモリの競合が発生した場合にシミュレーションの動作を変更できます。詳細は次のとおりです。</p> <ul style="list-style-type: none"> <li>ALL に設定すると、警告メッセージが出力され、関連する出力およびメモリの値が不定 (X) になります。</li> <li>WARNING_ONLY に設定すると、警告メッセージのみが出力され、関連する出力およびメモリの値はそのまま保持されます。</li> <li>GENERATE_X_ONLY に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値が不定 (X) になります。</li> <li>NONE に設定すると、警告メッセージは出力されず、関連する出力およびメモリの値はそのまま保持されます。</li> </ul> <p><b>メモ：</b> ALL 以外の値に設定すると、シミュレーション中にデザインの問題を認識できなくなるため、この値を変更する場合は注意が必要です。詳細は、『合成/シミュレーション デザイン ガイド』を参照してください。</p>
SRVAL_A	2 進数/ 16 進数	任意の値	すべてゼロ	RSTA ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の A ポートの幅によって決まります。
SRVAL_B	2 進数/ 16 進数	任意の値	すべてゼロ	RSTB ピンをアサートすると、DOA/DOB 出力ポートがセット (1) またはリセット (0) に設定されるよう指定できます。ビット幅は、RAM の B ポートの幅によって決まります。
WRITE_MODE_A	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。
WRITE_MODE_B	文字列	WRITE_FIRST、 READ_FIRST、 NO_CHANGE	WRITE_FIRST	書き込みコマンドが DOA/DOB ポートで実行される際のポートの動作を指定します。WRITE_FIRST に設定すると、値がポートに書き込まれてから出力されます。READ_FIRST に設定すると、新たに値が書き込まれる前に、RAM の以前の値が出力ポートに出力されます。NO_CHANGE に設定すると、出力ポートの以前の値が維持され、出力ポートの値が変化しません。RAM のポートから値を読み出さない場合は、このモードに設定することをお勧めします。

## VHDL 記述 (インスタンスエーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
```

```
use UNISIM.vcomponents.all;

-- RAMB16_S9_S9: 2k x 8 + 1 Parity bit Dual-Port RAM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

RAMB16_S9_S9_inst : RAMB16_S9_S9
generic map (
  INIT_A => X"000", -- Value of output RAM registers on Port A at startup
  INIT_B => X"000", -- Value of output RAM registers on Port B at startup
  SRVAL_A => X"000", -- Port A output value upon SSR assertion
  SRVAL_B => X"000", -- Port B output value upon SSR assertion
  WRITE_MODE_A => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  WRITE_MODE_B => "WRITE_FIRST", -- WRITE_FIRST, READ_FIRST or NO_CHANGE
  SIM_COLLISION_CHECK => "ALL", -- "NONE", "WARNING", "GENERATE_X_ONLY", "ALL"
  -- The following INIT_xx declarations specify the initial contents of the RAM
  -- Address 0 to 511
  INIT_00 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_01 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_02 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_03 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_07 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_08 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_09 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_0F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 512 to 1023
  INIT_10 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_11 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_12 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_13 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_14 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_15 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_16 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_17 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_18 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_19 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_1F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 1024 to 1535
  INIT_20 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_21 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_22 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_23 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_24 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_25 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_26 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_27 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_28 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_29 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2A => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2B => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2C => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2D => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2E => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_2F => X"0000000000000000000000000000000000000000000000000000000000000000",
  -- Address 1536 to 2047
  INIT_30 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_31 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_32 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_33 => X"0000000000000000000000000000000000000000000000000000000000000000",
  INIT_34 => X"0000000000000000000000000000000000000000000000000000000000000000",
```





Spartan-3E ライブラリ ガイド (HDL 用)  
UG617 (v12.2) 2010 年 7 月 23 日

```

INITP_04 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_05 => X"0000000000000000000000000000000000000000000000000000000000000000",
-- Address 1536 to 2047
INITP_06 => X"0000000000000000000000000000000000000000000000000000000000000000",
INITP_07 => X"0000000000000000000000000000000000000000000000000000000000000000")
port map (
  DOA => DOA,      -- Port A 8-bit Data Output
  DOB => DOB,      -- Port B 8-bit Data Output
  DOPA => DOPA,     -- Port A 1-bit Parity Output
  DOPB => DOPB,     -- Port B 1-bit Parity Output
  ADDRA => ADDRA,   -- Port A 11-bit Address Input
  ADDRb => ADDRb,   -- Port B 11-bit Address Input
  CLKA => CLKA,     -- Port A Clock
  CLKB => CLKB,     -- Port B Clock
  DIA => DIA,       -- Port A 8-bit Data Input
  DIB => DIB,       -- Port B 8-bit Data Input
  DIPa => DIPa,     -- Port A 1-bit parity Input
  DIPB => DIPB,     -- Port B 1-bit parity Input
  ENA => ENA,       -- Port A RAM Enable Input
  ENB => ENB,       -- Port B RAM Enable Input
  SSRA => SSRA,     -- Port A Synchronous Set/Reset Input
  SSRB => SSRB,     -- Port B Synchronous Set/Reset Input
  WEA => WEA,       -- Port A Write Enable Input
  WEB => WEB        -- Port B Write Enable Input
);

-- End of RAMB16_S9_S9_inst instantiation

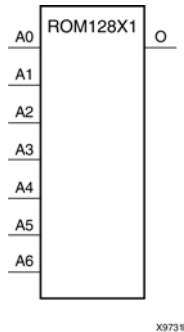
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## ROM128X1

: 128-Deep by 1-Wide ROM



X9731

### 概要

このデザイン エLEMENTは 128 ワード X 1 ビットの ROM です。データ出力 (O) には、7 ビットのアドレス (A6 ~ A0) で選択されたワードが出力されます。ROM は、コンフィギュレーションの際に INIT=value で指定した値に初期化されます。初期値は 32 桁の 16 進数で、ROM には最上位ビット A=FH から最下位ビット A=0H の順に書き込まれます。INIT=value を指定しないと、エラーになります。

### 論理表

入力				出力
I0	I1	I2	I3	O
0	0	0	0	INIT(0)
0	0	0	1	INIT(1)
0	0	1	0	INIT(2)
0	0	1	1	INIT(3)
0	1	0	0	INIT(4)
0	1	0	1	INIT(5)
0	1	1	0	INIT(6)
0	1	1	1	INIT(7)
1	0	0	0	INIT(8)
1	0	0	1	INIT(9)
1	0	1	0	INIT(10)
1	0	1	1	INIT(11)
1	1	0	0	INIT(12)
1	1	0	1	INIT(13)
1	1	1	0	INIT(14)
1	1	1	1	INIT(15)

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	128 ビット値	すべてゼロ	ROM の値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- ROM128X1: 128 x 1 Asynchronous Distributed (LUT) ROM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

ROM128X1_inst : ROM128X1
generic map (
  INIT => X"00000000000000000000000000000000"
)
port map (
  O => O,    -- ROM output
  A0 => A0,   -- ROM address[0]
  A1 => A1,   -- ROM address[1]
  A2 => A2,   -- ROM address[2]
  A3 => A3,   -- ROM address[3]
  A4 => A4,   -- ROM address[4]
  A5 => A5,   -- ROM address[5]
  A6 => A6    -- ROM address[6]
);

-- End of ROM128X1_inst instantiation
```

## Verilog 記述（インスタンス化）

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- ROM128X1: 128 x 1 Asynchronous Distributed (LUT) ROM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

ROM128X1_inst : ROM128X1
generic map (
  INIT => X"00000000000000000000000000000000")
port map (
  O => O,    -- ROM output
  A0 => A0,  -- ROM address[0]
  A1 => A1,  -- ROM address[1]
  A2 => A2,  -- ROM address[2]
  A3 => A3,  -- ROM address[3]
  A4 => A4,  -- ROM address[4]
  A5 => A5,  -- ROM address[5]
  A6 => A6   -- ROM address[6]
);

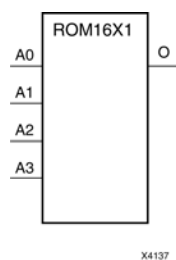
-- End of ROM128X1_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## ROM16X1

: 16-Deep by 1-Wide ROM



### 概要

このデザイン エLEMENTは 16 ワード X 1 ビットの ROM です。データ出力 (O) には、4 ビットのアドレス (A3 ~ A0) で選択されたワードが出力されます。ROM は、コンフィギュレーションの際に INIT=value で指定した値に初期化されます。初期値は 4 桁の 16 進数で、ROM には最上位ビット A=FH から最下位ビット A=0H の順に書き込まれます。たとえば、INIT=10A7 と指定すると、「0001 0000 1010 0111」というデータストリームが生成されます。INIT=value を指定しないと、エラーになります。

### 論理表

入力				出力
I0	I1	I2	I3	O
0	0	0	0	INIT(0)
0	0	0	1	INIT(1)
0	0	1	0	INIT(2)
0	0	1	1	INIT(3)
0	1	0	0	INIT(4)
0	1	0	1	INIT(5)
0	1	1	0	INIT(6)
0	1	1	1	INIT(7)
1	0	0	0	INIT(8)
1	0	0	1	INIT(9)
1	0	1	0	INIT(10)
1	0	1	1	INIT(11)
1	1	0	0	INIT(12)
1	1	0	1	INIT(13)
1	1	1	0	INIT(14)
1	1	1	1	INIT(15)

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	ROM の値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- ROM16X1: 16 x 1 Asynchronous Distributed (LUT) ROM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

ROM16X1_inst : ROM16X1
generic map (
  INIT => X"0000")
port map (
  O => O,    -- ROM output
  A0 => A0,  -- ROM address[0]
  A1 => A1,  -- ROM address[1]
  A2 => A2,  -- ROM address[2]
  A3 => A3   -- ROM address[3]
);

-- End of ROM16X1_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- ROM16X1: 16 x 1 Asynchronous Distributed (LUT) ROM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

ROM16X1_inst : ROM16X1
generic map (
  INIT => X"0000")
port map (
  O => O,    -- ROM output
  A0 => A0,  -- ROM address[0]
  A1 => A1,  -- ROM address[1]
  A2 => A2,  -- ROM address[2]
  A3 => A3   -- ROM address[3]
);

-- End of ROM16X1_inst instantiation
```

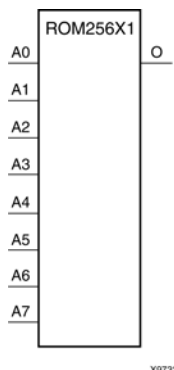


## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## ROM256X1

: 256-Deep by 1-Wide ROM



### 概要

このデザイン エLEMENTは 256 ワード X 1 ビットの ROM です。データ出力 (O) には、8 ビットのアドレス (A7 ~ A0) で選択されたワードが出力されます。ROM は、コンフィギュレーションの際に INIT=value で指定した値に初期化されます。初期値は 64 桁の 16 進数で、ROM には最上位ビット A=FH から最下位ビット A=0H の順に書き込まれます。

INIT=value を指定しないと、エラーになります。

### 論理表

入力				出力
I0	I1	I2	I3	O
0	0	0	0	INIT(0)
0	0	0	1	INIT(1)
0	0	1	0	INIT(2)
0	0	1	1	INIT(3)
0	1	0	0	INIT(4)
0	1	0	1	INIT(5)
0	1	1	0	INIT(6)
0	1	1	1	INIT(7)
1	0	0	0	INIT(8)
1	0	0	1	INIT(9)
1	0	1	0	INIT(10)
1	0	1	1	INIT(11)
1	1	0	0	INIT(12)
1	1	0	1	INIT(13)
1	1	1	0	INIT(14)
1	1	1	1	INIT(15)

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	256 ビット値	すべてゼロ	ROM の値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- ROM256X1: 256 x 1 Asynchronous Distributed (LUT) ROM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

ROM256X1_inst : ROM256X1
generic map (
  INIT => X"0000000000000000000000000000000000000000000000000000000000000000"
)
port map (
  O => O,    -- ROM output
  A0 => A0,  -- ROM address[0]
  A1 => A1,  -- ROM address[1]
  A2 => A2,  -- ROM address[2]
  A3 => A3,  -- ROM address[3]
  A4 => A4,  -- ROM address[4]
  A5 => A5,  -- ROM address[5]
  A6 => A6,  -- ROM address[6]
  A7 => A7,  -- ROM address[7]
);

-- End of ROM256X1_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- ROM256X1: 256 x 1 Asynchronous Distributed (LUT) ROM
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

ROM256X1_inst : ROM256X1
generic map (
  INIT => X"0000000000000000000000000000000000000000000000000000000000000000"
)
port map (
  O => O,    -- ROM output
  A0 => A0,   -- ROM address[0]
  A1 => A1,   -- ROM address[1]
  A2 => A2,   -- ROM address[2]
  A3 => A3,   -- ROM address[3]
  A4 => A4,   -- ROM address[4]
  A5 => A5,   -- ROM address[5]
  A6 => A6,   -- ROM address[6]
  A7 => A7    -- ROM address[7]
);

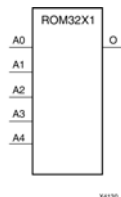
-- End of ROM256X1_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## ROM32X1

: 32-Deep by 1-Wide ROM



### 概要

このデザイン エLEMENTは 32 ワード X 1 ビットの ROM です。データ出力 (O) には、5 ビットのアドレス (A4 ~ A0) で選択されたワードが出力されます。ROM は、コンフィギュレーションの際に INIT=value で指定した値に初期化されます。初期値は 8 桁の 16 進数で、ROM には最上位ビット A=1FH から最下位ビット A=00H の順に書き込まれます。

たとえば、INIT=10A78F39 と指定すると、次のデータストリームが生成されます。0001 0000 1010 0111 1000 1111 0011 1001. INIT=value を指定しないと、エラーになります。

### 論理表

入力				出力
I0	I1	I2	I3	O
0	0	0	0	INIT(0)
0	0	0	1	INIT(1)
0	0	1	0	INIT(2)
0	0	1	1	INIT(3)
0	1	0	0	INIT(4)
0	1	0	1	INIT(5)
0	1	1	0	INIT(6)
0	1	1	1	INIT(7)
1	0	0	0	INIT(8)
1	0	0	1	INIT(9)
1	0	1	0	INIT(10)
1	0	1	1	INIT(11)
1	1	0	0	INIT(12)
1	1	0	1	INIT(13)
1	1	1	0	INIT(14)
1	1	1	1	INIT(15)

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	32 ビット値	すべてゼロ	ROM の値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- ROM32X1: 32 x 1 Asynchronous Distributed (LUT) ROM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

ROM32X1_inst : ROM32X1
generic map (
  INIT => X"00000000")
port map (
  O => O,    -- ROM output
  A0 => A0,  -- ROM address[0]
  A1 => A1,  -- ROM address[1]
  A2 => A2,  -- ROM address[2]
  A3 => A3,  -- ROM address[3]
  A4 => A4   -- ROM address[4]
);
-- End of ROM32X1_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- ROM32X1: 32 x 1 Asynchronous Distributed (LUT) ROM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

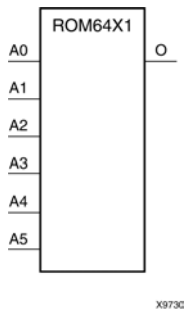
ROM32X1_inst : ROM32X1
generic map (
  INIT => X"00000000")
port map (
  O => O,    -- ROM output
  A0 => A0,  -- ROM address[0]
  A1 => A1,  -- ROM address[1]
  A2 => A2,  -- ROM address[2]
  A3 => A3,  -- ROM address[3]
  A4 => A4   -- ROM address[4]
);
-- End of ROM32X1_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリ データシート](#)

## ROM64X1

: 64-Deep by 1-Wide ROM



### 概要

このデザイン エLEMENTは 64 ワード X 1 ビットの ROM です。データ出力 (O) には、6 ビットのアドレス (A5 ～ A0) で選択されたワードが出力されます。ROM は、コンフィギュレーションの際に INIT=value で指定した値に初期化されます。初期値は 16 桁の 16 進数で、ROM には最上位ビット A=FH から最下位ビット A=0H の順に書き込まれます。INIT=value を指定しないと、エラーになります。

### 論理表

入力				出力
I0	I1	I2	I3	O
0	0	0	0	INIT(0)
0	0	0	1	INIT(1)
0	0	1	0	INIT(2)
0	0	1	1	INIT(3)
0	1	0	0	INIT(4)
0	1	0	1	INIT(5)
0	1	1	0	INIT(6)
0	1	1	1	INIT(7)
1	0	0	0	INIT(8)
1	0	0	1	INIT(9)
1	0	1	0	INIT(10)
1	0	1	1	INIT(11)
1	1	0	0	INIT(12)
1	1	0	1	INIT(13)
1	1	1	0	INIT(14)
1	1	1	1	INIT(15)



## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	64 ビット値	すべてゼロ	ROM の値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- ROM64X1: 64 x 1 Asynchronous Distributed (LUT) ROM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

ROM64X1_inst : ROM64X1
generic map (
  INIT => X"0000000000000000")
port map (
  O => O,    -- ROM output
  A0 => A0,  -- ROM address[0]
  A1 => A1,  -- ROM address[1]
  A2 => A2,  -- ROM address[2]
  A3 => A3,  -- ROM address[3]
  A4 => A4,  -- ROM address[4]
  A5 => A5   -- ROM address[5]
);

-- End of ROM64X1_inst instantiation
```

## Verilog 記述（インスタンス化）

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- ROM64X1: 64 x 1 Asynchronous Distributed (LUT) ROM
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

ROM64X1_inst : ROM64X1
generic map (
  INIT => X"0000000000000000")
port map (
  O => O,    -- ROM output
  A0 => A0,  -- ROM address[0]
  A1 => A1,  -- ROM address[1]
  A2 => A2,  -- ROM address[2]
  A3 => A3,  -- ROM address[3]
  A4 => A4,  -- ROM address[4]
  A5 => A5   -- ROM address[5]
);

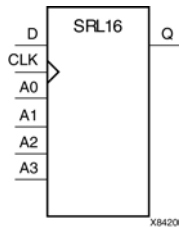
-- End of ROM64X1_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## SRL16

: 16-Bit Shift Register Look-Up Table (LUT)



### 概要

このデザイン エLEMENTは、シフトレジスタ ルックアップ テーブル (LUT) です。シフトレジスタの長さは、入力 A3、A2、A1、A0 の値によって決定されます。

シフトレジスタの長さは、固定することも、変動させることもできます。

- ・ **固定長のシフトレジスタを作成するには：** 入力 A3 ～ A0 の値を一定の値にします。シフトレジスタは 1 ～ 16 ビットの長さに設定できます。アドレス入力の値によるシフトレジスタの長さは、長さ =  $(8 \times A3) + (4 \times A2) + (2 \times A1) + A0 + 1$  という式で算出できます。A3、A2、A1、A0 がすべてゼロの場合 (0000) はシフトレジスタの長さは 1 ビットになり、すべて 1 の場合 (1111) は 16 ビットになります。
- ・ **シフトレジスタ長を動的に変化させるには：** 入力 A3 ～ A0 の値を変化させます。たとえば、A2、A1、A0 がすべて 1 の場合 (111) に A3 を 1 から 0 に切り替えると、シフトレジスタの長さは 16 ビットから 8 ビットに変化します。内部的には、シフトレジスタの長さは常に 16 ビットで、どのビットの値が出力されるかは入力 A3 ～ A0 の値によって決定されます。

シフトレジスタ LUT の初期値を指定するには、INIT 属性に 4 桁の 16 進数を割り当てます。一番左の桁が最上位ビットになります。INIT の値を指定しない場合は、シフトレジスタ LUT の内容はコンフィギュレーション中にゼロ (0000) にクリアされます。

クロック (CLK) が Low から High に切り替わるときに、D の値がシフトレジスタの第 1 ビットにロードされます。次にクロックが Low から High に切り替わると、シフトレジスタの値は次の高位ビットにシフトされ、新しい値がロードされます。アドレス入力の値によってシフトレジスタの長さが決まり、Q にその値が出力されます。

### 論理表

入力			出力
A <sub>m</sub>	CLK	D	Q
A <sub>m</sub>	X	X	Q(A <sub>m</sub> )
A <sub>m</sub>	↑	D	Q(A <sub>m</sub> - 1)
m = 0、1、2、3			

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	コンフィギュレーション後の Q 出力の初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRL16: 16-bit shift register LUT operating on posedge of clock
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRL16_inst : SRL16
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  A0 => A0,         -- Select[0] input
  A1 => A1,         -- Select[1] input
  A2 => A2,         -- Select[2] input
  A3 => A3,         -- Select[3] input
  CLK => CLK,       -- Clock input
  D => D            -- SRL data input
);

-- End of SRL16_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRL16: 16-bit shift register LUT operating on posedge of clock
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRL16_inst : SRL16
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  A0 => A0,         -- Select[0] input
  A1 => A1,         -- Select[1] input
  A2 => A2,         -- Select[2] input
  A3 => A3,         -- Select[3] input
  CLK => CLK,       -- Clock input
  D => D            -- SRL data input
);

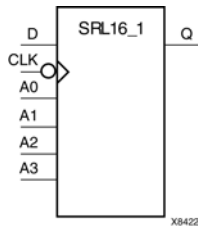
-- End of SRL16_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## SRL16\_1

: 16-Bit Shift Register Look-Up Table (LUT) with Negative-Edge Clock



## 概要

このデザイン エLEMENTは、シフトレジスタ ルックアップ テーブル (LUT) です。シフトレジスタの長さは、入力 A3、A2、A1、A0 の値によって決定されます。

シフトレジスタの長さは、固定することも、変動させることもできます。

- ・ **固定長のシフトレジスタを作成するには：** 入力 A3 ～ A0 の値を一定の値にします。シフトレジスタは 1 ～ 16 ビットの長さに設定できます。アドレス入力の値によるシフトレジスタの長さは、長さ =  $(8 \times A3) + (4 \times A2) + (2 \times A1) + A0 + 1$  という式で算出できます。A3、A2、A1、A0 がすべてゼロの場合 (0000) はシフトレジスタの長さは 1 ビットになり、すべて 1 の場合 (1111) は 16 ビットになります。
- ・ **シフトレジスタ長を動的に変化させるには：** 入力 A3 ～ A0 の値を変化させます。たとえば、A2、A1、A0 がすべて 1 の場合 (111) に A3 を 1 から 0 に切り替えると、シフトレジスタの長さは 16 ビットから 8 ビットに変化します。内部的には、シフトレジスタの長さは常に 16 ビットで、どのビットの値が出力されるかは入力 A3 ～ A0 の値によって決定されます。

シフトレジスタ LUT の初期値を指定するには、INIT 属性に 4 桁の 16 進数を割り当てます。一番左の桁が最上位ビットになります。INIT の値を指定しない場合は、シフトレジスタ LUT の内容はコンフィギュレーション中にゼロ (0000) にクリアされます。

クロック (CLK) が High から Low に切り替わるときに、D の値がシフトレジスタの第 1 ビットにロードされます。次にクロックが High から Low に切り替わると、シフトレジスタの値は次の高位ビットにシフトされ、新しい値がロードされます。アドレス入力の値によってシフトレジスタの長さが決まり、Q にその値が出力されます。

## 論理表

入力			出力
A <sub>m</sub>	CLK	D	Q
A <sub>m</sub>	X	X	Q(A <sub>m</sub> )
A <sub>m</sub>	↓	D	Q(A <sub>m</sub> - 1)
m = 0、1、2、3			

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	コンフィギュレーション後の Q 出力の初期値を指定

## VHDL 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRL16_1: 16-bit shift register LUT operating on negedge of clock
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRL16_1_inst : SRL16_1
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  A0 => A0,         -- Select[0] input
  A1 => A1,         -- Select[1] input
  A2 => A2,         -- Select[2] input
  A3 => A3,         -- Select[3] input
  CLK => CLK,       -- Clock input
  D => D            -- SRL data input
);

-- End of SRL16_1_inst instantiation
```

## Verilog 記述 (インスタンスレーション)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRL16_1: 16-bit shift register LUT operating on negedge of clock
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRL16_1_inst : SRL16_1
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  A0 => A0,         -- Select[0] input
  A1 => A1,         -- Select[1] input
  A2 => A2,         -- Select[2] input
  A3 => A3,         -- Select[3] input
  CLK => CLK,       -- Clock input
  D => D            -- SRL data input
);

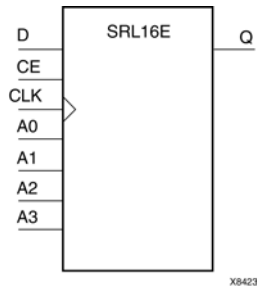
-- End of SRL16_1_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## SRL16E

: 16-Bit Shift Register Look-Up Table (LUT) with Clock Enable



### 概要

このデザイン エLEMENTは、シフトレジスタ ルックアップ テーブル (LUT) です。シフトレジスタの長さは、入力 A3、A2、A1、A0 の値によって決定されます。

シフトレジスタの長さは、固定することも、変動させることもできます。

- ・ **固定長のシフトレジスタを作成するには：** 入力 A3 ～ A0 の値を一定の値にします。シフトレジスタは 1 ～ 16 ビットの長さに設定できます。アドレス入力の値によるシフトレジスタの長さは、長さ =  $(8 \times A3) + (4 \times A2) + (2 \times A1) + A0 + 1$  という式で算出できます。A3、A2、A1、A0 がすべてゼロの場合 (0000) はシフトレジスタの長さは 1 ビットになり、すべて 1 の場合 (1111) は 16 ビットになります。
- ・ **シフトレジスタ長を動的に変化させるには：** 入力 A3 ～ A0 の値を変化させます。たとえば、A2、A1、A0 がすべて 1 の場合 (111) に A3 を 1 から 0 に切り替えると、シフトレジスタの長さは 16 ビットから 8 ビットに変化します。内部的には、シフトレジスタの長さは常に 16 ビットで、どのビットの値が出力されるかは入力 A3 ～ A0 の値によって決定されます。

シフトレジスタ LUT の初期値を指定するには、INIT 属性に 4 桁の 16 進数を割り当てます。一番左の桁が最上位ビットになります。INIT の値を指定しない場合は、シフトレジスタ LUT の内容はコンフィギュレーション中にゼロ (0000) にクリアされます。

CE が High の場合、クロック (CLK) が Low から High に切り替わるときに、D の値がシフトレジスタの第 1 ビットにロードされます。次にクロックが Low から High に切り替わるときに CE が High の場合、シフトレジスタの値は次の高位ビットにシフトされ、新しい値がロードされます。アドレス入力の値によってシフトレジスタの長さが決まり、Q にその値が出力されます。CE が Low の場合、クロック遷移は無視されます。

### 論理表

入力				出力
Am	CE	CLK	D	Q
Am	0	X	X	Q(Am)
Am	1	↑	D	Q(Am - 1)
m = 0、1、2、3				

## ポートの説明

ポート名	方向	幅	機能
Q	出力	1	シフトレジスタ データ出力
D	入力	1	シフトレジスタ データ入力
CLK	入力	1	クロック
CE	入力	1	アクティブ High のクロック イネーブル
A	入力	4	SRL のワード数のダイナミック選択 ・ A=0000 ==> 1 ビットシフト長 ・ A=1111 ==> 16 ビットシフト長

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	コンフィギュレーション後のシフトレジスタと出力の初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```

Library UNISIM;
use UNISIM.vcomponents.all;

-- SRL16E: 16-bit shift register LUT with clock enable operating on posedge of clock
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRL16E_inst : SRL16E
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  A0 => A0,         -- Select[0] input
  A1 => A1,         -- Select[1] input
  A2 => A2,         -- Select[2] input
  A3 => A3,         -- Select[3] input
  CE => CE,         -- Clock enable input
  CLK => CLK,       -- Clock input
  D => D            -- SRL data input
);

-- End of SRL16E_inst instantiation

```



## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRL16E: 16-bit shift register LUT with clock enable operating on posedge of clock
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRL16E_inst : SRL16E
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  A0 => A0,        -- Select[0] input
  A1 => A1,        -- Select[1] input
  A2 => A2,        -- Select[2] input
  A3 => A3,        -- Select[3] input
  CE => CE,        -- Clock enable input
  CLK => CLK,      -- Clock input
  D => D           -- SRL data input
);

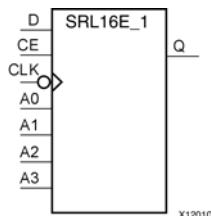
-- End of SRL16E_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## SRL16E\_1

: 16-Bit Shift Register Look-Up Table (LUT) with Negative-Edge Clock and Clock Enable



## 概要

このデザイン エLEMENTは、クロック イネーブル (CE) があるシフトレジスタ ルックアップ テーブル (LUT) で、クロックの立ち下がりエッジで動作します。シフトレジスタの長さは、入力 A3、A2、A1、A0 の値によって決定されます。

シフトレジスタの長さは、固定することも、変動させることもできます。

- ・ **固定長のシフトレジスタを作成するには：** 入力 A3 ～ A0 の値を一定の値にします。シフトレジスタは 1 ～ 16 ビットの長さに設定できます。アドレス入力の値によるシフトレジスタの長さは、長さ =  $(8 \times A3) + (4 \times A2) + (2 \times A1) + A0 + 1$  という式で算出できます。A3、A2、A1、A0 がすべてゼロの場合 (0000) はシフトレジスタの長さは 1 ビットになり、すべて 1 の場合 (1111) は 16 ビットになります。
- ・ **シフトレジスタ長を動的に変化させるには：** 入力 A3 ～ A0 の値を変化させます。たとえば、A2、A1、A0 がすべて 1 の場合 (111) に A3 を 1 から 0 に切り替えると、シフトレジスタの長さは 16 ビットから 8 ビットに変化します。内部的には、シフトレジスタの長さは常に 16 ビットで、どのビットの値が出力されるかは入力 A3 ～ A0 の値によって決定されます。

シフトレジスタ LUT の初期値を指定するには、INIT 属性に 4 桁の 16 進数を割り当てます。一番左の桁が最上位ビットになります。INIT の値を指定しない場合は、シフトレジスタ LUT の内容はコンフィギュレーション中にゼロ (0000) にクリアされます。

CE が High の場合、クロック (CLK) が High から Low に切り替わるときに、D の値がシフトレジスタの第 1 ビットにロードされます。次にクロックが High から Low に切り替わるときに CE が High の場合、シフトレジスタの値は次の高位ビットにシフトされ、新しい値がロードされます。アドレス入力の値によってシフトレジスタの長さが決まり、Q にその値が出力されます。CE が Low の場合、クロック遷移は無視されます。

## 論理表

入力				出力
Am	CE	CLK	D	Q
Am	0	X	X	Q(Am)
Am	1	↓	D	Q(Am - 1)
m = 0、1、2、3				

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	コンフィギュレーション後のシフトレジスタと出力の初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRL16E_1: 16-bit shift register LUT with clock enable operating on negedge of clock
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRL16E_1_inst : SRL16E_1
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  A0 => A0,         -- Select[0] input
  A1 => A1,         -- Select[1] input
  A2 => A2,         -- Select[2] input
  A3 => A3,         -- Select[3] input
  CE => CE,         -- Clock enable input
  CLK => CLK,       -- Clock input
  D => D            -- SRL data input
);

-- End of SRL16E_1_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRL16E_1: 16-bit shift register LUT with clock enable operating on negedge of clock
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRL16E_1_inst : SRL16E_1
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  A0 => A0,         -- Select[0] input
  A1 => A1,         -- Select[1] input
  A2 => A2,         -- Select[2] input
  A3 => A3,         -- Select[3] input
  CE => CE,         -- Clock enable input
  CLK => CLK,       -- Clock input
  D => D            -- SRL data input
);

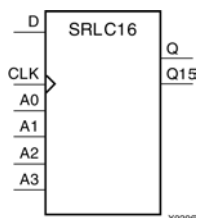
-- End of SRL16E_1_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## SRLC16

### : 16-Bit Shift Register Look-Up Table (LUT) with Carry



## 概要

このデザイン エLEMENTは、キャリーがあるシフトレジスタ ルックアップ テーブル (LUT) です。シフトレジスタの長さは、入力 A3、A2、A1、A0 の値によって決定されます。

シフトレジスタの長さは、固定することも、変動させることもできます。

- ・ **固定長のシフトレジスタを作成するには：** 入力 A3 ～ A0 の値を一定の値にします。シフトレジスタは 1 ～ 16 ビットの長さに設定できます。アドレス入力の値によるシフトレジスタの長さは、長さ =  $(8 \times A3) + (4 \times A2) + (2 \times A1) + A0 + 1$  という式で算出できます。A3、A2、A1、A0 がすべてゼロの場合 (0000) はシフトレジスタの長さは 1 ビットになり、すべて 1 の場合 (1111) は 16 ビットになります。
- ・ **シフトレジスタ長を動的に変化させるには：** 入力 A3 ～ A0 の値を変化させます。たとえば、A2、A1、A0 がすべて 1 の場合 (111) に A3 を 1 から 0 に切り替えると、シフトレジスタの長さは 16 ビットから 8 ビットに変化します。内部的には、シフトレジスタの長さは常に 16 ビットで、どのビットの値が出力されるかは入力 A3 ～ A0 の値によって決定されます。

シフトレジスタ LUT の初期値を指定するには、INIT 属性に 4 桁の 16 進数を割り当てます。一番左の桁が最上位ビットになります。INIT の値を指定しない場合は、シフトレジスタ LUT の内容はコンフィギュレーション中にゼロ (0000) にクリアされます。

クロック (CLK) が Low から High に切り替わるときに、D の値がシフトレジスタの第 1 ビットにロードされます。次にクロックが Low から High に切り替わると、シフトレジスタの値は次の高位ビットにシフトされ、新しい値がロードされます。アドレス入力の値によってシフトレジスタの長さが決まり、Q にその値が出力されます。

**メモ：** Q15 の出力を使用すると、複数のシフトレジスタ LUT をカスケード接続でき、より大きなシフトレジスタを作成できます。

## 論理表

入力			出力
A <sub>m</sub>	CLK	D	Q
A <sub>m</sub>	X	X	Q(A <sub>m</sub> )
A <sub>m</sub>	↑	D	Q(A <sub>m</sub> - 1)
m = 0、1、2、3			

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	コンフィギュレーション後のシフトレジスタと出力の初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRLC16: 16-bit cascadable shift register LUT operating on posedge of clock
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRLC16_inst : SRLC16
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  Q15 => Q15,      -- Carry output (connect to next SRL)
  A0 => A0,        -- Select[0] input
  A1 => A1,        -- Select[1] input
  A2 => A2,        -- Select[2] input
  A3 => A3,        -- Select[3] input
  CLK => CLK,      -- Clock input
  D => D           -- SRL data input
);

-- End of SRLC16_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRLC16: 16-bit cascadable shift register LUT operating on posedge of clock
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRLC16_inst : SRLC16
generic map (
    INIT => X"0000")
port map (
    Q => Q,          -- SRL data output
    Q15 => Q15,      -- Carry output (connect to next SRL)
    A0 => A0,        -- Select[0] input
    A1 => A1,        -- Select[1] input
    A2 => A2,        -- Select[2] input
    A3 => A3,        -- Select[3] input
    CLK => CLK,      -- Clock input
    D => D           -- SRL data input
);

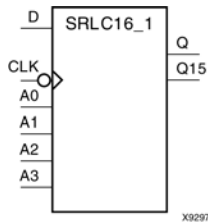
-- End of SRLC16_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## SRLC16\_1

： 16-Bit Shift Register Look-Up Table (LUT) with Carry and Negative-Edge Clock



### 概要

このデザイン エLEMENTは、キャリーがあるシフトレジスタ ルックアップ テーブル (LUT) で、クロックの立ち下がりエッジで動作します。シフトレジスタの長さは、入力 A3、A2、A1、A0 の値によって決定されます。

シフトレジスタの長さは、固定することも、変動させることもできます。

- ・ **固定長のシフトレジスタを作成するには：** 入力 A3 ～ A0 の値を一定の値にします。シフトレジスタは 1 ～ 16 ビットの長さに設定できます。アドレス入力の値によるシフトレジスタの長さは、長さ =  $(8 \times A3) + (4 \times A2) + (2 \times A1) + A0 + 1$  という式で算出できます。A3、A2、A1、A0 がすべてゼロの場合 (0000) はシフトレジスタの長さは 1 ビットになり、すべて 1 の場合 (1111) は 16 ビットになります。
- ・ **シフトレジスタ長を動的に変化させるには：** 入力 A3 ～ A0 の値を変化させます。たとえば、A2、A1、A0 がすべて 1 の場合 (111) に A3 を 1 から 0 に切り替えると、シフトレジスタの長さは 16 ビットから 8 ビットに変化します。内部的には、シフトレジスタの長さは常に 16 ビットで、どのビットの値が出力されるかは入力 A3 ～ A0 の値によって決定されます。

シフトレジスタ LUT の初期値を指定するには、INIT 属性に 4 桁の 16 進数を割り当てます。一番左の桁が最上位ビットになります。INIT の値を指定しない場合は、シフトレジスタ LUT の内容はコンフィギュレーション中にゼロ (0000) にクリアされます。

**メモ：** Q15 の出力を使用すると、複数のシフトレジスタ LUT をカスケード接続でき、より大きなシフトレジスタを作成できます。

### 論理表

入力			出力	
A <sub>m</sub>	CLK	D	Q	Q15
A <sub>m</sub>	X	X	Q(A <sub>m</sub> )	変化なし
A <sub>m</sub>	↓	D	Q(A <sub>m</sub> - 1)	Q14
m = 0、1、2、3				

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	コンフィギュレーション後のシフトレジスタと出力の初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRLC16_1: 16-bit cascadable shift register LUT operating on negedge of clock
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRLC16_1_inst : SRLC16_1
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  Q15 => Q15,       -- Carry output (connect to next SRL)
  A0 => A0,         -- Select[0] input
  A1 => A1,         -- Select[1] input
  A2 => A2,         -- Select[2] input
  A3 => A3,         -- Select[3] input
  CLK => CLK,       -- Clock input
  D => D           -- SRL data input
);

-- End of SRLC16_1_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRLC16_1: 16-bit cascadable shift register LUT operating on negedge of clock
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRLC16_1_inst : SRLC16_1
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  Q15 => Q15,       -- Carry output (connect to next SRL)
  A0 => A0,         -- Select[0] input
  A1 => A1,         -- Select[1] input
  A2 => A2,         -- Select[2] input
  A3 => A3,         -- Select[3] input
  CLK => CLK,       -- Clock input
  D => D           -- SRL data input
);

-- End of SRLC16_1_inst instantiation
```

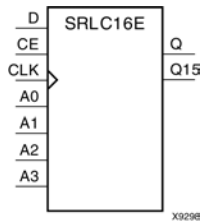
## 詳細情報

- [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- [Spartan-3E FPGA ファミリー データシート](#)



## SRLC16E

： 16-Bit Shift Register Look-Up Table (LUT) with Carry and Clock Enable



### 概要

このデザイン エLEMENTは、キャリーとクロック イネーブルがあるシフトレジスタ ルックアップ テーブル (LUT) です。シフトレジスタの長さは、入力 A3、A2、A1、A0 の値によって決定されます。

シフトレジスタの長さは、固定することも、変動させることもできます。

- ・ **固定長のシフトレジスタを作成するには：** 入力 A3 ～ A0 の値を一定の値にします。シフトレジスタは 1 ～ 16 ビットの長さに設定できます。アドレス入力の値によるシフトレジスタの長さは、長さ =  $(8 \times A3) + (4 \times A2) + (2 \times A1) + A0 + 1$  という式で算出できます。A3、A2、A1、A0 がすべてゼロの場合 (0000) はシフトレジスタの長さは 1 ビットになり、すべて 1 の場合 (1111) は 16 ビットになります。
- ・ **シフトレジスタ長を動的に変化させるには：** 入力 A3 ～ A0 の値を変化させます。たとえば、A2、A1、A0 がすべて 1 の場合 (111) に A3 を 1 から 0 に切り替えると、シフトレジスタの長さは 16 ビットから 8 ビットに変化します。内部的には、シフトレジスタの長さは常に 16 ビットで、どのビットの値が出力されるかは入力 A3 ～ A0 の値によって決定されます。

シフトレジスタ LUT の初期値を指定するには、INIT 属性に 4 桁の 16 進数を割り当てます。一番左の桁が最上位ビットになります。INIT の値を指定しない場合は、シフトレジスタ LUT の内容はコンフィギュレーション中にゼロ (0000) にクリアされます。

クロック (CLK) が Low から High に切り替わるときに、D の値がシフトレジスタの第 1 ビットにロードされます。次にクロックが Low から High に切り替わるときに CE が High の場合、シフトレジスタの値は次の高位ビットにシフトされ、新しいデータがロードされます。アドレス入力の値によってシフトレジスタの長さが決まり、Q にその値が出力されます。

**メモ：** Q15 の出力を使用すると、複数のシフトレジスタ LUT をカスケード接続でき、より大きなシフトレジスタを作成できます。

### 論理表

入力				出力	
A <sub>m</sub>	CLK	CE	D	Q	Q15
A <sub>m</sub>	X	0	X	Q(A <sub>m</sub> )	Q(15)
A <sub>m</sub>	X	1	X	Q(A <sub>m</sub> )	Q(15)
A <sub>m</sub>	↑	1	D	Q(A <sub>m</sub> - 1)	Q15
m = 0、1、2、3					

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	コンフィギュレーション後のシフトレジスタと出力の初期値を指定

## VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRLC16E: 16-bit cascable shift register LUT with clock enable operating on posedge of clock
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRLC16E_inst : SRLC16E
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  Q15 => Q15,      -- Carry output (connect to next SRL)
  A0 => A0,        -- Select[0] input
  A1 => A1,        -- Select[1] input
  A2 => A2,        -- Select[2] input
  A3 => A3,        -- Select[3] input
  CE => CE,        -- Clock enable input
  CLK => CLK,      -- Clock input
  D => D           -- SRL data input
);

-- End of SRLC16E_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRLC16E: 16-bit cascable shift register LUT with clock enable operating on posedge of clock
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRLC16E_inst : SRLC16E
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  Q15 => Q15,      -- Carry output (connect to next SRL)
  A0 => A0,        -- Select[0] input
  A1 => A1,        -- Select[1] input
  A2 => A2,        -- Select[2] input
  A3 => A3,        -- Select[3] input
  CE => CE,        -- Clock enable input
  CLK => CLK,      -- Clock input
  D => D           -- SRL data input
);

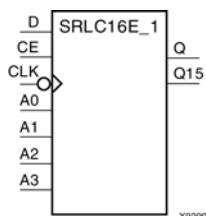
-- End of SRLC16E_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## SRLC16E\_1

: 16-Bit Shift Register Look-Up Table (LUT) with Carry, Negative-Edge Clock, and Clock Enable



### 概要

このデザイン エLEMENTは、キャリーおよびクロック イネーブルがあるシフトレジスタ ルックアップ テーブル (LUT) で、クロックの立ち下がりエッジで動作します。シフトレジスタの長さは、入力 A3、A2、A1、A0 の値によって決定されます。

シフトレジスタの長さは、固定することも、変動させることもできます。

- ・ **固定長のシフトレジスタを作成するには：** 入力 A3 ～ A0 の値を一定の値にします。シフトレジスタは 1 ～ 16 ビットの長さに設定できます。アドレス入力の値によるシフトレジスタの長さは、長さ =  $(8 \times A3) + (4 \times A2) + (2 \times A1) + A0 + 1$  という式で算出できます。A3、A2、A1、A0 がすべてゼロの場合 (0000) はシフトレジスタの長さは 1 ビットになり、すべて 1 の場合 (1111) は 16 ビットになります。
- ・ **シフトレジスタ長を動的に変化させるには：** 入力 A3 ～ A0 の値を変化させます。たとえば、A2、A1、A0 がすべて 1 の場合 (111) に A3 を 1 から 0 に切り替えると、シフトレジスタの長さは 16 ビットから 8 ビットに変化します。内部的には、シフトレジスタの長さは常に 16 ビットで、どのビットの値が出力されるかは入力 A3 ～ A0 の値によって決定されます。

シフトレジスタ LUT の初期値を指定するには、INIT 属性に 4 桁の 16 進数を割り当てます。一番左の桁が最上位ビットになります。INIT の値を指定しない場合は、シフトレジスタ LUT の内容はコンフィギュレーション中にゼロ (0000) にクリアされます。

CE が High の場合、クロック (CLK) が High から Low に切り替わるときに、D の値がシフトレジスタの第 1 ビットにロードされます。次にクロックが High から Low に切り替わるときに CE が High の場合、シフトレジスタの値は次の高位ビットにシフトされ、新しい値がロードされます。アドレス入力の値によってシフトレジスタの長さが決まり、Q にその値が出力されます。

**メモ：** Q15 の出力を使用すると、複数のシフトレジスタ LUT をカスケード接続でき、より大きなシフトレジスタを作成できます。

### 論理表

入力				出力	
Am	CE	CLK	D	Q	Q15
Am	0	X	X	Q(Am)	変化なし
Am	1	X	X	Q(Am)	変化なし
Am	1	↓	D	Q(Am-1)	Q14
m = 0、1、2、3					

## デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

## 使用可能な属性

属性	タイプ	値	デフォルト	説明
INIT	16 進数	16 ビット値	すべてゼロ	コンフィギュレーション後のシフトレジスタと出力の初期値を指定

## VHDL 記述（インスタンス化）

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRLC16E_1: 16-bit shift register LUT with clock enable operating on negedge of clock
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRLC16E_1_inst : SRLC16E_1
generic map (
  INIT => X"0000")
port map (
  Q => Q,          -- SRL data output
  Q15 => Q15,      -- Carry output (connect to next SRL)
  A0 => A0,        -- Select[0] input
  A1 => A1,        -- Select[1] input
  A2 => A2,        -- Select[2] input
  A3 => A3,        -- Select[3] input
  CE => CE,        -- Clock enable input
  CLK => CLK,      -- Clock input
  D => D           -- SRL data input
);

-- End of SRLC16E_1_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- SRLC16E_1: 16-bit shift register LUT with clock enable operating on negedge of clock
--           Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

SRLC16E_1_inst : SRLC16E_1
generic map (
  INIT => X"0000")
port map (
  Q => Q,           -- SRL data output
  Q15 => Q15,       -- Carry output (connect to next SRL)
  A0 => A0,         -- Select[0] input
  A1 => A1,         -- Select[1] input
  A2 => A2,         -- Select[2] input
  A3 => A3,         -- Select[3] input
  CE => CE,         -- Clock enable input
  CLK => CLK,       -- Clock input
  D => D            -- SRL data input
);

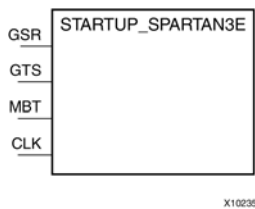
-- End of SRLC16E_1_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## STARTUP\_SPARTAN3E

: Spartan®-3E User Interface to the GSR, GTS, Configuration Startup Sequence and Multi-Boot Trigger Circuitry



### 概要

このデザイン エLEMENTは、ポートまたはユーザー回路を接続して、FPGA 内の特定の専用回路および配線を制御します。このコンポーネントの GSR ポートに接続された信号は、グローバル セット/リセット (GSR) を制御できます。GSR ネットは、デバイスのすべてのレジスタに接続され、レジスタを初期値にします。GTS ポートに信号を接続すると、各ピンのトライステート出力を制御する専用配線にポートが接続されます。CLK 入力にクロック信号を接続すると、コンフィギュレーション後のスタートアップ シーケンスはユーザー定義のクロックに同期します。MBT (マルチブートトリガ) ピンを使用すると、新しいコンフィギュレーションが開始されます (デバイスでこの機能が設定されている場合)。

### デザインの入力方法

インスタンス化	推奨
推論	不可
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- STARTUP_SPARTAN3E: Startup primitive for GSR, GTS, startup sequence
--                   control and Multi-Boot Configuration.
--                   Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

STARTUP_SPARTAN3E_inst : STARTUP_SPARTAN3E
port map (
  CLK => CLK,          -- Clock input for start-up sequence
  GSR => GSR_PORT,     -- Global Set/Reset input (GSR cannot be used for the port name)
  GTS => GTS_PORT      -- Global 3-state input (GTS cannot be used for the port name)
  MBT => MBT -- Multi-Boot Trigger input
);

-- End of STARTUP_SPARTAN3E_inst instantiation
```

## Verilog 記述（インスタンス化）

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- STARTUP_SPARTAN3E: Startup primitive for GSR, GTS, startup sequence
--                   control and Multi-Boot Configuration.
--                   Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

STARTUP_SPARTAN3E_inst : STARTUP_SPARTAN3E
port map (
    CLK => CLK,          -- Clock input for start-up sequence
    GSR => GSR_PORT, -- Global Set/Reset input (GSR cannot be used for the port name)
    GTS => GTS_PORT  -- Global 3-state input (GTS cannot be used for the port name)
    MBT => MBT -- Multi-Boot Trigger input
);

-- End of STARTUP_SPARTAN3E_inst instantiation
```

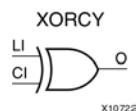
## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)



## XORCY

: XOR for Carry Logic with General Output



### 概要

このデザイン エLEMENTは、一般出力 (O) のある特殊な XOR ゲートで、高速で小型の演算ファンクションを生成するために使用します。このプリミティブは、スライスのキャリーチェーン ロジック内の専用 XOR ファンクションで、演算ファンクション (加算または除算) または多入力ロジック ファンクション (多入力 AND または OR ゲート) を高速かつ効率的に作成できます。

### 論理表

入力		出力
LI	CI	O
0	0	0
0	1	1
1	0	1
1	1	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- XORCY: Carry-Chain XOR-gate with general output
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

XORCY_inst : XORCY
port map (
    O => O,    -- XOR output signal
    CI => CI,  -- Carry input signal
    LI => LI   -- LUT4 input signal
);

-- End of XORCY_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- XORCY: Carry-Chain XOR-gate with general output
--      Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

XORCY_inst : XORCY
port map (
    O => O,    -- XOR output signal
    CI => CI,  -- Carry input signal
    LI => LI   -- LUT4 input signal
);

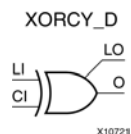
-- End of XORCY_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## XORCY\_D

: XOR for Carry Logic with Dual Output



### 概要

このデザイン エLEMENTは、一般出力 (O) とローカル出力 (LO) のある特殊な XOR ゲートで、高速で小型の演算ファンクションを生成するために使用します。

### 論理表

入力		出力
LI	CI	O および LO
0	0	0
0	1	1
1	0	1
1	1	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- XORCY_D: Carry-Chain XOR-gate with local and general outputs
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

XORCY_D_inst : XORCY_D
port map (
    LO => LO, -- XOR local output signal
    O  => O,  -- XOR general output signal
    CI => CI, -- Carry input signal
    LI => LI  -- LUT4 input signal
);

-- End of XORCY_D_inst instantiation
```

## Verilog 記述（インスタンス化）

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- XORCY_D: Carry-Chain XOR-gate with local and general outputs
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

XORCY_D_inst : XORCY_D
port map (
  LO => LO, -- XOR local output signal
  O  => O,  -- XOR general output signal
  CI => CI, -- Carry input signal
  LI => LI  -- LUT4 input signal
);

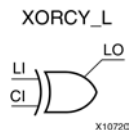
-- End of XORCY_D_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)

## XORCY\_L

: XOR for Carry Logic with Local Output



### 概要

このデザイン エLEMENTは、ローカル出力 (LO) のある特殊な XOR ゲートで、高速で小型の演算ファンクションを生成するために使用します。

### 論理表

入力		出力
LI	CI	LO
0	0	0
0	1	1
1	0	1
1	1	0

### デザインの入力方法

インスタンス化	可
推論	推奨
CORE Generator™ およびウィザード	不可
マクロのサポート	不可

### VHDL 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- XORCY_L: Carry-Chain XOR-gate with local  => direct-connect output
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

XORCY_L_inst : XORCY_L
port map (
    LO => LO, -- XOR local output signal
    CI => CI, -- Carry input signal
    LI => LI  -- LUT4 input signal
);

-- End of XORCY_L_inst instantiation
```

## Verilog 記述 (インスタンス化)

次の 2 つの構文が存在しない場合はコピーし、エンティティ宣言の前に貼り付けます。

```
Library UNISIM;
use UNISIM.vcomponents.all;

-- XORCY_L: Carry-Chain XOR-gate with local  => direct-connect output
--          Spartan-3E
-- Xilinx HDL Libraries Guide, version 12.2

XORCY_L_inst : XORCY_L
port map (
    LO => LO, -- XOR local output signal
    CI => CI, -- Carry input signal
    LI => LI  -- LUT4 input signal
);

-- End of XORCY_L_inst instantiation
```

## 詳細情報

- ・ [Spartan-3 ジェネレーション FPGA ユーザー ガイド](#)
- ・ [Spartan-3E FPGA ファミリー データシート](#)