



Gowin User Flash

ユーザーガイド

UG295-1.4.4J, 2024-09-27

著作権について(2024)

著作権に関する全ての権利は、**Guangdong Gowin Semiconductor Corporation** に留保されています。

GOWIN高云、**Gowin**、及び**LittleBee**は、当社により、中国、米国特許商標庁、及びその他の国において登録されています。商標又はサービスマークとして特定されたその他全ての文字やロゴは、それぞれの権利者に帰属しています。何れの団体及び個人も、当社の書面による許可を得ず、本文書の内容の一部もしくは全部を、いかなる視聴覚的、電子的、機械的、複写、録音等の手段によりもしくは形式により、伝搬又は複製をしてはなりません。

免責事項

当社は、GOWINSEMI Terms and Conditions of Sale(GOWINSEMI取引条件)に規定されている内容を除き、(明示的か又は黙示的かに拘わらず)いかなる保証もせず、また、知的財産権や材料の使用によりあなたのハードウェア、ソフトウェア、データ、又は財産が被った損害についても責任を負いません。当社は、事前の通知なく、いつでも本文書の内容を変更することができます。本文書を参照する何れの団体及び個人も、最新の文書やエラッタ(不具合情報)については、当社に問い合わせる必要があります。

バージョン履歴

日付	バージョン	説明
2020/08/24	1.0J	初版。
2021/01/12	1.1J	FLASH256KA の説明を追加。
2021/07/14	1.2J	<ul style="list-style-type: none"> ● デバイス GW1N-2B、GW1N-1P5、GW1N-1P5B、GW1NR-2B のサポートを追加。 ● IP 呼び出しの図面を更新。 ● FLASH256KA を FLASH96KA に変更。 ● FLASH96K の説明を更新。
2021/11/14	1.3J	<ul style="list-style-type: none"> ● サポートされるデバイスを更新。 ● クロック周波数の説明を更新。 ● 一部の説明を更新。
2022/11/04	1.4J	デバイス(GW1NS-2、GW1NS-2C、GW1NSE-2C、GW1NSR-2、GW1NSR-2C)を削除。
2023/01/05	1.4.1J	IP 呼び出しの一部の図面を更新、Device Version オプションを追加。
2023/02/03	1.4.2J	<ul style="list-style-type: none"> ● Flash の特性のクロック周波数をアクセス時間に変更。 ● タイミングパラメータの説明を更新。
2023/02/22	1.4.3J	FLASH64K、FLASH256K、FLASH96KA、FLASH608K のタイミングパラメータおよびタイミング図の説明を更新。
2024/09/27	1.4.4J	「表 3-1 サポートされるデバイス」を更新。

目次

目次	i
図一覧	ii
表一覧	iii
1 本マニュアルについて	1
1.1 マニュアルの内容	1
1.2 関連ドキュメント	1
1.3 用語、略語	1
1.4 テクニカル・サポートとフィードバック	1
2 概要	2
3 プリミティブの紹介	3
3.1 FLASH96K	3
3.2 FLASH64KZ	11
3.3 FLASH64K	17
3.4 FLASH256K	21
3.5 FLASH96KA	24
3.6 FLASH608K	28
4 IP の呼び出し	32

図一覧

図 3-1 FLASH96K のポート図	4
図 3-2 FLASH96K の読み出しモード	10
図 3-3 FLASH96K のページラッチ書き込みモード	10
図 3-4 FLASH96K のページラッチ・クリア・モード	10
図 3-5 FLASH96K の High レベル時間	11
図 3-6 FLASH64KZ のポート図	12
図 3-7 FLASH64KZ の読み出しタイミング	16
図 3-8 FLASH64KZ の書き込みタイミング	16
図 3-9 FLASH64KZ の消去タイミング	16
図 3-10 FLASH64K のポート図	17
図 3-11 FLASH256K のポート図	22
図 3-12 FLASH96KA のポート図	25
図 3-13 FLASH608K のポート図	29
図 4-1 User Flash の構成ウィンドウ	33

表一覧

表 1-1 用語、略語	1
表 3-1 サポートされるデバイス	3
表 3-2 FLASH96K のポートの説明	4
表 3-3 FLASH96K 出力データ幅の選択	5
表 3-4 FLASH96K 入力データ幅の選択	5
表 3-5 FLASH96K 操作モードの選択	5
表 3-6 FLASH96K のタイミングパラメータ	9
表 3-7 FLASH64KZ のポートの説明	12
表 3-8 FLASH64KZ のユーザーモードの真理値表	13
表 3-9 FLASH64KZ のタイミングパラメータ	14
表 3-10 FLASH64K のポートの説明	17
表 3-11 FLASH64K のユーザーモードの真理値表	18
表 3-12 FLASH256K のポートの説明	22
表 3-13 FLASH256K のユーザーモードの真理値表	23
表 3-14 FLASH96KA のポートの説明	25
表 3-15 FLASH96KA のユーザーモードの真理値表	26
表 3-16 FLASH608K のポートの説明	29
表 3-17 FLASH608K のユーザーモードの真理値表	29

1 本マニュアルについて

1.1 マニュアルの内容

本マニュアルでは、User Flash の機能、プリミティブの定義、およびその使用法について説明します。

1.2 関連ドキュメント

GOWIN セミコンダクターのホームページ www.gowinsemi.com/ja から、以下の関連ドキュメントがダウンロード、参考できます：Gowin Flash Controller IP ユーザーガイド([IPUG901](#))。

1.3 用語、略語

本マニュアルで使用される用語、略語、及びその意味を表 1-1 に示します。

表 1-1 用語、略語

用語、略語	正式名称	意味
FPGA	Field Programmable Gate Array	フィールド・プログラマブル・ゲート・アレイ
IP Core	Intellectual Property Core	設計資産コア

1.4 テクニカル・サポートとフィードバック

GOWIN セミコンダクターは、包括的な技術サポートをご提供しています。使用に関するご質問、ご意見については、直接弊社までお問い合わせください。

ホームページ : www.gowinsemi.com/ja

E-mail : support@gowinsemi.com

2 概要

Gowin LittleBee ファミリーFPGA 製品は、User Flash を提供します。サポートされる User Flash の容量は、FPGA デバイスによって異なります。FLASH96K、FLASH64K、FLASH64KZ、FLASH256K、FLASH96KA、FLASH608K などがあります。

3 プリミティブの紹介

User Flash プリミティブとサポートされるデバイスの対応関係を表 3-1 に示します。

表 3-1 サポートされるデバイス

プリミティブ	サポートされるデバイス
FLASH96K	GW1N-1、GW1N-1S、GW1NR-1
FLASH64KZ	GW1NZ-LV1
FLASH64K	GW1NZ-ZV1、GW1NZ-1C
FLASH256K	GW1N-4、GW1N-4B、GW1N-4D、GW1NR-4、GW1NR-4B、GW1NR-4D、GW1NRF-4B、GW1NS-4、GW1NS-4C、GW1NSR-4、GW1NSR-4C、GW1NSER-4C
FLASH96KA	GW1N-2、GW1N-2B、GW1N-2C、GW1N-1P5、GW1N-1P5B、GW1N-1P5C、GW1NR-2、GW1NR-2B、GW1NR-2C、GW1NZ-2B、GW1NZ-2C
FLASH608K	GW1N-9、GW1N-9C、GW1NR-9、GW1NR-9C

注記：

GW1NS-4C、GW1NSR-4C、GW1NSER-4C の User Flash は MCU 専用です。

3.1 FLASH96K

プリミティブの紹介

FLASH96K(96K ビット User Flash)の記憶領域は 96K ビットです。この Flash の幅と深さは構成できません。幅が 4 バイト(32 ビット)、アドレス深さが 3K の不揮発性メモリです。初期値機能はサポートされていません。

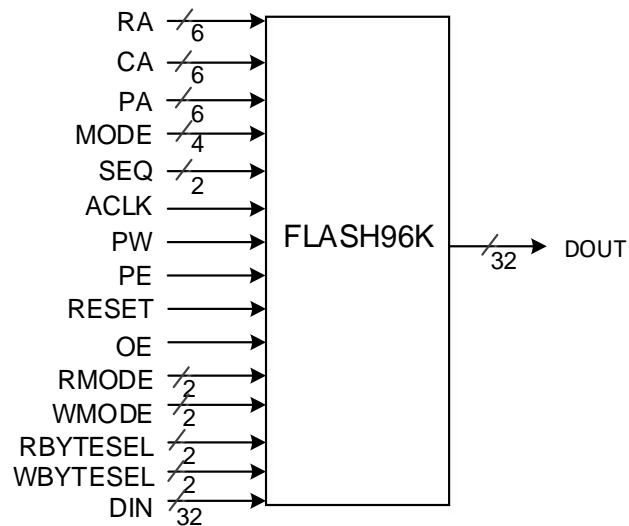
FLASH96K には以下の特徴があります。

- 100,000 回の書き込みサイクル
- 10 年以上のデータ保持時間(+85°C)
- 構成可能なデータ幅：8/16/32 ビット
- 容量：48 行*64 列*32 ビット = 96 K ビット

- ページサイズ : 256 バイト
- $3\mu\text{A}$ のスタンバイ電流
- ページ書き込み時間 : 8.2ms

ポート図

図 3-1 FLASH96K のポート図



ポートの説明

表 3-2 FLASH96K のポートの説明

ポート	I/O	説明
DOUT[31:0]	出力	データ出力バス
DIN[31:0]	入力	データ入力バス
RA[5:0]	入力	メモリの行を選択するために使用される行アドレスバス。
CA[5:0]	入力	メモリの列を選択するために使用される列アドレスバス。
PA[5:0]	入力	ページラッチアドレスの列を選択するために使用されます。
MODE[3:0]	入力	操作モード
SEQ[1:0]	入力	操作の順序の制御。
ACLK	入力	読み出しと書き込みの同期クロック
PW	入力	ページラッチ・データのクロック入力。
RESET	入力	リセット信号、アクティブ High。
PE	入力	チャージポンプイネーブル
OE	入力	データ出力イネーブル
RMODE[1:0]	入力	読み出しデータ幅の制御
WMODE[1:0]	入力	書き込みデータ幅の制御

ポート	I/O	説明
RBYTESEL[1:0]	入力	読み出しデータのバイト選択
WBYTESEL[1:0]	入力	書き込みデータのバイト選択

構成モード

ユーザーは、読み出し/書き込みモードおよび読み出し/書き込みバイト選択信号を使用して、入力および出力データの幅を変更できます。データの幅と制御信号の対応関係を表 3-3 および表 3-4 に示します。

表 3-3 FLASH96K 出力データ幅の選択

RMOD[1:0]	RBYTESEL		DOUT			
	[1]	[0]	[31:24]	[23:16]	[15:8]	[7:0]
00	Yes	Yes	No	No	No	Yes
01	Yes	No	No	No	Yes	Yes
1X	No	No	Yes	Yes	Yes	Yes

表 3-4 FLASH96K 入力データ幅の選択

WMOD[1:0]	WBYTESEL		DIN			
	[1]	[0]	[31:24]	[23:16]	[15:8]	[7:0]
00	Yes	Yes	No	No	No	Yes
01	Yes	No	No	No	Yes	Yes
1X	No	No	Yes	Yes	Yes	Yes

注記：

「Yes」は有効であることを表し、「No」は無効であることを表します。

動作モード

ユーザーは、MODE[3:0]の値を設定して操作モードを選択することができます。詳しくは表 3-5 に示す通りです。

表 3-5 FLASH96K 操作モードの選択

MODE[3:0]	説明
0000	通常読み出し及びページラッチ・データ書き込み
0001	プレプログラミング・ビットをセット。書き込み開始後自動的にクリア
0100	ページラッチ・データをクリア
1000	ページ(または行)を消去
1100	ページ(または行)書き込み

- 読み出し

MODE を「0000」に設定した後、ACLK の立ち上がりエッジで読み

出しモードに入ります。読み出しモードでは、SEQ[1:0]の値を「00」のままにする必要があります。データ取得時間の条件(<=38ns)が満たされた後、データは出力ピン DOUT に現れます。

- 書き込み

User Flash の書き込みは、5つのステップで構成されます：

1. ページラッチ・データをクリア
2. データをページラッチに書き込む。
3. 選択したメモリユニットを仮想の「1」にプレプログラム。
4. 選択したメモリユニットを消去。
5. ページラッチ内のデータをメモリユニットに書き込む。

メモリユニットを消去した後、データは「0」になります。メモリユニットに書き込んだ後は、データは「1」になります。メモリユニットの「0」は、書き込みによって「1」になれますが、「1」は書き込みによって「0」になることができません。そのため、書き込み前に、メモリユニットを消去する必要があります。

- ページラッチに書き込む

ページラッチは、Flash に書き込まれるデータをバッファリングする SRAM のようなものです。ページラッチへの書き込み操作は、PW 信号によってのみ制御され、ACLK とは関係ありません。PA(Page Address)信号は、書き込まれるページラッチのアドレスを指定します。

ページラッチへの書き込みの前に、まずデータを消去する必要があります。ページラッチにデータを1つずつ書き込み、MODE 値を「0000」、SEQ [1 : 0]を「00」に設定します。ページラッチへの書き込みとデータの読み出しは完全に独立しています。

- ページラッチをクリア

ページラッチへの書き込みと違って、ページラッチのクリアは ACLK によって制御されます。MODE を「0100」に設定した後、ACLK の立ち上がりエッジの際ページラッチ・クリア・モードに入ります。このモードでは、SEQ[1:0]を「00」のままにする必要があります。ページラッチ・データは1つの ACLK サイクル内でクリアされます。

- 消去および書き込み

消去および書き込み操作では、SEQ の値を順番に 1>2>3>0 に設定する必要があります。これらの操作にはミリ秒レベルの時間が必要です。消去後に同じページに2回書き込むことは禁止されています。

消去および書き込み操作前は、選択したメモリユニットにプレプログラミング操作によって仮想の「1」を書き込む必要があります。プレプログラミングの操作手順は次のとおりです。

1. PEP(pre-program)ビットをセットします(MODE=「0001」)。

2. High レベルの間に選択したエリアに書き込みます(MODE=「1100」)。このプロセスは 100 マイクロ秒以上続きます。

その中で、MODE の値を「1100」に設定した後の手順は、消去および書き込み操作と同じく SEQ 1-> 2-> 3-> 0 を実行する必要がありますが、一部のタイミングが異なります(図 3-5)。例えば、Tpe が異なります(表 3-6)。

プリミティブのインスタンス化

プリミティブを直接インスタンス化するか、IP Core Generator で生成できます。詳しくは、[4 IP の呼び出し](#)を参照してください。

Verilog でのインスタンス化 :

```
FLASH96K flash96k_inst(
    .RA(ra[5:0]),
    .CA(ca[5:0]),
    .PA(pa[5:0]),
    .MODE(mode[3:0]),
    .SEQ(seq[1:0]),
    .ACLK(aclk),
    .PW(pw),
    .RESET(reset),
    .PE(pe),
    .OE(oe),
    .RMODE(rmode[1:0]),
    .WMODE(wmode[1:0]),
    .RBYTESEL(rbytesel[1:0]),
    .WBYTESEL(wbytesel[1:0]),
    .DIN(din[31:0]),
    .DOUT(dout[31:0])
);
```

VHDL でのインスタンス化 :

```
COMPONENT FLASH96K
    PORT(
        RA:IN std_logic_vector(5 downto 0);
        CA:IN std_logic_vector(5 downto 0);
        PA:IN std_logic_vector(5 downto 0);
        MODE:IN std_logic_vector(3 downto 0);
```

```
        SEQ:IN std_logic_vector(1 downto 0);
        ACLK:IN std_logic;
        PW:IN std_logic;
        RESET:IN std_logic;
        PE:IN std_logic;
        OE:IN std_logic;
        RMODE:IN std_logic_vector(1 downto 0);
        WMODE:IN std_logic_vector(1 downto 0);
        RBYTESEL:IN std_logic_vector(1 downto 0);
        WBYTESEL:IN std_logic_vector(1 downto 0);
        DIN:IN std_logic_vector(31 downto 0);
        DOUT:OUT std_logic_vector(31 downto 0)
    );
END COMPONENT;
 uut: FLASH96K
    PORT MAP (
        RA=>ra,
        CA=>ca,
        PA=>pa,
        MODE=>mode,
        SEQ=>seq,
        RESET=>reset,
        ACLK=>aclk,
        PW=>pw,
        PE=>pe,
        OE=>oe,
        RMODE=>rmode,
        WMODE=>wmode,
        RBYTESEL=>rbytesel,
        WBYTESEL=> wbytesel,
        DIN=>din,
        DOUT=>dout
    );
```

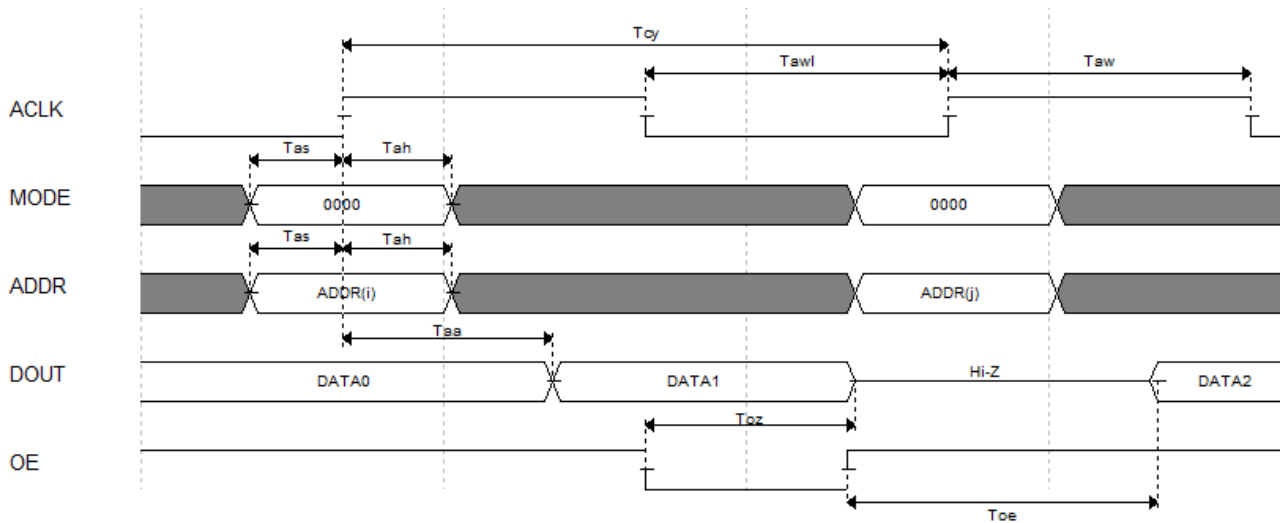
タイミングパラメータ

表 3-6 FLASH96K のタイミングパラメータ

パラメータ	説明	仕様			単位
		最小値	標準値	最大値	
Taa	データ取得時間	-	-	38	ns
Tcy	読み出しサイクル	43	-	-	ns
Taw	ACLK の High レベル時間	10	-	-	ns
Tawl	ACLK の Low レベル時間	10	-	-	ns
Tas	セットアップ時間	3	-	-	ns
Tah	ホールド時間	3	-	-	ns
Toz	OE プルダウンから高インピーダンスの時間	-	-	2	ns
Toe	OE プルアップからデータ出力の時間	-	-	2	ns
Twcy	書き込みサイクル	40	-	-	ns
Tpw	PW の High レベル時間	16	-	-	ns
Tpwl	PW の Low レベル時間	16	-	-	ns
Tpas	ページアドレスのセットアップ時間	3	-	-	ns
Tpah	ページアドレスのホールド時間	3	-	-	ns
Tds	データのセットアップ時間	16	-	-	ns
Tdh	データのホールド時間	3	-	-	ns
Ts0	SEQ=0 の場合のサイクル	6	-	-	μ s
Ts1	SEQ=1 の場合のサイクル	15	-	-	μ s
Ts2p	ACLK から PE 立ち上がりエッジのセットアップ時間	5	-	10	μ s
Ts3	SEQ=3 の場合のサイクル	5	-	10	μ s
Tps3	PE 立ち下がりエッジから ACLK のセットアップ時間	60	-	-	μ s
Tpe	MODE=1000 の場合、消去時間	5.7	6	6.3	ms
	MODE=1100 の場合、書き込み時間	1.9	2	2.1	ms
	MODE=11xx の場合、プレプログラミング時間	190	200	210	μ s

タイミング図

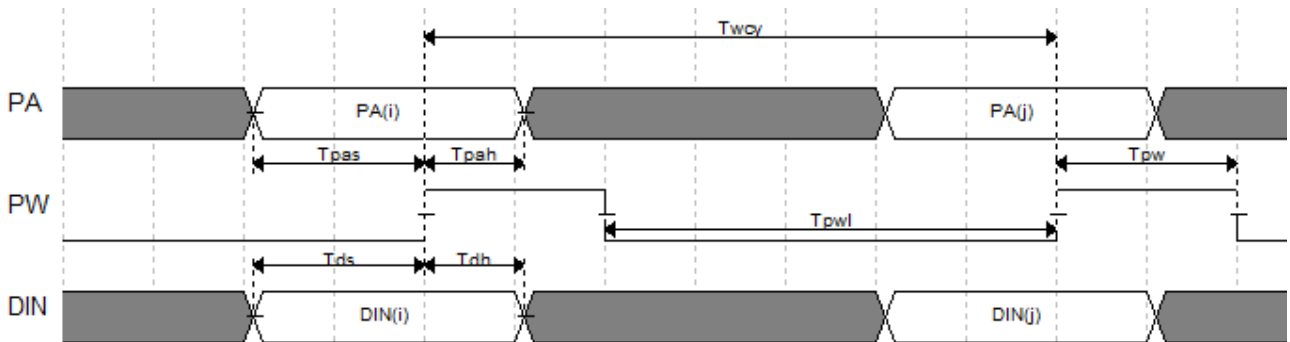
図 3-2 FLASH96K の読み出しモード



注記：

読み出しモードの場合、SEQ=0。ADDR 信号には RA、CA、RMOD、RBYTESEL が含まれます。

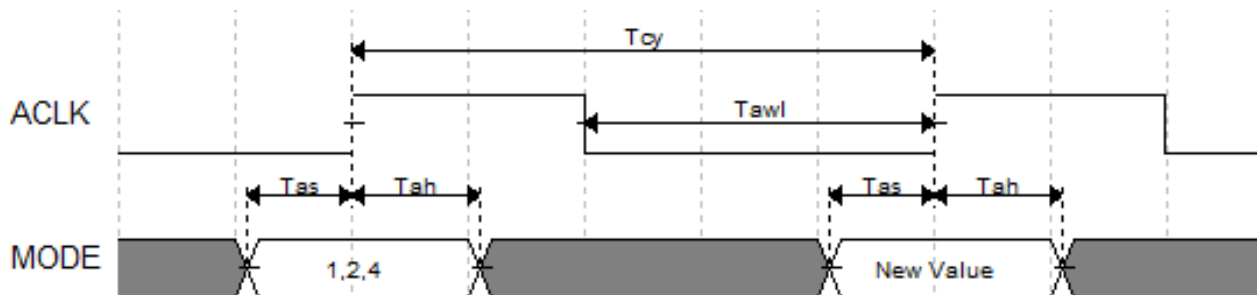
図 3-3 FLASH96K のページラッチ書き込みモード



注記：

ページラッチ書き込みモードの場合、SEQ=0、MODE=0000 です。

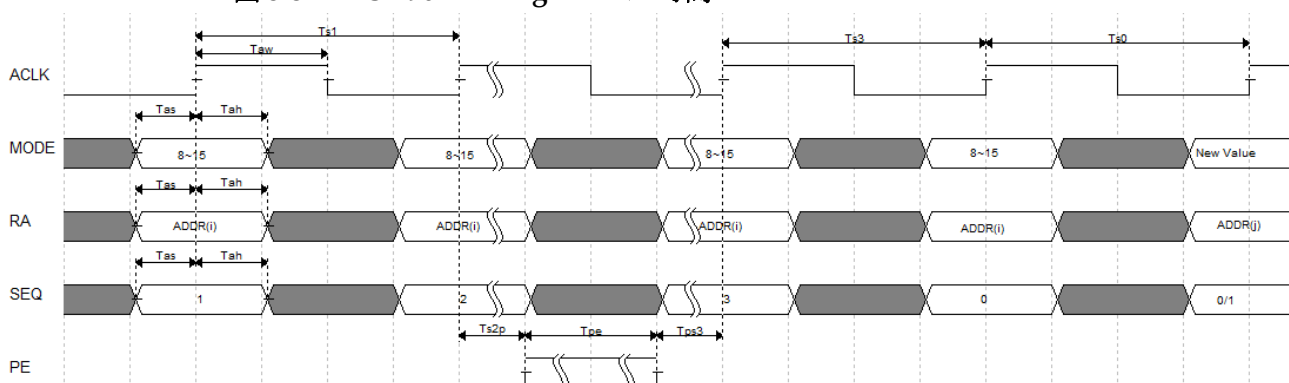
図 3-4 FLASH96K のページラッチ・クリア・モード



注記：

プレプログラミング・ビットのセットのタイミングと、データをすべてのページに書き込むタイミングと、ページラッチ・クリアのタイミングのパラメータは同じで、MODE 値だけが異なります。

図 3-5 FLASH96K の High レベル時間



3.2 FLASH64KZ

プリミティブの紹介

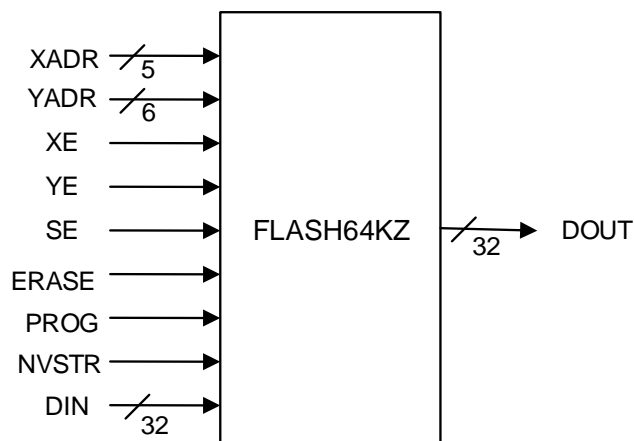
FLASH64KZ(64K ビット User Flash)の記憶領域は 64K ビットです。この Flash の幅と深さは構成できません。不揮発性メモリです。初期値機能はサポートされていません。

FLASH64KZ には以下の特徴があります。

- 10,000 回の書き込みサイクル
- 容量 : 32 行*64 列*32 ビット = 64 K ビット
- 10 年以上のデータ保持時間(+85°C)
- ページ消去をサポート : 2,048 バイト
- 高速なページ消去/書き込み
- アクセス時間 : 最大 25ns
- プログラミング時間 : 最大 16 μ s
- ページ消去時間 : 最大 120ms
- 電流
 - 読み出し : 2.19mA/25ns (V_{CC}) & 0.5mA/25ns (V_{CCX})(MAX)
 - 書き込み/消去 : 12/12mA(Max)

ポート図

図 3-6 FLASH64KZ のポート図



ポートの説明

表 3-7 FLASH64KZ のポートの説明

ポート	I/O	説明
DOUT[31:0]	出力	データ出力バス
DIN[31:0]	入力	データ入力バス
XADR[4:0]	入力	ページ内の特定の行を選択するために使用される X アドレスバス。
YADR[5:0]	入力	行内の特定の列の選択に使用される Y アドレスバス。
XE	入力	X アドレスイネーブル信号。XE が 0 の場合、すべての行アドレスが無効になります。
YE	入力	Y アドレスイネーブル信号。YE が 0 の場合、すべての列アドレスが無効になります。
SE	入力	センスアンパイネーブル信号、アクティブ High。
ERASE	入力	消去信号、アクティブ High。
PROG	入力	書き込み信号、アクティブ High。
NVSTR	入力	Flash データストレージ信号、アクティブ High。

構成モード

GW1NZ シリーズ FPGA 製品の User Flash は、汎用モードと低消費電力モードがあります。FLASH64KZ は汎用モードの User Flash です。

FLASH64KZ はデフォルトでオンになっており、電源を投入した後、消去/読み出しと書き込みなどの通常の操作を実行できます。オフ状態への切り替えはサポートされていません。

動作モード

表 3-8 FLASH64KZ のユーザーモードの真理値表

モード	XE	YE	SE	PROG	ERASE	NVSTR
読み出しモード	H	H	H	L	L	L
書き込みモード	H	H	L	H	L	H
ページ消去モード	H	L	L	L	H	H

注記：

「H」と「L」は、それぞれ High レベルと Low レベルを示します。

プリミティブのインスタンス化

プリミティブを直接インスタンス化するか、IP Core Generator で生成できます。詳しくは、[4 IP の呼び出し](#)を参照してください。

Verilog でのインスタンス化：

```
FLASH64KZ flash64kz_inst(
    .XADR(xadr[4:0]),
    .YADR(yadr[5:0]),
    .XE(xe),
    .YE(ye),
    .SE(se),
    .ERASE(erase),
    .PROG(prog),
    .NVSTR(nvstr),
    .DIN(din[31:0]),
    .DOUT(dout[31:0])
);
```

VHDL でのインスタンス化：

```
COMPONENT FLASH64KZ
PORT(
    XADR:IN std_logic_vector(4 downto 0);
    YADR:IN std_logic_vector(5 downto 0);
    XE:IN std_logic;
    YE:IN std_logic;
    SE:IN std_logic;
    ERASE:IN std_logic;
```

```

        PROG:IN std_logic;
        NVSTR:IN std_logic;
        DIN:IN std_logic_vector(31 downto 0);
        DOUT:OUT std_logic_vector(31 downto 0)
    );
END COMPONENT;
 uut: FLASH64KZ
    PORT MAP (
        XADR=>xadr,
        YADR=>yadr,
        XE=>xe,
        YE=>ye,
        SE=>se,
        ERASE=>erase,
        PROG=>prog,
        NVSTR=>nvstr,
        DIN=>din,
        DOUT=>dout
    );

```

タイミングパラメータ^{[4][5][6]}

表 3-9 FLASH64KZ のタイミングパラメータ

ユーザーモード	パラメータ	記号	最小値	最大値	単位
アクセス時間 ^[2]	WC1	$T_{acc}^{[3]}$	-	25	ns
	TC		-	22	ns
	BC		-	21	ns
	LT		-	21	ns
	WC		-	25	ns
書き込み/消去からデータ保存のセットアップ時間		T_{nvs}	5	-	μs
データ保存のホールド時間		T_{nh}	5	-	μs
データ保存のホールド時間(マス消去)		T_{nh1}	100	-	μs
データ保存から書き込みのセットアップ時間		T_{pgs}	10	-	μs
プログラミングのホールド時間		T_{pgh}	20	-	ns
プログラミング時間		T_{prog}	8	16	μs

ユーザーモード	パラメータ	記号	最小値	最大値	単位
	書き込みの準備時間	T_{wpr}	>0	-	ns
	書き込みのホールド時間	T_{whd}	>0	-	ns
	制御信号から書き込み/消去までのセットアップ時間	T_{cps}	-10	-	ns
	SE から読み出しまでのセットアップ時間	T_{as}	0.1	-	ns
	SE パルスの High レベル時間	T_{pws}	5	-	ns
	アドレス/データのセットアップ時間	T_{ads}	20	-	ns
	アドレス/データのホールド時間	T_{adh}	20	-	ns
	データのホールド時間	T_{dh}	0.5	-	ns
読み出しモードのアドレスのホールド時間 ^[3]	WC1	T_{ah}	25	-	ns
	TC	-	22	-	ns
	BC	-	21	-	ns
	LT	-	21	-	ns
	WC	-	25	-	ns
	SE パルスの Low レベル時間	T_{nws}	2	-	ns
	リカバリー時間	T_{rcv}	10	-	μs
	データ保持時間	T_{hv} ^[4]	-	6	ms
	消去時間	T_{erase}	100	120	ms
	マス消去時間	T_{me}	100	120	ms
	パワーダウンから待機モードまでの Wake-up 時間	T_{wk_pd}	7	-	μs
	待機のホールド時間	T_{sbh}	100	-	ns
	V _{cc} のセットアップ時間	T_{ps}	0	-	ns
	V _{ccx} のホールド時間	T_{ph}	0	-	ns

注記：

- ^[1]これらの設定値は変更される場合があります。
- ^[2]これらの数値はシミュレーションデータに過ぎません。
- ^[3]信号 XADR、YADR、XE、及び YE が有効になった後、T_{acc}の開始時間は SE 信号の立ち上がりエッジになります。読み出されたデータ DOUT は、次の有効な読み出しまで保存されます。
- ^[4]T_{hv}(Cumulative program high voltage period)は書き込みの開始から次のデータ消去までの累積時間です。次の消去前に同じアドレスに 2 回書き込むことはできません。次の消去前に同じメモリユニットに 2 回書き込むことはできません。このような制限は安全のためです。
- ^[5]すべての波形には、1ns の立ち上がりエッジ時間と 1ns の立ち下がりエッジ時間があります。
- ^[6]制御信号 XADR、YADR、XE、及び YE は少なくとも T_{acc}ホールドする必要がある、T_{acc}は SE の立ち上がりエッジから開始します。

タイミング図

図 3-7 FLASH64KZ の読み出しタイミング

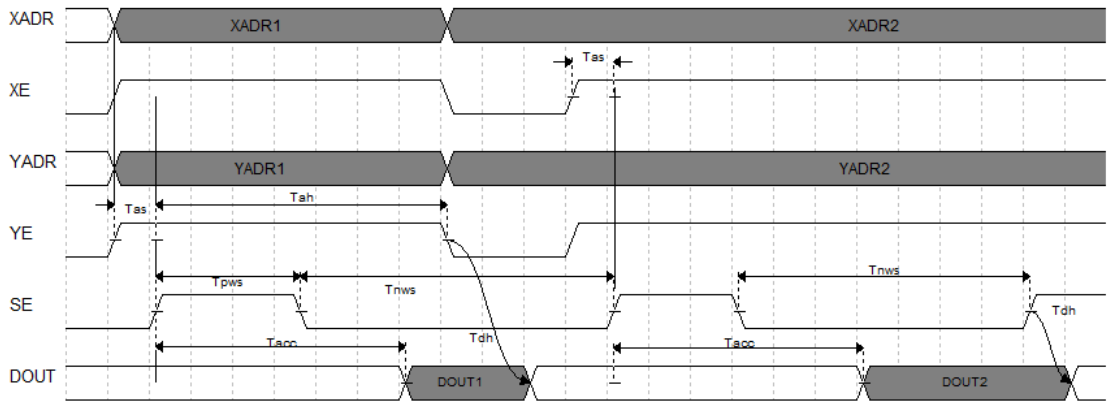


図 3-8 FLASH64KZ の書き込みタイミング

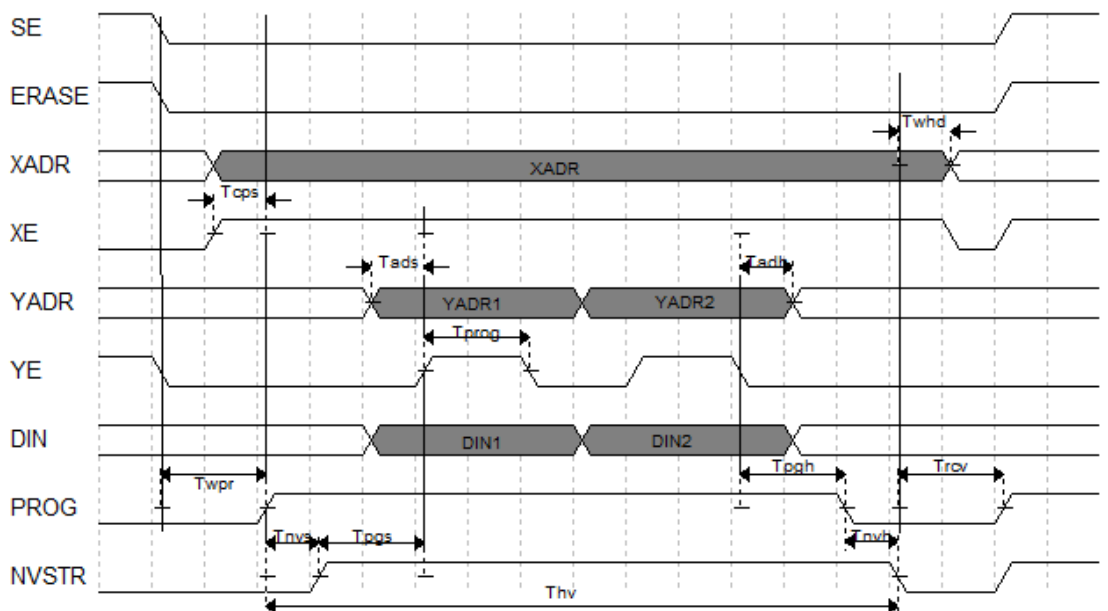
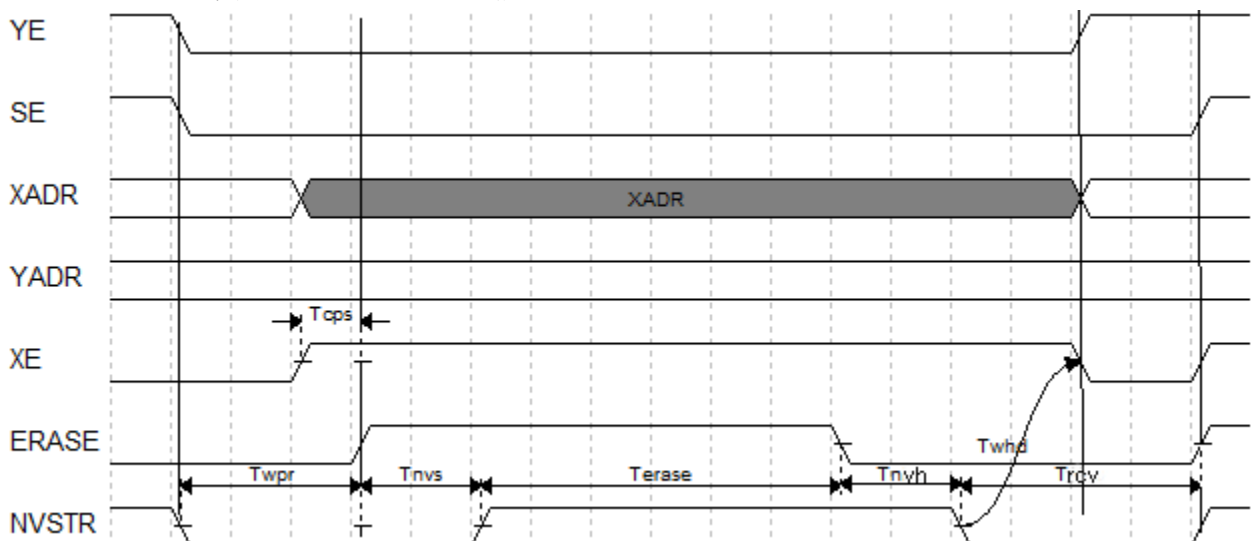


図 3-9 FLASH64KZ の消去タイミング



3.3 FLASH64K

プリミティブの紹介

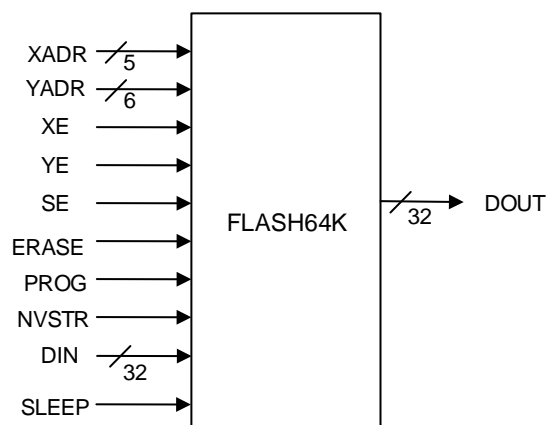
FLASH64K(64K ビット User Flash)の記憶領域は 64K ビットです。この Flash の幅と深さは構成できません。不揮発性メモリです。初期値機能はサポートされていません。FLASH64K にはスリープモードがあります。SLEEP 信号が High の場合、FLASH はスリープモードに入ります。

FLASH64K には以下の特徴があります：

- 10,000 回の書き込みサイクル
- 容量：32 行*64 列*32 ビット = 64 K ビット
- 10 年以上のデータ保持時間(+85℃)
- ページ消去をサポート：2,048 バイト
- 高速なページ消去/書き込み
- アクセス時間：最大 25ns
- プログラミング時間：最大 16 μ s
- ページ消去時間：最大 120ms
- 電流
 - 読み出し：2.19mA/25ns (V_{CC}) & 0.5mA/25ns (V_{CCX})(MAX)
 - 書き込み/消去：12/12mA(Max)

ポート図

図 3-10 FLASH64K のポート図



ポートの説明

表 3-10 FLASH64K のポートの説明

ポート	I/O	説明
DOUT[31:0]	出力	データ出力バス。
DIN[31:0]	入力	データ入力バス

ポート	I/O	説明
XADR[4:0]	入力	ページ内の特定の行を選択するために使用される X アドレスバス。
YADR[5:0]	入力	行内の特定の列の選択に使用される Y アドレスバス。
XE	入力	X アドレスイネーブル信号。XE が 0 の場合、すべての行アドレスが無効になります。
YE	入力	Y アドレスイネーブル信号。YE が 0 の場合、すべての列アドレスが無効になります。
SE	入力	センスアンパイネーブル信号、アクティブ High。
ERASE	入力	消去信号、アクティブ High。
PROG	入力	書き込み信号、アクティブ High。
NVSTR	入力	Flash データストレージ信号、アクティブ High。
SLEEP	入力	低消費電力 User Flash 状態の切り替え信号。 ● High : オン ● Low : オフ

構成モード

GW1NZ シリーズ FPGA 製品の User Flash は、汎用モードと低消費電力モードがあります。FLASH64K は低消費電力モードの User Flash です。

FLASH64K はデフォルトでオフになっており、消費電力を効果的に削減できます。SLEEP ピンを制御することでオン/オフを動的に切り替えることができます。オンに切り替えると、FLASH64KZ のように消去/読み出しと書き込みが可能です。

動作モード

表 3-11 FLASH64K のユーザーモードの真理値表

モード	XE	YE	SE	PROG	ERASE	NVSTR
読み出しモード	H	H	H	L	L	L
書き込みモード	H	H	L	H	L	H
ページ消去モード	H	L	L	L	H	H

注記：

「H」と「L」は、それぞれ High レベルと Low レベルを示します。

プリミティブのインスタンス化

プリミティブを直接インスタンス化するか、IP Core Generator で生成できます。詳しくは、[4 IP の呼び出し](#)を参照してください。

Verilog でのインスタンス化：

```
FLASH64K flash64k_inst(
```



```
.XADR(xadr[4:0]),  
.YADR(yadr[5:0]),  
.XE(xe),  
.YE(ye),  
.SE(se),  
.ERASE(erase),  
.PROG(prog),  
.NVSTR(nvstr),  
.DIN(din[31:0]),  
.SLEEP(sleep),  
.DOUT(dout[31:0])  
);
```

VHDL でのインスタンス化 :

```
COMPONENT FLASH64K  
  PORT(  
    XADR:IN std_logic_vector(4 downto 0);  
    YADR:IN std_logic_vector(5 downto 0);  
    XE:IN std_logic;  
    YE:IN std_logic;  
    SE:IN std_logic;  
    ERASE:IN std_logic;  
    PROG:IN std_logic;  
    NVSTR:IN std_logic;  
    DIN:IN std_logic_vector(31 downto 0);  
    SLEEP:IN std_logic;  
    DOUT:OUT std_logic_vector(31 downto 0)  
  );  
END COMPONENT;  
 uut: FLASH64K  
  PORT MAP (  
    XADR=>xadr,  
    YADR=>yadr,  
    XE=>xe,  
    YE=>ye,  
    SE=>se,
```

```
ERASE=>erase,  
PROG=>prog,  
NVSTR=>nvstr,  
DIN=>din,  
SLEEP=>sleep,  
DOUT=>dout
```

```
);
```

タイミングパラメータとタイミング図

FLASH64K のタイミングパラメータとタイミング図は FLASH64KZ と同様であるため、そのタイミングパラメータおよびタイミング図については、[3.2 FLASH64KZ](#) を参照してください。

3.4 FLASH256K

プリミティブの紹介

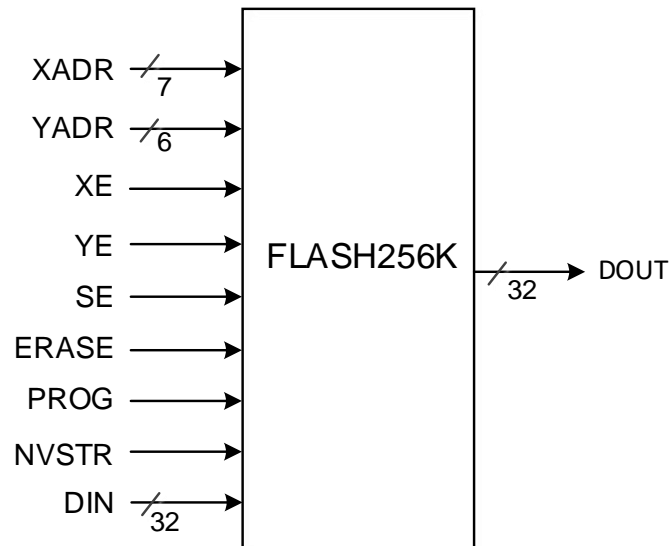
FLASH256K(256K ビット User Flash)の記憶領域は 256K ビットです。この Flash の幅と深さは構成できません。不揮発性メモリです。初期値機能はサポートされていません。

FLASH256K は、行メモリユニットと列メモリユニットで構成されています。1 行メモリユニットは 64 列メモリユニットで構成されています。1 列メモリユニットの容量は 32 ビットで、1 行メモリユニットの容量は $64 \times 32 = 2048$ ビットです。ページ消去がサポートされており、1 ページの容量は 2048 バイトです。すなわち、1 ページには 8 行があります。その特徴は以下に示す通りです：

- 10,000 回の書き込みサイクル
- 10 年以上のデータ保持時間(+85°C)
- データ幅：32 ビット
- 容量：128 行*64 列*32 ビット = 256 K ビット
- ページ消去をサポート：2,048 バイト
- 高速ページ消去/書き込み/読み出し
- アクセス時間：最大 25ns
- プログラミング時間：最大 16 μ s
- ページ消去時間：最大 120ms
- 電流
 - 読み出し電流/継続時間：2.19mA/25ns (V_{CC}) & 0.5mA/25ns (V_{CCX})(MAX)
 - 書き込み/消去：12/12mA(Max)

ポート図

図 3-11 FLASH256K のポート図



ポートの説明

表 3-12 FLASH256K のポートの説明

ポート	I/O	説明
DOUT[31:0]	出力	データ出力バス。
DIN[31:0]	入力	データ入力バス
XADR[6:0]	入力	X アドレスバス。XADR [n : 3]は特定のページを選択するために使用されます。XADR [2 : 0]は1 ページ内の特定の行を選択するために使用されます。1 ページは 8 行で構成され、1 行は 64 列で構成されています。
YADR[5:0]	入力	行から特定の列を選択するために使用される Y アドレスバス。1 行は 64 列で構成されています。
XE	入力	X アドレスイネーブル信号。XE が 0 の場合、すべての行アドレスが無効になります。
YE	入力	Y アドレスイネーブル信号。YE が 0 の場合、すべての列アドレスが無効になります。
SE	入力	センスアンパイネーブル信号、アクティブ High。
PROG	入力	書き込み信号、アクティブ High。
ERASE	入力	消去信号、アクティブ High。
NVSTR	入力	Flash データストレージ信号、アクティブ High。

動作モード

表 3-13 FLASH256K のユーザーモードの真理値表

モード	XE	YE	SE	PROG	ERASE	NVSTR
読み出しモード	H	H	H	L	L	L
書き込みモード	H	H	L	H	L	H
ページ消去モード	H	L	L	L	H	H

注記：

「H」と「L」は、それぞれ High レベルと Low レベルを示します。

プリミティブのインスタンス化

プリミティブを直接インスタンス化するか、IP Core Generator で生成できます。詳しくは、[4 IP の呼び出し](#)を参照してください。

Verilog でのインスタンス化：

```
FLASH256K flash256k_inst(
    .XADR(xadr[6:0]),
    .YADR(yadr[5:0]),
    .XE(xe),
    .YE(ye),
    .SE(se),
    .ERASE(erase),
    .PROG(prog),
    .NVSTR(nvstr),
    .DIN(din[31:0]),
    .DOUT(dout[31:0])
);
```

VHDL でのインスタンス化：

```
COMPONENT FLASH256K
PORT(
    DIN:IN std_logic_vector(31 downto 0);
    XADR:IN std_logic_vector(6 downto 0);
    YADR:IN std_logic_vector(5 downto 0);
    XE:IN std_logic;
    YE:IN std_logic;
    SE:IN std_logic;
```

```

        ERASE:IN std_logic;
        PROG:IN std_logic;
        NVSTR:IN std_logic;
        DOUT:OUT std_logic_vector(31 downto 0)
    );
END COMPONENT;
 uut: FLASH256K
    PORT MAP (
        DIN=>din,
        XADR=>xadr,
        YADR=>yadr,
        XE=>xe,
        YE=>ye,
        SE=>se,
        ERASE=>erase,
        PROG=>prog,
        NVSTR=>nvstr,
        DOUT=>dout
    );

```

タイミングパラメータとタイミング図

FLASH256K のタイミングパラメータとタイミング図は FLASH64KZ と同様であるため、そのタイミングパラメータおよびタイミング図については、[3.2 FLASH64KZ](#) を参照してください。

3.5 FLASH96KA

プリミティブの紹介

FLASH96KA(96K ビット User Flash)の記憶領域は 96K ビットです。

そのタイミングは FLASH256K と同様です。FLASH256K に比べて、スリープ機能が追加され、SLEEP 信号が High になると FLASH96KA はスリープモードに入ります。この Flash の幅と深さは構成できません。不揮発性メモリです。初期値機能はサポートされていません。

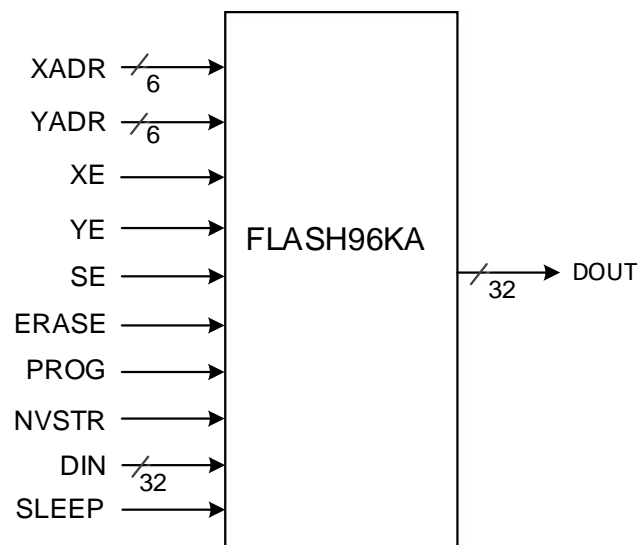
FLASH96KA は、行メモリユニットと列メモリユニットで構成されています。1 行メモリユニットは 64 列メモリユニットで構成されています。1 列メモリユニットの容量は 32 ビットで、1 行メモリユニットの容量は $64 \times 32 = 2048$ ビットです。ページ消去がサポートされ、ページあたりの容量は 2048 バイトです。すなわち、1 ページには 8 行があります。そ

の特性は以下に示す通りです：

- 10,000 回の書き込みサイクル
- 10 年以上のデータ保持時間(+85°C)
- データ幅：32 ビット
- 容量：48 行*64 列*32 ビット = 96 K ビット
- ページ消去をサポート：2,048 バイト
- 高速ページ消去/書き込み/読み出し
- アクセス時間：最大 25ns
- プログラミング時間：最大 16 μ s
- ページ消去時間：最大 120ms
- 電流
 - 読み出し電流/継続時間：2.19mA/25ns (V_{CC}) & 0.5mA/25ns (V_{CCX})(MAX)
 - 書き込み/消去：12/12mA(Max)

ポート図

図 3-12 FLASH96KA のポート図



ポートの説明

表 3-14 FLASH96KA のポートの説明

ポート	I/O	説明
DOUT[31:0]	出力	データ出力バス。
DIN[31:0]	入力	データ入力バス

ポート	I/O	説明
XADR[5:0]	入力	X アドレスバス。XADR [n : 3]は特定のページを選択するために使用されます。XADR [2 : 0]は1ページ内の特定の行を選択するために使用されます。1 ページは 8 行で構成され、1 行は 64 列で構成されています。
YADR[5:0]	入力	行から特定の列を選択するために使用される Y アドレスバス。1 行は 64 列で構成されています。
XE	入力	X アドレスイネーブル信号。XE が 0 の場合、すべての行アドレスが無効になります。
YE	入力	Y アドレスイネーブル信号。YE が 0 の場合、すべての列アドレスが無効になります。
SE	入力	センスアンパイネーブル信号、アクティブ High。
PROG	入力	書き込み信号、アクティブ High。
ERASE	入力	消去信号、アクティブ High。
NVSTR	入力	Flash データストレージ信号、アクティブ High。
SLEEP	入力	スリープ信号。アクティブ High。

動作モード

表 3-15 FLASH96KA のユーザーモードの真理値表

モード	XE	YE	SE	PROG	ERASE	NVSTR
読み出しモード	H	H	H	L	L	L
書き込みモード	H	H	L	H	L	H
ページ消去モード	H	L	L	L	H	H

注記：

「H」と「L」は、それぞれ High レベルと Low レベルを示します。

プリミティブのインスタンス化

プリミティブを直接インスタンス化するか、IP Core Generator で生成できます。詳しくは、[4 IP の呼び出し](#)を参照してください。

Verilog でのインスタンス化：

```
FLASH96KA FLASH96KA_inst(
    .XADR(xadr[5:0]),
    .YADR(yadr[5:0]),
    .XE(xe),
    .YE(ye),
```



```
.SE(se),  
.ERASE(erase),  
.PROG(prog),  
.NVSTR(nvstr),  
.DIN(din[31:0]),  
.SLEEP(sleep),  
.DOUT(dout[31:0])  
);
```

VHDL でのインスタンス化 :

```
COMPONENT FLASH96KA  
  PORT(  
    DIN:IN std_logic_vector(31 downto 0);  
    XADR:IN std_logic_vector(5 downto 0);  
    YADR:IN std_logic_vector(5 downto 0);  
    XE:IN std_logic;  
    YE:IN std_logic;  
    SE:IN std_logic;  
    ERASE:IN std_logic;  
    PROG:IN std_logic;  
    NVSTR:IN std_logic;  
    SLEEP:IN std_logic;  
    DOUT:OUT std_logic_vector(31 downto 0)  
  );  
END COMPONENT;  
 uut: FLASH96KA  
  PORT MAP (  
    DIN=>din,  
    XADR=>xadr,  
    YADR=>yadr,  
    XE=>xe,  
    YE=>ye,  
    SE=>se,  
    ERASE=>erase,  
    PROG=>prog,  
    NVSTR=>nvstr,
```

```
SLEEP=>sleep,  
DOUT=>dout  
);
```

タイミングパラメータとタイミング図

FLASH96KA のタイミングパラメータとタイミング図は FLASH64KZ と同様であるため、そのタイミングパラメータおよびタイミング図については、[3.2 FLASH64KZ](#) を参照してください。

3.6 FLASH608K

プリミティブの紹介

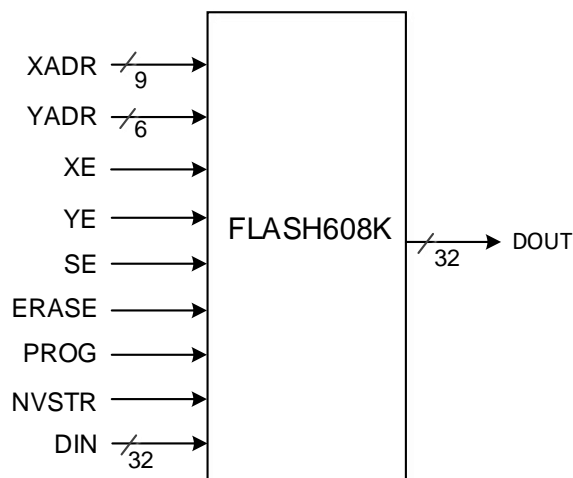
FLASH608K(608K ビット User Flash)の記憶領域は 608K ビットです。この Flash の幅と深さは構成できません。不揮発性メモリです。初期値機能はサポートされていません。

FLASH608K は、行メモリユニットと列メモリユニットで構成されています。1 行メモリユニットは 64 列メモリユニットで構成されています。1 列メモリユニットの容量は 32 ビットで、1 行メモリユニットの容量は $64 \times 32 = 2048$ ビットです。ページ消去がサポートされ、ページあたりの容量は 2048 バイトです。すなわち、1 ページには 8 行があります。その特性は以下に示す通りです：

- 10,000 回の書き込みサイクル
- 10 年以上のデータ保持時間(+85°C)
- データ幅：32 ビット
- 容量：304 行*64 列*32 ビット = 608 K ビット
- ページ消去をサポート：2,048 バイト
- 高速ページ消去/書き込み/読み出し
- アクセス時間：最大 25ns
- プログラミング時間：最大 16 μ s
- ページ消去時間：最大 120ms
- 電流
 - 読み出し電流/継続時間：2.19mA/25ns (Vcc) & 0.5mA/25ns (Vccx)(MAX)
 - 書き込み/消去：12/12mA(Max)

ポート図

図 3-13 FLASH608K のポート図



ポートの説明

表 3-16 FLASH608K のポートの説明

ポート	I/O	説明
DOUT[31:0]	出力	データ出力バス。
DIN[31:0]	入力	データ入力バス
XADR[8:0]	入力	X アドレスバス。XADR [n : 3]は特定のページを選択するために使用されます。XADR [2 : 0]は 1 ページ内の特定の行を選択するために使用されます。1 ページは 8 行で構成され、1 行は 64 列で構成されています。
YADR[5:0]	入力	行から特定の列を選択するために使用される Y アドレスバス。1 行は 64 列で構成されています。
XE	入力	X アドレスイネーブル信号。XE が 0 の場合、すべての行アドレスが無効になります。
YE	入力	Y アドレスイネーブル信号。YE が 0 の場合、すべての列アドレスが無効になります。
SE	入力	センスアンプイネーブル信号、アクティブ High。
PROG	入力	書き込み信号、アクティブ High。
ERASE	入力	消去信号、アクティブ High。
NVSTR	入力	Flash データストレージ信号、アクティブ High。

動作モード

表 3-17 FLASH608K のユーザーモードの真理値表

モード	XE	YE	SE	PROG	ERASE	NVSTR
読み出しモード	H	H	H	L	L	L

モード	XE	YE	SE	PROG	ERASE	NVSTR
書き込みモード	H	H	L	H	L	H
ページ消去モード	H	L	L	L	H	H

注記：

「H」と「L」は、それぞれ High レベルと Low レベルを示します。

プリミティブのインスタンス化

プリミティブを直接インスタンス化するか、IP Core Generator で生成できます。詳しくは、[4 IP の呼び出し](#)を参照してください。

Verilog でのインスタンス化：

```
FLASH608K flash608k_inst(
    .XADR(xadr[8:0]),
    .YADR(yadr[5:0]),
    .XE(xe),
    .YE(ye),
    .SE(se),
    .ERASE(erase),
    .PROG(prog),
    .NVSTR(nvstr),
    .DIN(din[31:0]),
    .DOUT(dout[31:0])
);
```

VHDL でのインスタンス化：

```
COMPONENT FLASH608K
    PORT(
        DIN:IN std_logic_vector(31 downto 0);
        XADR:IN std_logic_vector(8 downto 0);
        YADR:IN std_logic_vector(5 downto 0);
        XE:IN std_logic;
        YE:IN std_logic;
        SE:IN std_logic;
        ERASE:IN std_logic;
        PROG:IN std_logic;
        NVSTR:IN std_logic;
        DOUT:OUT std_logic_vector(31 downto 0)
```

```
);  
END COMPONENT;  
uut: FLASH608K  
  PORT MAP (  
    DIN=>din,  
    XADR=>xadr,  
    YADR=>yadr,  
    XE=>xe,  
    YE=>ye,  
    SE=>se,  
    ERASE=>erase,  
    PROG=>prog,  
    NVSTR=>nvstr,  
    DOUT=>dout  
  );
```

タイミングパラメータとタイミング図

FLASH608K のタイミングパラメータとタイミング図は FLASH64KZ と同様であるため、そのタイミングパラメータおよびタイミング図については、[3.2 FLASH64KZ](#) を参照してください。

4 IP の呼び出し

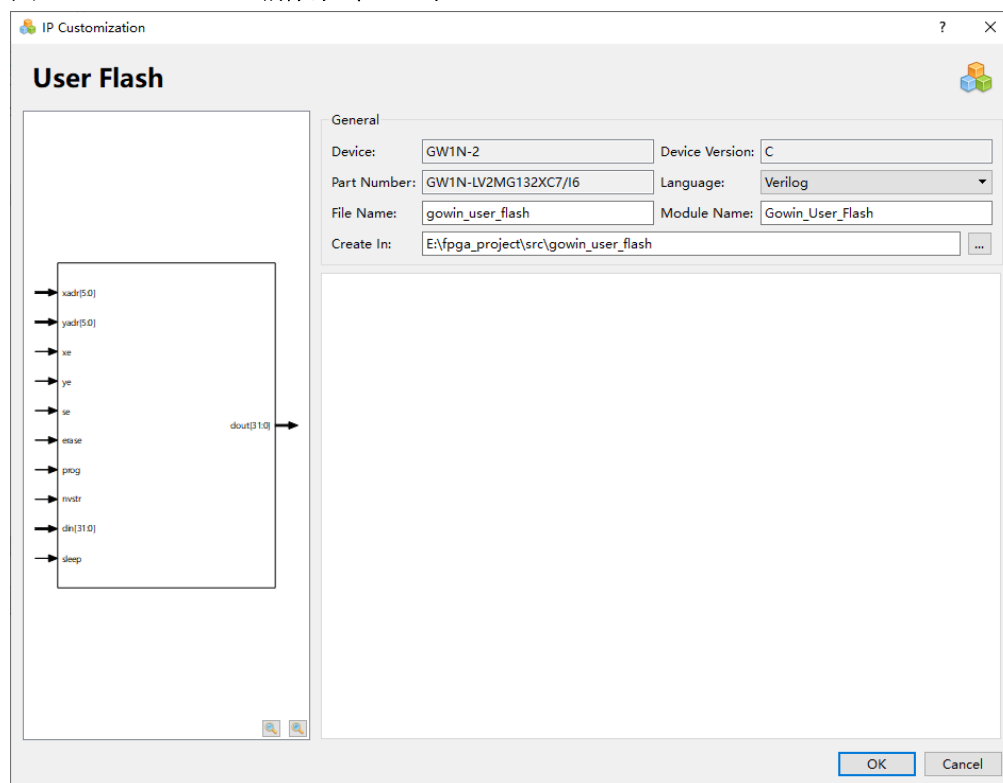
IP Core Generator はハードコアの User Flash とソフトコアの Gowin Flash Controller IP の生成をサポートします。Gowin Flash Controller IP の機能および GUI での呼び出しについては、『Gowin Flash Controller IP ユーザーガイド([JPUG901](#))』を参照してください。IP Core Generator > Hard Module > User Flash を選択して User Flash ハードコア IP を構成してください。

IP Core Generator の画面で「User Flash」をクリックすると右側に User Flash の概要が表示されます。

IP の構成

IP Core Generator 画面で「User Flash」をダブルクリックすると、User Flash の「IP Customization」ウィンドウがポップアップします。このウィンドウには「General」構成ボックスとポート図があります(図 4-1)。

図 4-1 User Flash の構成ウィンドウ



1. 「General」構成ボックス：IP ファイルの構成に使用されます。
 - Device：対象デバイス。
 - Device Version：デバイスのバージョン。
 - Part Number：部品番号。
 - Language：IP を実現するハードウェア記述言語。右側のドロップダウン・リストからターゲット言語(Verilog または VHDL)を選択します。
 - Module Name：生成される IP ファイルのモジュール名。右側のテキストボックスで編集できます。Module Name をプリミティブ名と同じにすることはできません。同じである場合、エラーが報告されます。
 - File Name：生成される IP ファイルのファイル名。右側のテキストボックスで再編集できます。
 - Create In：生成される IP ファイルのパス。右側のテキストボックスでパスを直接編集するか、テキストボックスの右側にある選択ボタンを使用してパスを選択できます。
2. ポート図：User Flash のポート図はデバイスによって異なります(図 4-1)。

生成されるファイル

IP の構成が完了すると、「File Name」によって命名された 3 つのファイルが生成されます：

- 「gowin_user_flash.v」は完全な verilog モジュールです。
- 「gowin_user_flash_tmp.v」は IP のテンプレートファイルです。
- 「gowin_user_flash.ipc」は IP の構成ファイルです。

注記：

VHDL が設計の言語として選択されている場合、生成される最初の 2 つのファイル名のサフィックスは.vhd になります。

