



## Virtex ファミリ FPGA による柔軟で高速な CAM のデザイン

XAPP203、1999 年 9 月 23 日 (バージョン 1.1)

アプリケーション ノート : Jean-Louis Brelet &amp; Bernie New

### 概要

CAM (Content Addressable Memories) を使用すると、メモリ内にある特定のデータを素早く検索できます。CAM の条件は、アプリケーションによって異なります。Virtex™ ファミリのスライスで CAM デザインをインプリメントすると、シフトレジスタとしてコンフィギュレーションした LUT に基づいて CAM の深さと幅を柔軟に設定できます。このアプリケーション ノートでは、1 クロック サイクルで一致を検索する高速な CAM デザインについて説明します。アプリケーション ノート XAPP201 『Virtex ファミリ デバイスにおける複数 CAM デザインの概要』では、CAM をインプリメントするときに利用できる各種のソリューションについて説明し、このアプリケーション ノートで説明するソリューションを紹介しています。

### Xilinx ファミリ

Virtex™ および Virtex™-E FPGA

### はじめに

アプリケーション ノート XAPP201 では、CAM と RAM の基本的な違いについて説明し、Virtex ファミリ デバイスにおける 3 個の CAM デザイン ソリューションの集積度と速度を比較しています。

CAM を使用すると、メモリへの入力データを同時に検索できます。これは、RAM に対する CAM の主要な利点です。一致が検出されると、CAM は対応するアドレスを出力します。

このアプリケーション ノートでは、Virtex または Virtex-E スライスで小規模から中規模の柔軟な CAM をデザインする手法を紹介します。Virtex スライスに組み込まれているシフトレジスタプリミティブを使用することによって、再コンフィギュレーション可能な LUT (スライスごとに 2 個の LUT) を使用し、読み出しが 1 クロック サイクルの CAM をインプリメントします。4 ビットの CAM ワードは、各 LUT に収まります。32 ワード × 16 ビットの CAM では、128 個の LUT が必要になります。書き込み動作ではシフトレジスタモードを使用するので、16 クロック サイクルが必要になります。

### 一般的な説明

オプションのインバータ付きの 4 入力 AND ゲートは 1 つの LUT に収まり、専用キャリー チェーンを通してカスケードできます。各 Virtex スライスには、2 個の LUT と 1 つの高速キャリー チェーンがあります。1 列のスライスは、各 LUT 出力を AND ワイヤで接続したキャリー チェーンを使用した高速のデコーダです。読み出し動作は、LUT とキャリー チェーンを通して入力データをデコードする動作と同じです。

1 列のスライス数は、Virtex ファミリ デバイスによって異なります。このスライス列 (または部分列) は同時に、LUT が再コンフィギュレーション可能な場合の CAM ワードの記憶装置の位置およびコンパレータでもあります。各 LUT

でシフトレジスタモード (SRL16E プリミティブ) を使用すると、デコーダへの書き込み動作が可能になります。

複数のスライス列は、CAM ワードのアレイと同じです。列の数も、Virtex ファミリ デバイスによって異なります。設計者は、正しい Virtex または Virtex-E デバイスを簡単に見つけて、特定の CAM サイズをインプリメントできます。

入力データバスは、Virtex ファミリ デバイスの高速配線機能を通して、それぞれのワード、つまり LUT に分散します。32 ワード × 16 ビットの CAM には、32 個の終端点がある 16 個のハイ ファンアウト ネットがあります。

### デザインの概要

このアプリケーション ノートで説明する参考デザインは、ワードの幅と深さを調節できます。基本的な CAM モジュールの出力は、デコードされた (ワンホット) アドレスです。多くの場合、クリティカル タイミングのパラメータは比較動作のときの CAM に対するアクセス時間です。このデザインは、Virtex ファミリ デバイスで 7.5ns の速度で (スピード グレードは -6)、1 クロック サイクルでデコードされたアドレスを提供します。デコードされたアドレスは、メモリ ロケーションごとに 1 ラインと定義されます。出力ラインの数は、CAM のワード数です。出力が High の場合、対応するワードは入力データに一致しています。

この参考デザインには、出力アドレスを生成するエンコーディング モジュールも含まれています。VHDL コードの構造は階層的なので、設計者の要求に応じてさまざまなサイズの CAM に、このモジュールを容易に使用できます。通常、エンコーダが出力アドレスと一致フラグの両方を生成するために必要となる追加クロック サイクルは、1 クロック サイクルのみです。すべてのデコードされたアドレスのワイド OR ゲートでこの一致フラグが生成されます。

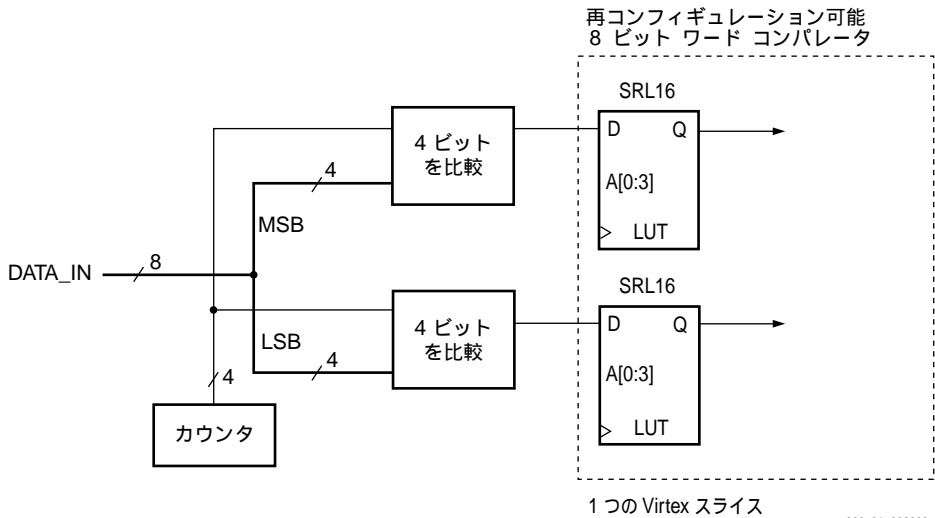


図 1: 8 ビット CAM ワードの書き込み動作 (16 クロック サイクル)

### 書き込み動作

あるロケーションに新しいデータを格納するには、シフトレジスタモードを使用します。書き込む入力データと 4 ビットダウンカウンタ (16 個のステート) を比較した結果は、16 クロックサイクルでシフトされます。カウンタの値と入力データの値が等しい場合、SRL16E プリミティブに 1 がシフトインされます。そうでなければ 0 がシフトインされます。この結果、16 クロックサイクル後に各 LUT で 4 ビットのデコーダになります。図 1 は、8 ビットの CAM ワードの書き込み動作です。

### 読み出し動作

比較する入力データは、シフトレジスタのアドレスとして使用されます。SRL16E の 16 個のロケーションの中で 1 つだけが、以前に格納されたデータに対応する 1 になっています。入力データがこのロケーションをアドレスしている場合は、一致が検出されます。キャリーチェーンはこの 1 を伝搬します (ワイド AND コンフィギュレーション)。特定のスライス列にあるすべての SRL16E が 1 を出力する場合には、一致が検出されたこととなります。キャリーチェーン全体の出力は、1 ラインのデコードされた CAM アドレスです。

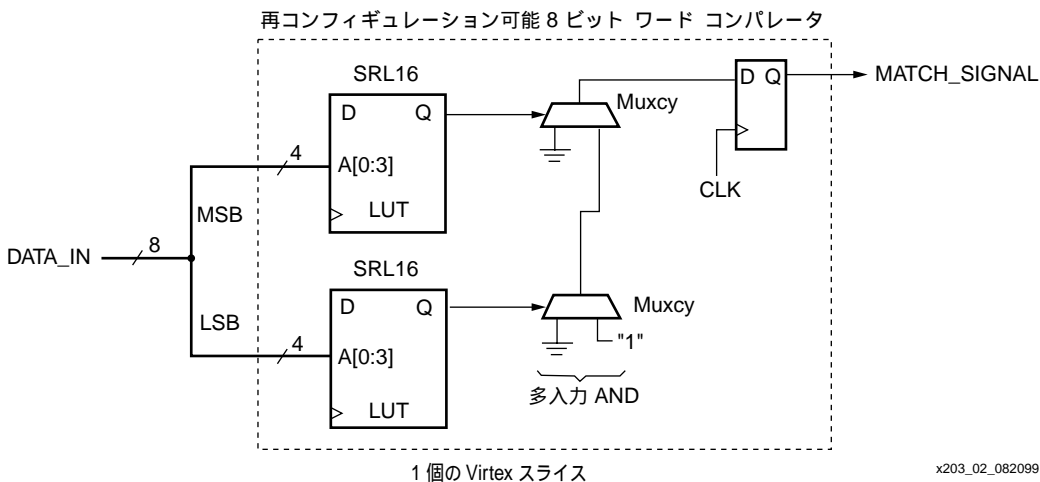


図 2: 8 ビット CAM ワードの読み出し動作 (1 クロック サイクル)

## Virtex ファミリの SRL16E プリミティブによる CAM のデザイン

このアプリケーション ノートを参考デザイン ファイル XAPP203.zip とともに使用すると、Virtex または Virtex-E スライスを使用して高速で柔軟な CAM を作成できます。Virtex ファミリ デバイスの CAM デザインの 1 番目の例は、階層的な HDL 参考デザインに示されています。**11 ページの付録 A** を参照してください。

### 特徴

- 高速な 1 クロック サイクルの読み出しと比較アクセス時間。
- 書き込みは 16 クロック サイクル。
- 4 ビットから 4 の倍数ビットの任意の値までジェネリックな word\_width。ただし、使用する Virtex ファミリ デバイスの 1 列のスライス数によって制限されます。たとえば、XCV50 には 1 列に 16 個のスライス (32 個の LUT) があるので、128 ビットワードまでをインプリメントできます。また、XCV300 には 1 列に 32 個のスライス (64 個の LUT) があるので、256 ビットワードまでをインプリメントできます。
- 基本となる 16 ワード ブロックを定義するジェネリックな nb\_cam\_16words。CAM の深さは、16

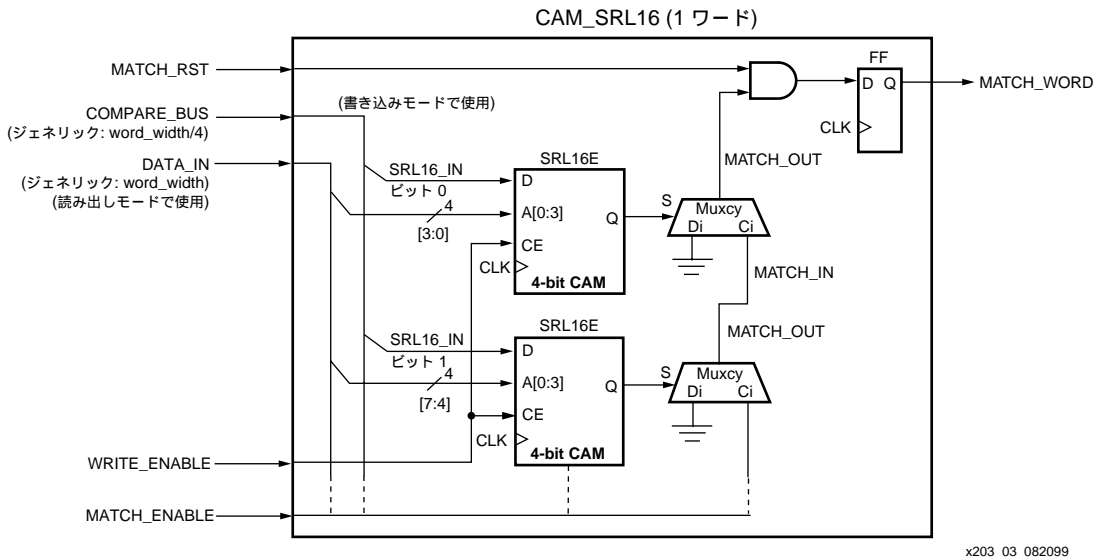
ワード値の倍数になります。ワードの幅と深さによりデザインに必要な Virtex ファミリが決まります。各 LUT は、4 ビットの基本ブロック CAM をインプリメントします。

- nb\_cam\_16words に直接結び付けられたアドレスラインの数を定義するジェネリックな addr\_width。5 アドレス ラインを必要とする 32 ワード CAM、6 アドレス ラインを必要とする 64 ワード CAM、7 アドレス ラインを必要とする 128 ワード CAM があります。

### 基本的な構成要素 : CAM\_SRL16

高速比較 CAM のインプリメンテーションは、4 個の CAM ビットを格納および比較するシフトレジスタ SRL16E と、この基本ブロックをカスケードしてワイド AND ゲートを構築するとき使用する、対応するキャリー チェーン MUXCY から構成されています。

ジェネリックなワード幅は、4 の倍数です (各 LUT は 4 ビットを格納します)。8 ビットのワード幅では 1 個の Virtex スライス (2 × SRL16E と 2 × MUXCY) が必要になり、16 ビットでは 2 個の Virtex スライス、32 ビットでは 4 個、64 ビットでは 8 個などようになります。



x203\_03\_082099

図 3: 1 個の Virtex スライス (カスケード可能) の 8 ビットワード (CAM\_SRL16)

このアプローチの主要な利点の 1 つとして、コントロールロジックがワード幅およびデザイン階層から独立しているということがあります。

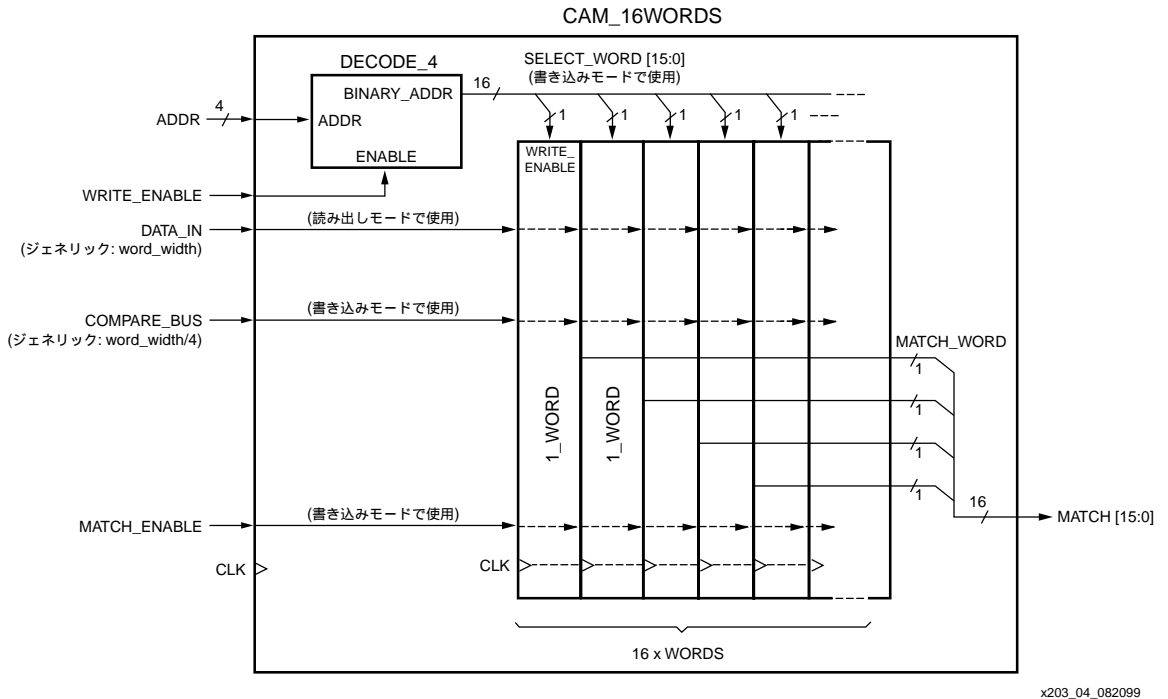
Virtex のキャリー チェーンは高速なので、ワード幅が大きくなっても (128 ビット以上) 速度にはあまり影響を与えません。ワード幅によって変化するパスは、MUXCY に構築したワイド AND ゲートだけです (各 4 ビットによって追加されるのは MUXCY 遅延だけです)。

# XAPP203: SRL における CAM

## モジュール : CAM\_16WORDS

CAMの1ワードは、1個の16ワードモジュールにグループ化されます。単純な4:16デコーダが、書き込みモードで選択したCAMワードにWRITE\_ENABLEを提供します。別のモジュールCOMPARE\_4は、4ビットカウンタに対する入力データを比較して、COMPARE\_BUSを生成します。

COMPARE\_BUSは、SRL16EのD入力に接続され、CAMワードにシフトインされます。16クロックサイクルの書き込み動作によって、入力データ値はSRL16Eの正しいロケーションにシフトされます。

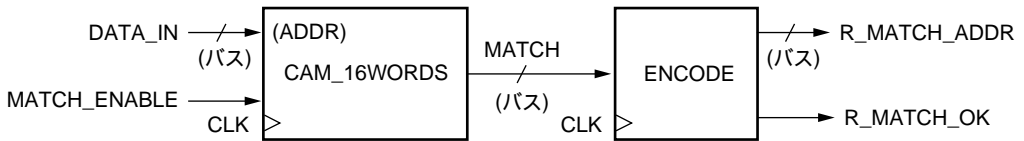
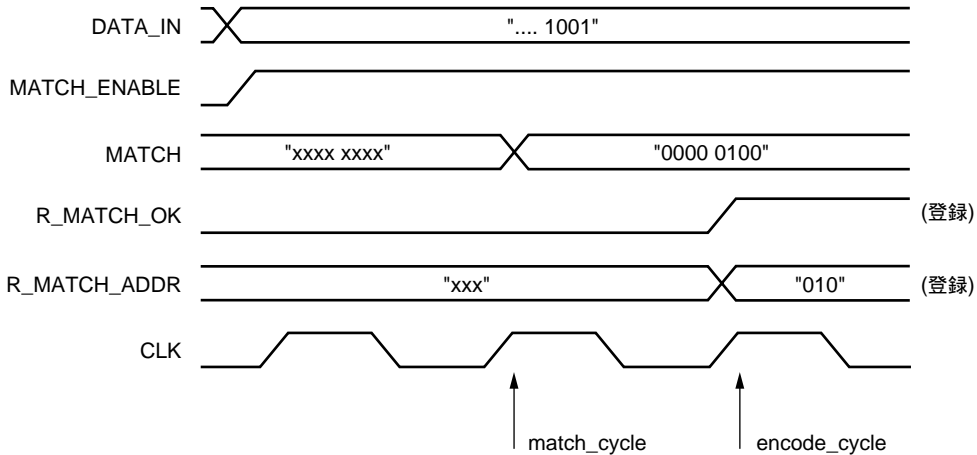


x203\_04\_082099

図 4: CAM\_16WORDS のブロック図

読み出しサイクル モード

読み出しモードでは、DATA\_IN は各 CAM ワードと並列して比較され、16 ビットの MATCH バスを生成します。たとえば、8 ビットの MATCH バスを使用する場合、DATA\_IN が WORD 2 で検出されると、MATCH バスはデコードされたアドレス 00000100 になります。一致が検出されない場合、MATCH バスは 00000000 になります。ロジックを追加すると、マスクや複数の一致（複数のロケーションでの DATA\_IN の検出）を扱うことができます。



x203\_05\_082099

図 5: CAM の読み出しモードの波形

# XAPP203: SRL における CAM

## モジュール : CAM\_generic\_word (読み出しモード)

必要なサイズの CAM を作成するには、複数の CAM\_16WORDS モジュールを CAM\_generic\_word モジュールでインスタンス化します。CAM の深さは nb\_cam\_16words というジェネリックな値で、16 ワードの倍数になります。1 個の CAM\_16WORDS ごとに、4 ビットの MATCH アドレス出力パスが 1 個ずつ必要になります。ジェネリック値 addr\_width は、CAM\_16WORDS の数、つまり CAM の深さに対応します。32 ワードの CAM では 5 個のアドレスラインが、64 ワードの CAM では 6 個のアドレスラインが必要になります。

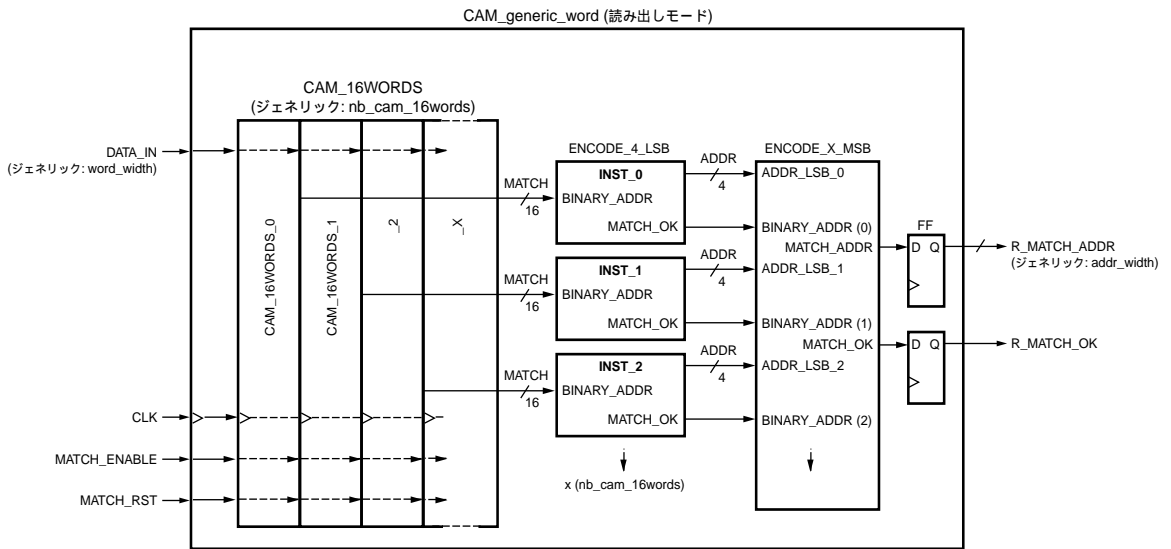


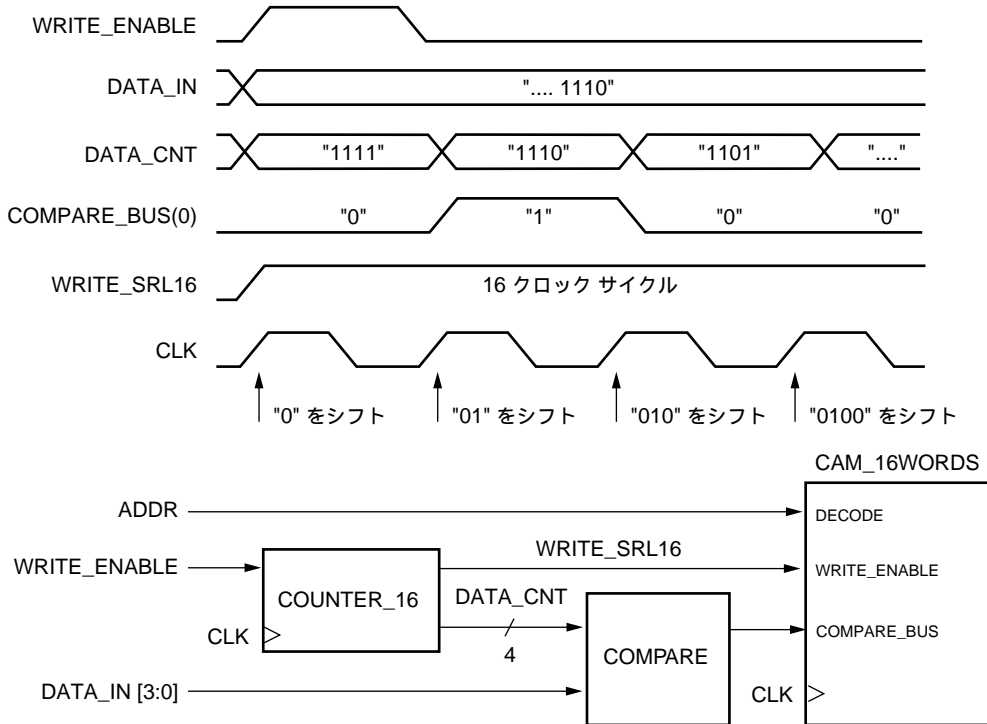
図 6: CAM の読み出しパス (1 クロック サイクル)

MATCH パスをエンコードして、一致出力 MATCH\_ADDR と MATCH\_OK 信号 (一致の検出時に High) の両方を生成します。

書き込みサイクル

WRITE\_ENABLE 信号は "1111" を "0000" ヘダウンする 4 ビット カウンタを起動し、書き込みイネーブル信号 WRITE\_SRL16 は 16 クロック サイクルの間アサートされたままになります。各カウンタ値は、各 4 ビット DATA\_IN パス（書き込むデータ）と並列して比較されます。

COMPARE 出力は、SRL16E アレイの各 D 入力に接続されます。



x201\_07\_082099

図 7: CAM の書き込みモードの波形

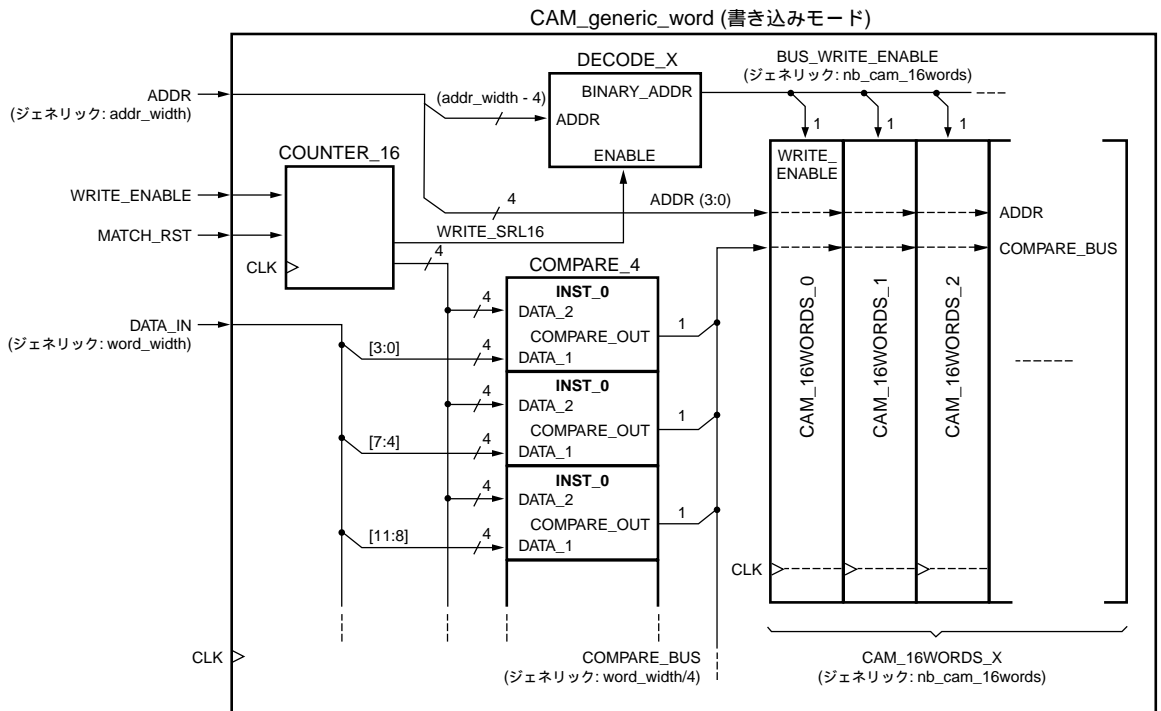
いずれかの DATA\_IN (4 ビット) がカウンタ値と等しい場合は 1 が、等しくなければ 0 が、選択した CAM ワードの SRL16E にシフトインされます。標準的なアドレス デコーダが、通常の RAM メモリと同じように書き込む CAM ワードを選択します。4 ビットの DATA\_IN 値が "1110" の場合、初めのクロック サイクルシフトで "000" がシフトされ、2 番目のクロック サイクルで "1" がシフトインされ、残りの 14 クロック サイクルで "0" がシフトインされます。

この結果、SRL16E の内容は 0100000000000000 になります。読み出しモードでは、この SRL16E の A0:A3 入力に DATA\_IN パスを接続すると、A(3:0) = "1110" というパターンのみで 1 (一致) が生成されます。読み出しモードの場合、SRL16E の出力は非同期で、LUT の出力と同じになります。

# XAPP203: SRL における CAM

## モジュール : CAM\_generic\_word (書き込みモード)

このモジュールの書き込みモード部分は、デザイン全体について 1 個のカウンタと 1 個のコンパレータ (COMPARE\_4 × "word\_width/4") をインスタンス化します。標準的な最上位レベルのアドレスデコーダは、各 CAM\_16WORDS について選択バス WRITE\_ENABLE を生成します。



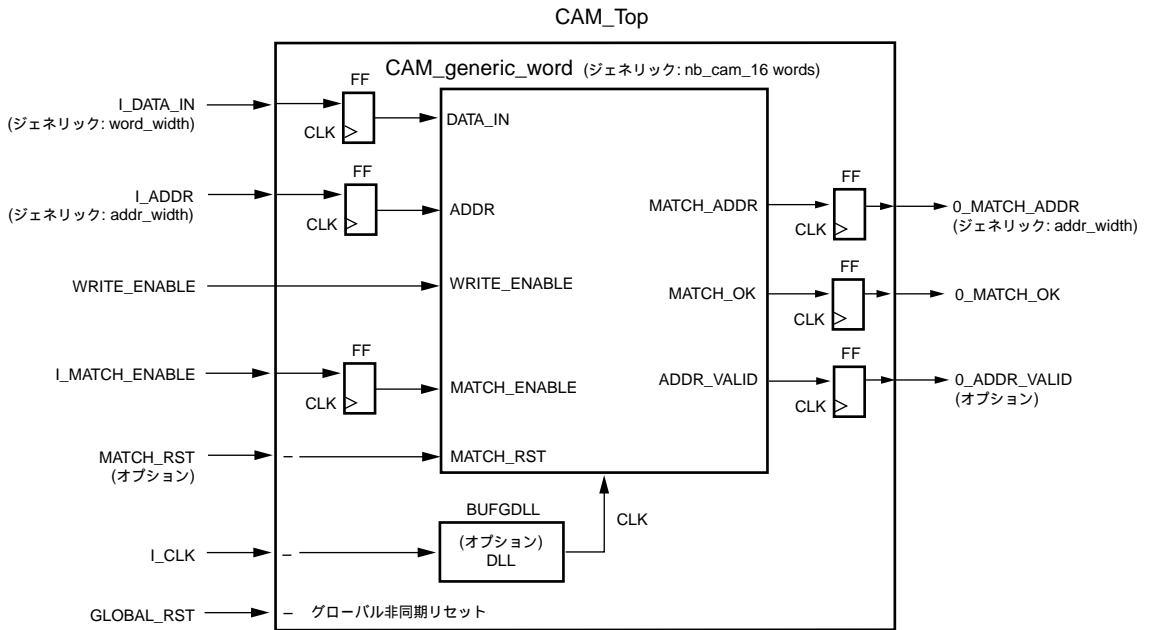
x203\_08\_082099

図 8: CAM の書き込みバス (15 から 0 にカウント ダウンする 16 クロック サイクル)



モジュール : CAM\_top

このモジュールは、ジェネリックな CAM の最上位ラッパーです。すべての入力および出力信号を登録します。



x203\_09\_082099

図 9: CAM デザインの最上位レベルブロック図

ピン配置 : (I\_XXX はすべて入力、O\_XXX はすべて出力)

- I\_DATA\_IN は、読み出し動作と書き込み動作の両方で使用するジェネリックなデータバスです。
- I\_ADDR は、新しいデータのみを選択したロケーションに書き込むために使用するジェネリックな入力アドレスバスです。
- WRITE\_ENABLE は、16 クロックサイクルの書き込み動作をイネーブルにする 1 クロックサイクルの信号です (アクティブ High)。
- I\_MATCH\_ENABLE は、読み出しアクセスをイネーブルにします (アクティブ High)。
- MATCH\_RST は、同期リセットで、CAM の内容を変更しません (オプション)。
- I\_CLK は、クロックで、オプションとして、DLL を通して配線できます。
- O\_MATCH\_ADDR は、読み出しモードのみのジェネリックな出力アドレスです。
- O\_MATCH\_OK は、一致が検出されると High になります (読み出し動作)。
- O\_ADDR\_VALID は、複数の一致が発生した場合のオプションの信号です。1 個の一致が検出された場合はアクティブ High です。

この参考デザインを合成するときは、その後のフロアプランのために階層を維持し、スタティック タイミング解析を行うことをお勧めします。クロック周期は、制約ファイル (UCF) で定義します。ハイ ファンアウト ネットなので、UCF ファイルは次のネットも制約します。

COMPARE\_BUS の COMPARE\_4 モジュールの内部出力と DATA\_IN バスは、各ラインのファンアウトが CAM の深さ以上になる 2 個のバスです。パフォーマンスを最高にするため、次の例のように、簡単な UCF 制約で各ラインを自動的に制約します。

```
NET "DATA_IN<*>" MAXDELAY = 3ns;
```

```
NET "CAM_generic_word_1/COMPARE_BUS<*>"  
MAXDELAY = 3ns
```

このアプリケーション ノートに付随する参考デザインは、異なる CAM モードにも容易に適合できます。

## おわりに

Virtex と Virtex-E の SRL16E に基づくインプリメンテーションは、小規模から大規模なワード幅に適しており、パフォーマンスに対する影響はわずかです。4 ビット幅を追加すると、増えるのは 1 個のキャリー チェーン MUXCY の遅延のみです。デザイン構造に変化はありません。ワード幅に対する制限があるとすれば、Virtex ファミリー デバイスの列ごとのスライス数のみです。しかし、スライスごとに 8 ビットなので、非常に大きなワード数が利用可能です。Virtex XCV50 と XCV50E は 1 列に 16 個のスライスが、XCV1000 と XCV1000E は 1 列に 64 個のスライスが、XCV3200E は 1 列に 104 個のスライスがあります。

CAM の深さ (スライス列の数) も拡張できるので、真の柔軟性が提供されます。デコードされたアドレスや一致したバスはすべて、CAM のサイズに関係なく 1 クロック サイクルで生成されます。参考デザインでは、ゲートまたはトライステート バッファでアドレス エンコーダと一致フラグを生成する方法を示しています。これらのオプションは、最終的な CAM インプリメンテーションに合わせて設計者が任意に選択できます。

アプリケーションの条件はさまざまなので、CAM ソリューションのアプローチにはさまざまなものがあります。Virtex ファミリーのユニークな機能は、このような柔軟なソリューションを提供するための主要な利点となっています。このアプリケーション ノートで提供されているデザインに加えて、XAPP202 『ATM アプリケーションにおける CAM デザイン』と XAPP204 『Block SelectRAM+ を使用した高いパフォーマンスの リード/ライト CAM』にも補足的なソリューションがあります。

## 付録 A : 合成可能な参考デザインの HDL コード

付録Aでは、ワード幅とメモリの深さがパラメータ化されたCAMをVirtexスライスでインプリメントする階層的で合成可能なデザインを示します。完全なHDLコードは、参考デザインとして使用できます(ファイル:xapp203.zipまたはxapp203.tar.z)。

各VHDLモジュールのヘッダを次に示します。

### モジュール : CAM\_top.vhd

```
--
-- Module:          CAM_Top / Top Level
-- Design:          CAM_Top
-- VHDL code:       Hierarchical wrapper
--                  Instantiated CAM_generic_word (variable depth and word width)
--
-- Synthesis        Synopsys FPGA Express ver. 3.2 - Option = Preserve Hierarchy
--                  Use of "pragma synthesis_off/on" and attributes
--
-- Description:     Instantiated a CAM implementation
--                  Registered inputs and outputs (CAM internal timing analysis)
--
-- Device:          Virtex Families (Virtex and Virtex-E)
--
-- Created by:      Jean-Louis BRELET / XILINX - Virtex Applications
-- Date:            July 23, 1999
-- Version:         1.0
--
-- History:
--   1.
--
-- Disclaimer:      THESE DESIGNS ARE PROVIDED "AS IS" WITH NO WARRANTY
--                  WHATSOEVER AND XILINX SPECIFICALLY DISCLAIMS ANY
--                  IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR
--                  A PARTICULAR PURPOSE, OR AGAINST INFRINGEMENT.
--
-- Copyright (c) 1999 Xilinx, Inc. All rights reserved.
```

---

### モジュール : CAM\_generic\_word.vhd

```
--
-- Module:          CAM_generic_word
-- Design:          CAM_Top
-- VHDL code:       Hierarchical RTL
--                  Instantiated COUNT_16
--                  Instantiated COMPARE_4
--                  Instantiated CAM_16WORDS
--                  Instantiated DECODE_4
--                  Instantiated CAM_SRL16
--                  Instantiated INIT_SRL16_AND
--                  Instantiated ENCODE_4_LSB
--                  Instantiated DECODE_X
--                  Instantiated ENCODE_X_MSB
--                  Choose the right ENCODE and DECODE modules according to the number of CAM_16WORDS
--                  If "nb_cam_16words" = 2 then ENCODE_1_MSB and DECODE_1 must be used:
1 bit to decode 2 CAM_16WORDS
--
--                  If "nb_cam_16words" = 4 then ENCODE_2_MSB and DECODE_2 must be used:
```

```

2 bits to decode 4 CAM_16WORDS
--           If "nb_cam_16words" = 8 then ENCODE_3_MSB and DECODE_3 must be used:
3 bits to decode 8 CAM_16WORDS
--           If "nb_cam_16words" = 16 then ENCODE_4_MSB and DECODE_4 must be used:
4 bits to decode 16 CAM_16WORDS
--           Note: synthesis tools do not support Configuration
--
-- Synthesis      Synopsys FPGA Express ver. 3.2 - Option = Preserve Hierarchy
--                Use of "pragma synthesis_off/on" and attributes
--
-- Description:   Instantiated "nb_cam_16words" CAM_16WORDS (see generic)
--                2 x 16 words depth x 8 bits width by default
--                1 clock cycle Read (or Match),
--                16 clock cycles Write
--                MATCH_OK indicates one or more matches is/are found.
--                MATCH_ADDR output the address of the match, if ONLY ONE is found
--                ADDR_VALID indicates when MATCH_ADDR is a valid address (Optional)
--
-- Device:        Virtex Families
--                modules CAM_16WORDS fits in 1 Virtex slices column
--                (4 CAM bits per LUT + global control logic)
--                If "nb_cam_16words" = 2 then CAM32xWORD
--                If "nb_cam_16words" = 4 then CAM64xWORD
--                If "nb_cam_16words" = 8 then CAM128xWORD
--                If "nb_cam_16words" = 16 then CAM256xWORD
-- ...

```

---

### モジュール : COUNT\_16.vhd

```

--
-- Module:        COUNT_16
-- Design:        CAM_Top
-- VHDL code:     RTL
--
-- Synthesis      Synopsys FPGA Express ver. 3.2
--                Use of "pragma synthesis_off/on" and attributes
--
-- Description:   4 bits counter 15 downto 0
--                Generate a 16 clock cycle wide enable signal
-- Device:        Virtex Families
-- ...

```

---

### モジュール : COMPARE\_4.vhd

```

--
-- Module:        COMPARE_4
-- Design:        CAM_Top
-- VHDL code:     Virtex primitives' instantiation
--
-- Synthesis      Synopsys FPGA Express ver. 3.2
--                Use of "pragma synthesis_off/on" and attributes
--
-- Description:   Basic building block of a CAM to compare 2 x 4 bits busses
--                Combinatorial module
--
-- Device:        Virtex Families

```

```
--          Gates and MUXF5 fitting in 1 Logic Cell.
```

```
-- ...
```

---

### モジュール : CAM\_16WORDS.vhd

```
--
-- Module:          CAM_16WORDS
-- Design:          CAM_Top
-- VHDL code:       Hierarchical RTL
--                   Instantiated DECODE_4
--                   Instantiated CAM_SRL16
--                   Instantiated INIT_SRL16_AND
--
-- Synthesis        Synopsys FPGA Express ver. 3.2
--                   Use of "pragma synthesis_off/on" and attributes
--
-- Description:     Building block of a CAM 16 words (variable word width)
--                   1 word depth x "word_width" bits width
--                   1 clock cycle Read (or Match), 16 clock cycles Write
--                   If NO match is found, or MATCH_ENABLE is low, the output MATCH =
--                   '0000000000000000'
--                   Initialized SRL16E in low level module
--
-- Device:          Virtex Families
-- ...
```

---

### モジュール : CAM\_SRL16.vhd

```
--
-- Module:          CAM_SRL16
-- Design:          CAM_Top
-- VHDL code:       Hierarchical RTL
--                   Instantiated INIT_SRL16_AND
--
-- Synthesis        Synopsys FPGA Express ver. 3.2
--                   Use of "pragma synthesis_off/on" and attributes
--
-- Description:     Basic building block of a CAM using 16-Bit Shift Register LUT
--                   1 word depth x "word_width" bits width
--                   1 clock cycle Read (or Match), 16 clock cycles Write
--                   If NO match is found, or MATCH_ENABLE is low, the output MATCH_WORD = '0'
--                   Initialized SRL16E in low level module
--
-- Device:          Virtex Families
-- ...
```

---

### モジュール : INIT\_SRL16\_AND.vhd

```
--
-- Module:          INIT_SRL16_AND
-- Design:          CAM_Top
-- VHDL code:       Virtex primitives' instantiation
--
-- Synthesis        Synopsys FPGA Express ver. 3.2
--                   Use of "pragma synthesis_off/on" and attributes
--
```

```
-- Description:  Basic building block of a CAM using 16-Bit Shift Register LUT
--                4 bits per LUT
--                Asynchronous Read (or Match) by ADDR -> MATCH_OUT
--                Initialized SRL16E: attributes to constraint PAR and simulation
--
-- Device:        Virtex Families
--                SRL16E and MUXCY
-- ...
```

---

### モジュール : ENCODE\_4\_LSB.vhd

```
--
-- Module:        ENCODE_4_LSB
-- Design:        CAM_Top
-- VHDL code:     RTL / Combinatorial
--
-- Synthesis      Synopsys FPGA Express ver. 3.2
--                Use of "pragma synthesis_off/on" and attributes
--
-- Description:   Encode a 16 bits binary address into 4 bits and find if a match occurs
--                if BINARY_ADDR = "0000000000100000" => MATCH_ADDR = "0101" / MATCH_OK = 1
--                Optional ADDR_VALID = 1 when only one Match (If simultaneous matches can
--                occur)
--                However, the ADDR_VALID generation double the size of the combinatorial
--                logic !
--                if no match found => MATCH_OK = 0 / ADDR_VALID = 0 (MATCH_ADDR is not a
--                valid address)
--                if 2 or more matches found => MATCH_OK = 1 / ADDR_VALID = 0 (MATCH_ADDR is
--                not valid address)
--
-- Device:        Virtex Families
-- ...
```

---

### モジュール : ENCODE\_4\_MSB.vhd

このモジュールは、1ビットの ENCODE\_1\_MSB (CAM 32 ワード)、2ビットの ENCODE\_2\_MSB (CAM 64 ワード)、3ビットの ENCODE\_3\_MSB (CAM 128 ワード)、4ビットの ENCODE\_4\_MSB (CAM 256 ワード) で使用できます。このモデルに従えば、深さが異なる CAM についてのモジュールも容易に作成できます。

```
--
-- Module:        ENCODE_4_MSB
-- Design:        CAM_Top
-- VHDL code:     RTL / Combinatorial
--
-- Synthesis      Synopsys FPGA Express ver. 3.2
--                Use of "pragma synthesis_off/on" and attributes
--
-- Description:   Encode a 16 bits binary address into 4 bits, map with the LSB address and
--                find if a match occurs
--                if BINARY_ADDR = "0000000000100000" => MATCH_ADDR = "0101" / MATCH_OK = 1
--                Optional ADDR_VALID = 1 when only one Match (If simultaneous matches can
--                occur)
--                However, the ADDR_VALID generation double the size of the combinatorial
--                logic !
--                if no match found => MATCH_OK = 0 / ADDR_VALID = 0 (MATCH_ADDR is not a
--                valid address)
```

```
--          if 2 or more matches found => MATCH_OK = 1 / ADDR_VALID = 0 (MATCH_ADDR is
not valid address)
--
--          Choice between GATES ONLY implementation or BUFT implementation. (See
comments)
--          Note: synthesis tools do not support Configuration
--
-- Device:      Virtex Families
-- ...
```

---

## モジュール : DECODE\_4.vhd

このモジュールは、1 ビットの DECODE\_1 (CAM 32 ワード)、2 ビットの DECODE\_2 (CAM 64 ワード)、3 ビットの DECODE\_3 (CAM 128 ワード)、4 ビットの DECODE\_4 (CAM 256 ワード) で使用できます。このモデルに従えば、深さが異なる CAM についてのモジュールも容易に作成できます。

```
--
-- Module:      DECODE_4
-- Design:      CAM_Top
-- VHDL code:   RTL / Combinatorial
--
-- Synthesis    Synopsys FPGA Express ver. 3.2
--              Use of "pragma synthesis_off/on" and attributes
--
-- Description:  Decode 4 bits address into 16 binary bits
--              Generate an ENABLE bus
--
-- Device:      Virtex Families
-- ...
```

---

参考デザイン ファイルには、同様のモジュール ENCODE\_3\_MSB、ENCODE\_2\_MSB、ENCODE\_1\_MSB、DECODE\_4、DECODE\_3、DECODE\_2、DECODE\_1 が含まれています。

**改訂履歴**

日付	リビジョン	改訂内容
1999 年 8 月 31 日	1.0	初期リリース
1999 年 9 月 23 日	1.1	Virtex-E の初期アップデート