Gowin User Flash

User Guide

UG295-1.4.3E, 02/22/2023

Disclaimer

GOWINSEMI assumes no liability and provides no warranty (either expressed or implied) and is not responsible for any damage incurred to your hardware, software, data, or property resulting from usage of the materials or intellectual property except as outlined in the GOWINSEMI Terms and Conditions of Sale. All information in this document should be treated as preliminary. GOWINSEMI may make changes to this document at any time without prior notice. Anyone relying on this documentation should contact GOWINSEMI for the current documentation and errata.

Revision History

Date	Version	Description
08/24/2020	1.0E	Initial version published.
01/12/2021	1.1E	FLASH256KA added.
07/14/2021	1.2E	● GW1N-2B, GW1N-1P5, GW1N-1P5B, GW1NR-2B added. ● Figures updated and "Help" information removed in chapter 4 IP Generation. ● FLASH256KA modified to FLASH96KA. ● FLASH96KA description updated.
11/14/2021	1.3E	● Devices supported updated. ● Clock frequency updated. ● Some descriptions updated.
11/04/2022	1.4E	GW1NS-2, GW1NS-2C, GW1NSE-2C, GW1NSR-2, GW1NSR-2C removed.
01/05/2023	1.4.1E	The configuration box "File" modified to "General" and "Device Version" option added on the IP interface.
02/03/2023	1.4.2E	● Clock frequency updated to access time. ● The descriptions of timing parameters updated.
02/22/2023	1.4.3E	The timing parameter & diagram descriptions of FLASH64K, FLASH256K, FLASH96KA, and FLASH608K optimized.

Contents

Contents i
List of Figures ii
List of Tables iii
1 About This Guide 1
1.1 Purpose 1
1.2 Related Documents 1
1.3 Terminology and Abbreviations 1
1.4 Support and Feedback 1
2 Overview 2
3 Primitive Introduction 3
3.1 FLASH96K 3
3.2 FLASH64KZ 10
3.3 FLASH64K 15
3.4 FLASH256K 19
3.5 FLASH96KA 22
3.6 FLASH608K 25
4 IP Generation 29

List of Figures

Figure 3-1 FLASH96K Ports Diagram 4
Figure 3-2 FLASH96K Read Mode 9
Figure 3-3 FLASH96K Write Page Latches Mode 9
Figure 3-4 FLASH96K Clear Page Latches Mode 10
Figure 3-5 FLASH96K High Level Period 10
Figure 3-6 FLASH64KZ Ports Diagram 11
Figure 3-7 FLASH64KZ Read Operation Timing 14
Figure 3-8 FLASH64KZ Write Operation Timing 15
Figure 3-9 FLASH64KZ Erase Operation Timing 15
Figure 3-10 FLASH64K Diagram 16
Figure 3-11 FLASH256K Ports Diagram 19
Figure 3-12 FLASH96KA Ports Diagram 23
Figure 3-13 FLASH608K Ports Diagram 26
Figure 4-1 IP Customization of User Flash 29

List of Tables

Table 1-1 Abbreviations and Terminology 1
Table 3-1 Devices Supported 3
Table 3-2 FLASH96K Port Description 4
Table 3-3 FLASH96K Output Bit Width Option 5
Table 3-4 FLASH96K Input Bit Width Option 5
Table 3-5 FLASH96K Operation Mode Option 5
Table 3-6 FLASH96K Timing Parameters 8
Table 3-7 FLASH64KZ Port Description 11
Table 3-8 FLASH64KZ Truth Table in User Mode 11
Table 3-9 FLASH64KZ Timing Parameters 13
Table 3-10 FLASH64K Port Description 16
Table 3-11 FLASH64K Truth Table in User Mode 17
Table 3-12 FLASH256K Port Description 20
Table 3-13 FLASH256K Truth Table in User Mode 20
Table 3-14 FLASH96KA Port Description 23
Table 3-15 FLASH96KA Truth Table in User Mode 23
Table 3-16 FLASH608K Port Description 26
Table 3-17 FLASH608K Truth Table in User Mode 27

1 About This Guide

1.1 Purpose

This manual describes the function, primitives and usage of Gowin User Flash.

1.2 Related Documents

The latest user guides are available on GOWINSEMI Website. You can find the related documents at www.gowinsemi.com: IPUG901, Gowin Flash Controller IP User Guide.

1.3 Terminology and Abbreviations

The terminology and abbreviations used in this manual are as shown in Table 1-1.

Table 1-1 Abbreviations and Terminology

Abbreviations and Terminology	Meaning
FPGA	Field Programmable Gate Array
IP Core	Intellectual Property Core

1.4 Support and Feedback

Gowin Semiconductor provides customers with comprehensive technical support. If you have any questions, comments, or suggestions, please feel free to contact us directly by the following ways.

Website: www.gowinsemi.com

E-mail: support@gowinsemi.com

2 Overview

The FPGA products of Gowin LittleBee® family provide User Flash. Different series of devices support different sizes of Flash, including FLASH96K, FLASH64K, FLASH64KZ, FLASH256K, FLASH96KA and FLASH608K.

3 Primitive Introduction

The correspondence between User Flash primitives and devices supported are shown in Table 3-1.

Table 3-1 Devices Supported

Primitive	Devices Supported
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-1P5, GW1N-1P5B, GW1NR-2, GW1NR-2B
FLASH608K	GW1N-9, GW1N-9C, GW1NR-9, GW1NR-9C

Note!

The User Flash of GW1NS-4C, GW1NSR-4C, and GW1NSER-4C is provided for MCU.

3.1 FLASH96K

Primitive Introduction

User Flash (FLASH96K) has a size of 96 Kbits. The width and depth of registers are fixed and unconfigurable. Its width is 4 bytes (32 bits) and the address depth is 3K. It has non-volatile and power-off saving functions, but without initial value.

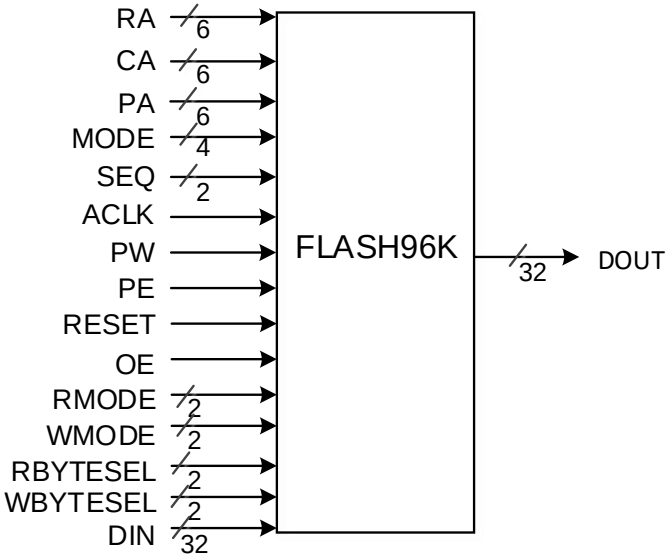
FLASH96K has the following features.

- 100,000 write cycles
- Data retention for more than 10 years (+85℃)
- 8/16/32 bits data-in and data-out
- Size: 48 rows * 64 columns * 32 bits = 96 Kbits
- Page size: 256 bytes
- 3 μA standby current

- Page write time: 8.2 ms

Port Diagram

Figure 3-1 FLASH96K Ports Diagram



Port Description

Table 3-2 FLASH96K Port Description

Port	I/O	Description
DOUT[31:0]	Output	Data output bus
DIN[31:0]	Input	Data input bus
RA[5:0]	Input	X address bus, used to select one row within memory block.
CA[5:0]	Input	Y address bus, used to select one column within memory block.
PA[5:0]	Input	Select one column of page latch address
MODE[3:0]	Input	Select operating mode
SEQ[1:0]	Input	Select sequence
ACLK	Input	Synchronization clock for read-write operations
PW	Input	Input clock for page latch data
RESET	Input	Reset signal, active-high
PE	Input	Charge pump enable
OE	Input	Data output enable
RMODE[1:0]	Input	Read data bit width selection
WMODE[1:0]	Input	Write data bit width selection
RBYTESEL[1:0]	Input	Read data byte selection
WBYTESEL[1:0]	Input	Write data bit width selection

Configuration Mode

You can select I/O bit width by read/ write modes and R/W byte selection singals. The correspondence between data bit width and control signal is shown in Table 3-3 and Table 3-4.

Table 3-3 FLASH96K Output Bit Width Option

RMOD[1:0]	RBYTESEL		DOUT			
	[1]	[0]	[31:24]	[23:16]	[15:8]	[7:0]
00	√	√	×	×	×	√
01	√	×	×	×	√	√
1X	×	×	√	√	√	√

Table 3-4 FLASH96K Input Bit Width Option

WMOD[1:0]	WBYTESEL		DIN			
	[1]	[0]	[31:24]	[23:16]	[15:8]	[7:0]
00	√	√	×	×	×	√
01	√	×	×	×	√	√
1X	×	×	√	√	√	√

Note!

"√" means valid input, and "×"means invalid input.

Operation Mode

You can set MODE [3: 0] to select different operation modes, as shown in Table 3-5.

Table 3-5 FLASH96K Operation Mode Option

MODE[3:0]	Description
0000	Read operation and page latch write-in
0001	Set the pre-programmed and reset automatically when the write operation starts.
0100	Clear page latch
1000	Erase page or row
1100	Write operation of page or row

- Read Operation

When MODE is set to "0000", the user flash enters into read operation mode at the rising edge of ACLK. Seq [1: 0] should be "00" for read operation mode. When the data access time (<= 38ns) is met, the data would be available on the output pin DOUT.

- Write Operation

The write operation of user flash includes five steps:

1. Clear page latch
2. Write data into the page latch

3. Preprogram the selected memory as virtual "1"
4. Erase the selected memory
5. Write page latch data into memory

After being erased, the data would be "0"; and after write operation, the data would be "1". An erased location "0" can be programmed to "1", but a programmed location"1" can not be programmed to "0", so erasing is always needed for a new write operation.

- Write Page Latch

Page latch can be regarded as one SRAM that will be wrote into Flash. Page latch write-in operation is controlled by PW signal, independent of ACLK. PA (Page Address) signal specifies the address to write to the page latches.

You should clear page latches before writing. Write Page latches one by one. Set MODE value as "0000" and MODE [1: 0] as "00". Page latch write and data read operation are completely independent.

- Clear Page Latch

Unlike page latch write, page latch clear is controlled by ACLK. When MODE is set as "0100", the FLASH enters to clear page latch mode at rising edge of ACLK. In this mode, ACLK [1: 0] should be "00" and page latch data will be cleared in one ACLK cycle.

- Erase and Write

The SEQ value should be 1-> 2 -> 3 -> 0 for erase and write operations, which require milliseconds. It is forbidden to write to the same page twice after an erasure operation.

Before erase and write operations, the selected memory needs to be written to the virtual "1" through the pre-program operation.

1. Set PEP (Pre-program) to MODE "0001".
2. Write to the selected at high-level, which takes hundreds of microseconds.

The steps after the pre-programmed MODE is set to "1100" are similar to the erase and write operation steps, both of which perform SEQ 1-> 2 -> 3 -> 0, as shown in Figure 3-5, but some timing is different (e.g., Tpe time in Table 3-6).

Primitive Instantiation

The primitives can be instantiated directly, or generated by the IP Core Generator. For the details, you can refer to Chapter 4 IP Generation.

Verilog Instantiation:

```
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 Instantiation:

```
    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
    );
```

Timing Parameters

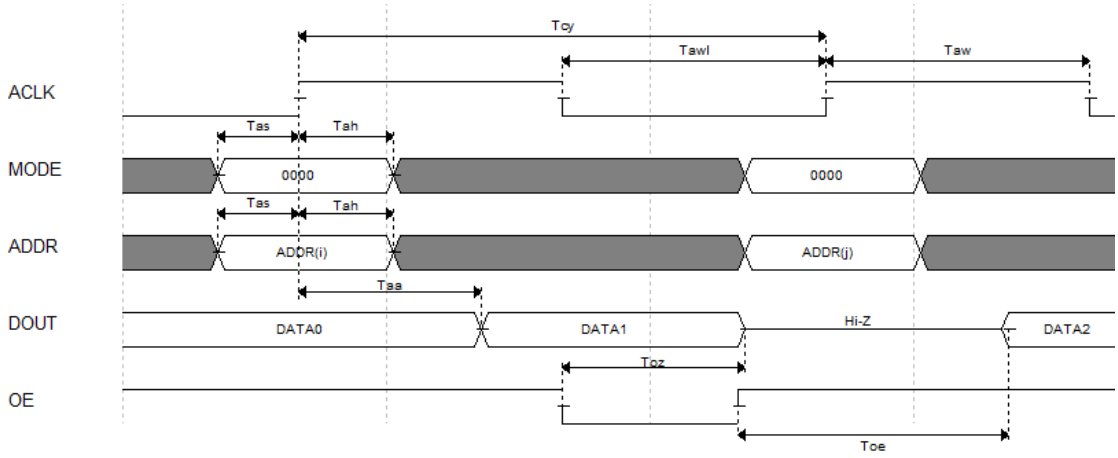
Table 3-6 FLASH96K Timing Parameters

Parameter	Description	Spec.			Unit
		Min.	Classical Value	Max.	
Taa	Data acquisition time	–	–	38	ns
Tcy	Read cycle	43	–	–	ns
Tawl	ACLK high-level time	10	–	–	ns
Tawl	ACLK low-level time	10	–	–	ns
Tas	Setup time	3	–	–	ns
Tah	Hold time	3	–	–	ns
Toz	Pull OE down to high resistance	–	–	2	ns
Toe	Pull OE up to DOUT	–	–	2	ns
Twcy	Write cycle	40	–	–	ns
Tpw	PW high-level time	16	–	–	ns
Tpwl	PW low-level time	16	–	–	ns
Tpas	Page address set up time	3	–	–	ns
Tpas	Page address hold time	3	–	–	ns

Parameter	Description	Spec.			Unit
		Min.	Classical Value	Max.	
Tds	Data set up time	16	–	–	ns
Tdh	Data hold-up time	3	–	–	ns
Ts0	SEQ0 cycle	6	–	–	μs
Ts1	SEQ1 cycle	15	–	–	μs
Ts2p	Set up time from ACLK to PE rising edge	5	–	10	μs
Ts3	SEQ3 cycle	5	–	10	μs
Tps3	Set up time from ACLK falling edge to ACLK	60	–		μs
Tpe	MODE=1000 erasure time	5.7	6	6.3	ms
	MODE=1100 Write operation time	1.9	2	2.1	ms
	MODE=11xx pre-program time	190	200	210	μs

Timing Diagrams

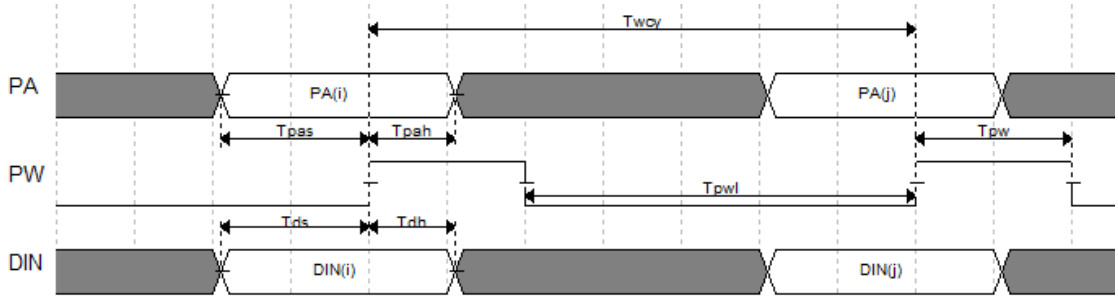
Figure 3-2 FLASH96K Read Mode



Note!

In read operation cycle, SEQ=0, ADDR including RA, CA, RMOD, and RBYTESEL.

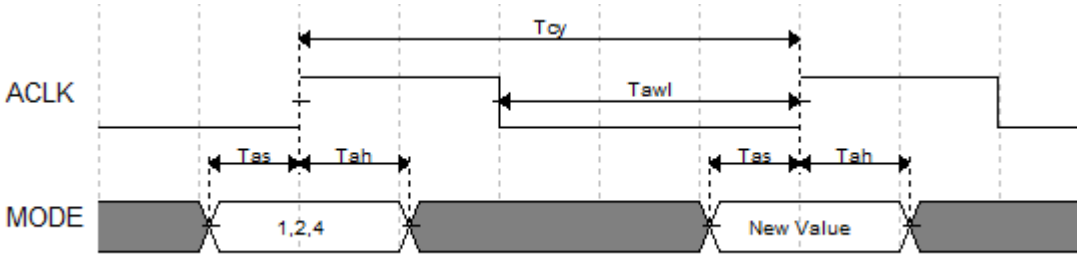
Figure 3-3 FLASH96K Write Page Latches Mode



Note!

In page latches write-in cycle, MODE=0 and MODE=0000.

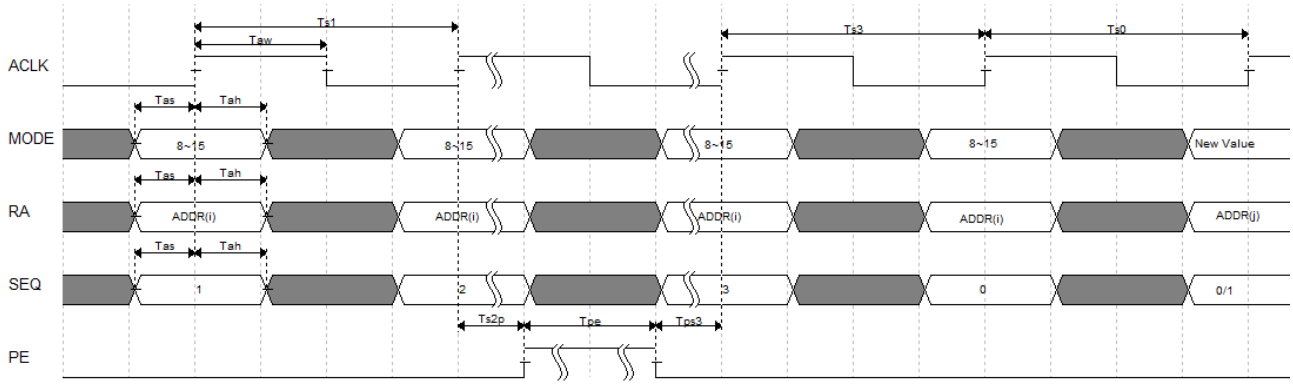
Figure 3-4 FLASH96K Clear Page Latches Mode



Note!

The timing parameters of setting PEP, writing to all pages, and clearing page latches are all the same, but the MODE values are different.

Figure 3-5 FLASH96K High Level Period



3.2 FLASH64KZ

Primitive Introduction

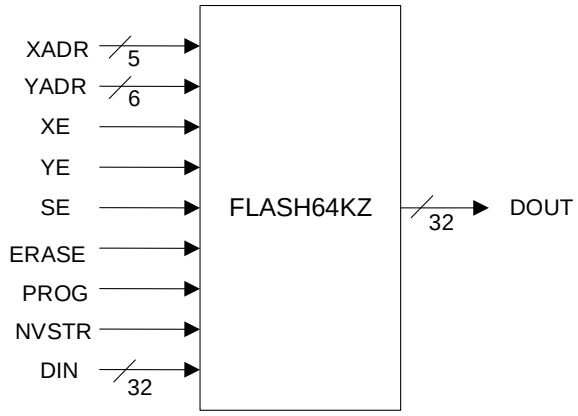
User Flash (FLASH64KZ) has a size of 64 Kbits. The width and depth of registers are fixed. It has non-volatile and power-off saving functions, but without initial value.

FLASH64KZ has the following features.

- 10,000 write cycles
- Size: 32 rows * 64 columns * 32 bits= 64 Kbits
- Data retention for more than 10 years (+85℃)
- Supports page erasure: 2,048 bytes per page
- Quick page erasure/write operation
- Access time: 25ns (Max)
- Program time: 16μs (Max)
- Page erasure time: 120ms (Max)
- Electric current
  - Read Operation: 2.19mA/25ns (V_{CC}) & 0.5mA/25ns (V_{CCX}) (Max)
  - Write operation/erase operation: 12/12 mA (Max)

Port Diagram

Figure 3-6 FLASH64KZ Ports Diagram



Port Description

Table 3-7 FLASH64KZ Port Description

Ports	I/O	Description
DOUT[31:0]	Output	Data output bus
DIN[31:0]	Input	Data input bus
XADR[4:0]	Input	X address bus, used to select one row in a memory cell.
YADR[5:0]	Input	Y address bus, used to select one column within memory block.
XE	Input	X address enable signal, if XE is 0, all row addresses disable.
YE	Input	Y address enable signal, if YE is 0, all of column addresses disable.
SE	Input	Detect amplifier enable signal, active-high.
ERASE	Input	Erase signal, active-high.
PROG	Input	Write signal, active-high.
NVSTR	Input	Flash data storage signal, active-high.

Configuration Mode

GW1NZ series of FPGA products have two types of user flash: general mode and low power mode, and FLASH64KZ is user flash with general mode.

FLASH64KZ is turned on by default, and the device can operate normally after power on, such as erase/read/write operations. And it does not support switching to off.

Operation Mode

Table 3-8 FLASH64KZ Truth Table in User Mode

Mode	XE	YE	SE	PROG	ERASE	NVSTR
Read mode	H	H	H	L	L	L
Write mode	H	H	L	H	L	H

Mode	XE	YE	SE	PROG	ERASE	NVSTR
Page Erasure Mode	H	L	L	L	H	H

Note!

"H" and "L" means high level and low level.

Primitive Instantiation

The primitives can be instantiated directly, or generated by the IP Core Generator. For the details, you can refer to Chapter 4 IP Generation.

Verilog Instantiation:

```
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 Instantiation:

```
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
    );
```

Timing Parameters[1][5][6]

Table 3-9 FLASH64KZ Timing Parameters

User Mode	Parameters	Name	Min.	Max.	Unit
Access time[2]	WC1	T_{acc}[3]	-	25	ns
	TC		-	22	ns
	BC		-	21	ns
	LT		-	21	ns
	WC		-	25	ns
Time from write/erase to data storage setup		T_{nvs}	5	-	μs
Data storage hold time		T_{nvh}	5	-	μs
Data storage hold time (Mass erase)		T_{nvh1}	100	-	μs
Time from data storage to write setup		T_{pgs}	10	-	μs
Program hold time		T_{pgh}	20	-	ns
Program time		T_{prog}	8	16	μs
Write prepare time		T_{wpr}	>0	-	ns
Write hold time		T_{whd}	>0	-	ns
Time from control signal to write/erase setup		T_{cps}	-10	-	ns
Time from SE to read setup		T_{as}	0.1	-	ns
SE pulse high level time		T_{pws}	5	-	ns
Adress/data setup time		T_{ads}	20	-	ns
Adress/data hold time		T_{adh}	20	-	ns
Data hold time		T_{dh}	0.5	-	ns
Readmode address hold time[3]	WC1	T_{ah}	25	-	ns
	TC	-	22	-	ns
	BC	-	21	-	ns

User Mode	Parameters	Name	Min.	Max.	Unit
	LT	-	21	-	ns
	WC	-	25	-	ns
SE pulse low level time		T_{nws}	2	-	ns
Recovery time		T_{rcv}	10	-	μs
Data storage time		T_{hv}[4]	-	6	ms
Erase time		T_{erase}	100	120	ms
Mass erase time		T_{me}	100	120	ms
Wake-up time from power down to standby mode		T_{wk_pd}	7	-	μs
Standby hold time		T_{sbh}	100	-	ns
V_{CC} setup time		T_{ps}	0	-	ns
V_{CCX} hold time		T_{ph}	0	-	ns

Note!

- [1]The parameter values may change.
- [2]The values are simulation data only.
- [3]After XADR, YADR, XE, and YE signals are valid, T_{acc} start time is SE rising edge. DOUT is kept until the next valid read operation.
- [4]T_{hv} time is the cumulative time between write and the next erase, and the same address cannot be written twice before the next erase, so does the same memory cell. This limitation is for safety.
- [5]Both the rising edge time and falling edge time for all waveforms is 1ns.
- [6]X, YADR, XE and YE control signals need to hold at least T_{acc} time, and T_{acc} starts from SE rising edge.

Timing Diagrams

Figure 3-7 FLASH64KZ Read Operation Timing

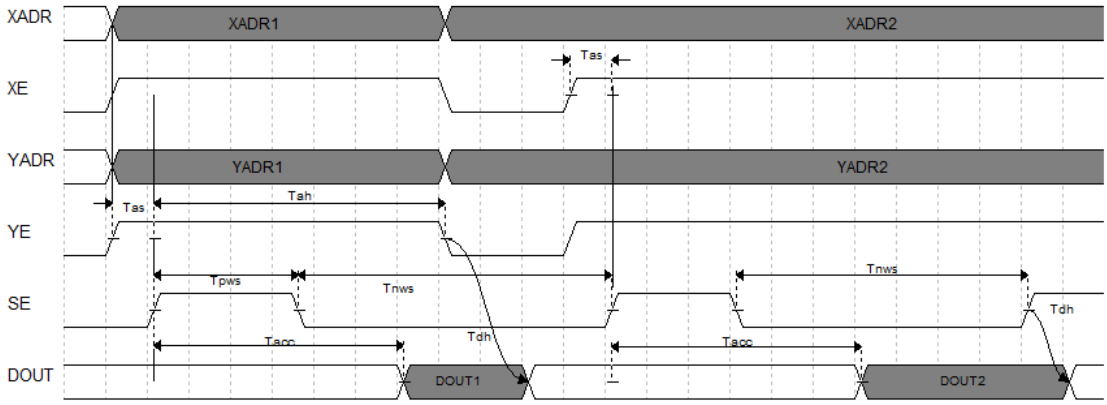


Figure 3-8 FLASH64KZ Write Operation Timing

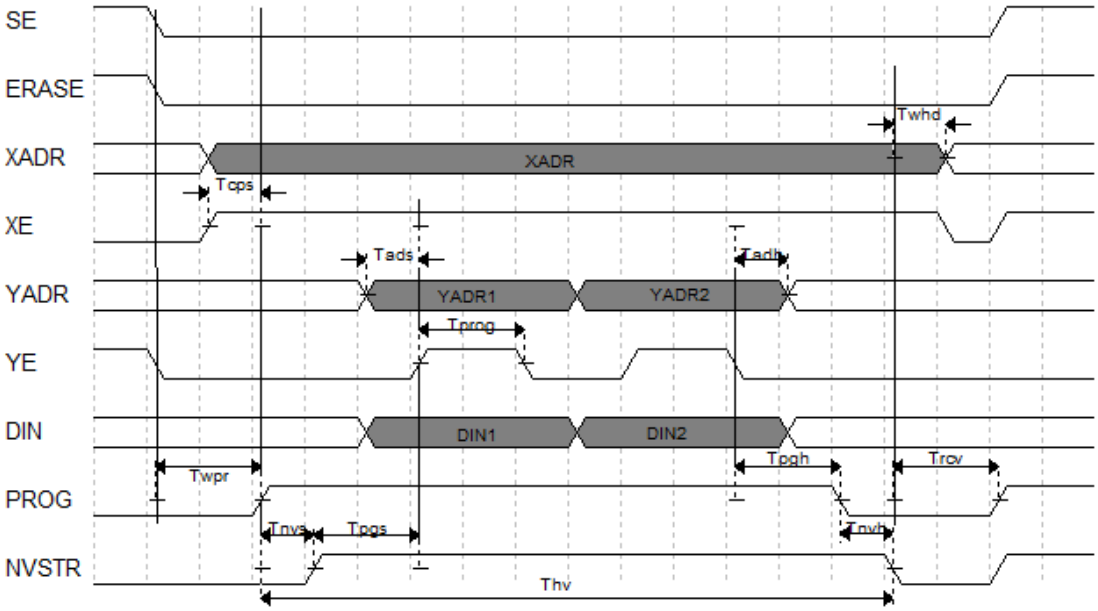
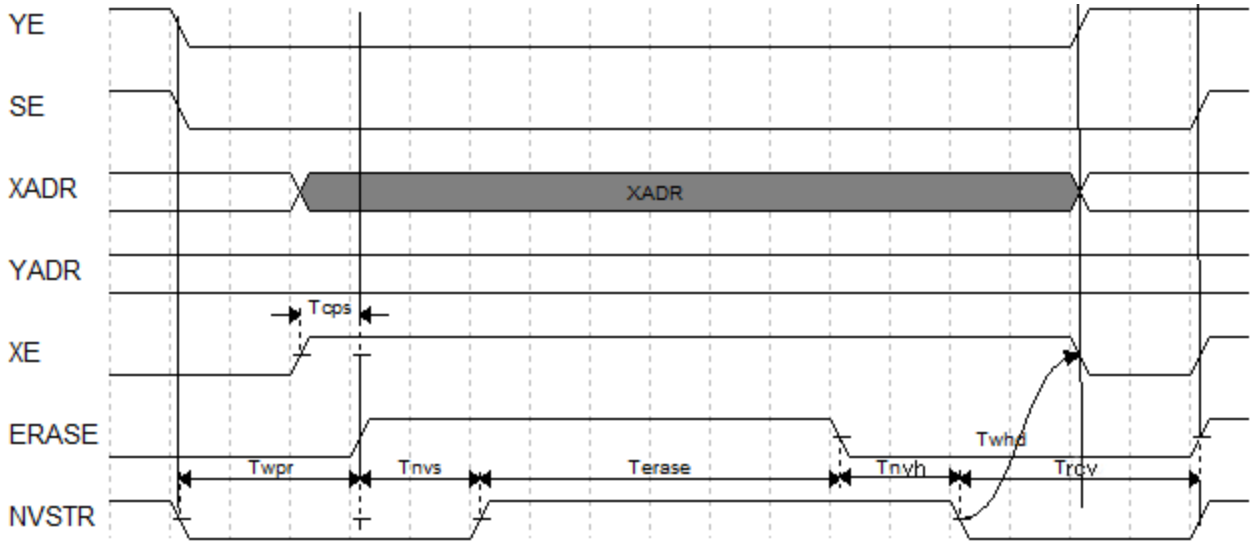


Figure 3-9 FLASH64KZ Erase Operation Timing



3.3 FLASH64K

Primitive Introduction

User Flash (FLASH64K) has a size of 64 Kbits. The width and depth of registers are fixed. It has non-volatile and power-off saving functions, but without initial value. FLASH64K has sleep mode. When SLEEP is high, FLASH enters sleep mode.

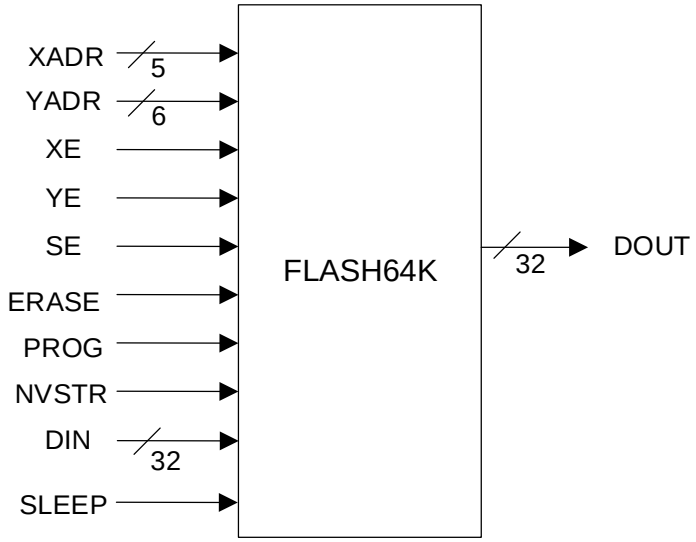
FLASH64K has the following features.

- 10,000 write cycles
- Size: 32 rows * 64 columns * 32 bits = 64 Kbits
- Data retention for more than 10 years (+85℃)

- Supports page erasure: 2,048 bytes per page
- Quick page erasure/Write operation
- Access time: 25ns (Max)
- Program time: 16μs (Max)
- Page erasure time: 120ms (Max)
- Electric current
  - Read Operation: 2.19mA/25ns (V_{CC}) & 0.5mA/25ns (V_{CCX}) (Max)
  - Write operation/erase operation: 12/12 mA(Max)

Port Diagram

Figure 3-10 FLASH64K Diagram



Port Description

Table 3-10 FLASH64K Port Description

Port	I/O	Description
DOUT[31:0]	Output	Data output bus
DIN[31:0]	Input	Data input bus
XADR[4:0]	Input	X address bus, used to select one row in a memory cell.
YADR[5:0]	Input	Y address bus, used to select one column within IOB memory block.
XE	Input	X address enable signal, if XE is 0, all row addresses disable.
YE	Input	Y address enable signal, if YE is 0, all of column addresses disable.
SE	Input	Detect amplifier enable signal, active-high.
ERASE	Input	Erase signal, active-high.
PROG	Input	Write signal, active-high.

Port	I/O	Description
NVSTR	Input	Flash data storage signal, active-high.
SLEEP	Input	Low-power user flash switches to control signal. High-level: on Low-level: off

Configuration Mode

GW1NZ series of FPGA products have two types of user flash: general mode and low power mode, and FLASH64KZ is user flash with lower power mode.

FLASH64K is off by default, which can effectively reduce power consumption. You can switch on/off by SLEEP pin. When switching to on, erase/read/write operations can be performed like FLASH64KZ.

Operation Mode

Table 3-11 FLASH64K Truth Table in User Mode

Mode	XE	YE	SE	PROG	ERASE	NVSTR
Read mode	H	H	H	L	L	L
Write mode	H	H	L	H	L	H
Page EraseMode	H	L	L	L	H	H

Note!

"H" and "L" means high level and low level.

Primitive Instantiation

The primitives can be instantiated directly, or generated by the IP Core Generator. For the details, you can refer to Chapter 4 IP Generation.

Verilog Instantiation:

```
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 Instantiation:

```
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
                        );
```

Timing Parameters & Diagrams

The timing of FLASH64K is the same as that of FLASH64KZ. For the timing parameters & diagrams, you can see 3.2 FLASH64KZ.

3.4 FLASH256K

Primitive Introduction

User Flash (FLASH256K) has a size of 256 Kbits. The width and depth of the register are fixed. It has non-volatile and power-off saving functions, but without the initial value.

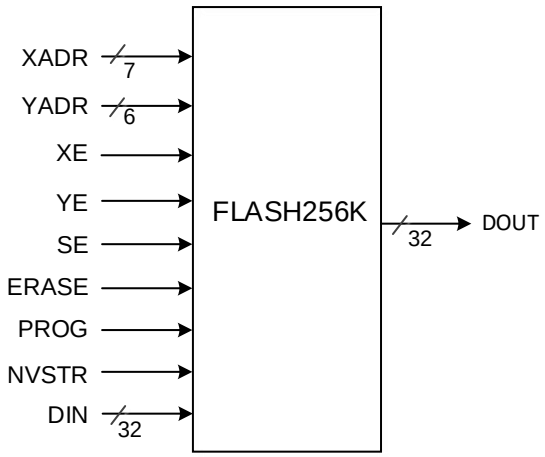
FLASH256K is composed of row memory cells and column memory cells. One row is composed of 64 column memory cells. The column memory size is 32 bits, the row memory size is 64*32=2048 bits. Page erasure is supported: 2048 bytes per page.

FLASH256K has following features.

- 10,000 write cycles
- Data retention for more than ten years (+85℃)
- Data Width: 32
- Size: 128 rows * 64 columns * 32bits = 256 Kbits
- page erasure: 2,048 bytes
- Quick page erasure/ read/write operations
- Access time: 25ns (Max)
- Program time: 16μs (Max)
- Page erasure time: 120ms (Max)
- Electric current
  - Read current/duration:2.19 mA/25 ns (V_{CC}) & 0.5 mA/25 ns (V_{CCX}) (Max)
  - Write operation/erase operations: 12/12 mA(Max)

Port Diagram

Figure 3-11 FLASH256K Ports Diagram



Port Description

Table 3-12 FLASH256K Port Description

Port	I/O	Description
DOUT[31:0]	Output	Data output bus
DIN[31:0]	Input	Data input bus
XADR[6:0]	Input	It is X address bus, which accesses the row address. XADR[n:3] is used to select a page, XADR[2:0] is used to select a row in a page. One page consists of 8 rows, and one row consists of 64 columns.
YADR[5:0]	Input	It is Y address bus, used to select one row in a memory cell, and a row consists of 64 columns.
XE	Input	It is X address enable signal. If XE is 0, all row addresses are not enabled.
YE	Input	It is Y address enable signal. If YE is 0, all column addresses are not enabled.
SE	Input	Detect amplifier enable signal, active-high.
PROG	Input	Write signal, active-high.
ERASE	Input	Erase signal, active-high.
NVSTR	Input	Flash data storage signal, active-high.

Operation Mode

Table 3-13 FLASH256K Truth Table in User Mode

Mode	XE	YE	SE	PROG	ERASE	NVSTR
Read mode	H	H	H	L	L	L
Write mode	H	H	L	H	L	H
Page Erasure Mode	H	L	L	L	H	H

Note!

"H" and "L" means high level and low level.

Primitive Instantiation

The primitives can be instantiated directly, or generated by the IP Core Generator. For the details, you can refer to Chapter 4 IP Generation.

Verilog Instantiation:

```
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 Instantiation:

```
    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
        );
```

Timing Parameters & Diagrams

The timing of FLASH256K is the same as that of FLASH64KZ. For the

timing parameters & diagrams, you can see 3.2 FLASH64KZ.

3.5 FLASH96KA

Primitive Introduction

User Flash (FLASH96KA) has a size of 96 Kbits, which is the same as FLASH256K timing; it supports sleep mode, and FLASH96KA enters to this mode when signal SLEEP is high. The width and depth of the registers are fixed and cannot be configured. It has non-volatile and power-off save functions, but does not support initial value function.

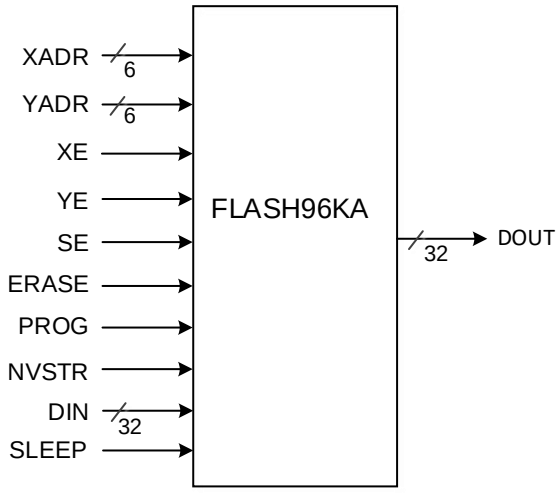
FLASH96KA is composed of row memory cells and column memory cells. One row is composed of 64 column memory cells. The column memory size is 32 bits, the row memory size is 64*32 bits=2048 bits. Page erasure is supported: 2048 bytes per page.

FLASH256KA has following features.

- 10,000 write cycles
- Data retention for more than ten years (+85℃)
- Data Width: 32
- Size: 48 rows * 64 columns * 32 bits = 96 Kbits
- Page erasure: 2,048 bytes
- Quick page erasure/ read/write operations
- Access time: 25ns (Max)
- Program time: 16μs (Max)
- Page erasure time: 120ms (Max)
- Electric current
  - Read current/duration:2.19 mA/25 ns (V_{CC}) & 0.5 mA/25 ns (V_{CCX}) (Max)
  - Write operation/erase operations: 12/12 mA(Max)

Port Diagram

Figure 3-12 FLASH96KA Ports Diagram



Port Description

Table 3-14 FLASH96KA Port Description

Port	I/O	Description
DOUT[31:0]	Output	Data output bus
DIN[31:0]	Input	Data input bus
XADR[6:0]	Input	It is X address bus, which accesses the row address. XADR[n:3] is used to select a page, XADR[2:0] is used to select a row in a page. One page consists of 8 rows, and one row consists of 64 columns.
YADR[5:0]	Input	It is Y address bus, used to select one row in a memory cell, and a row consists of 64 columns.
XE	Input	It is X address enable signal. If XE is 0, all row addresses are not enabled.
YE	Input	It is Y address enable signal. If YE is 0, all column addresses are not enabled.
SE	Input	Detect amplifier enable signal, active-high.
PROG	Input	Write signal, active-high.
ERASE	Input	Erase signal, active-high.
NVSTR	Input	Flash data storage signal, active-high.
SLEEP	Input	When SLEEP is high, Flash stops working.

Operation Mode

Table 3-15 FLASH96KA Truth Table in User Mode

Mode	XE	YE	SE	PROG	ERASE	NVSTR
Read mode	H	H	H	L	L	L
Write mode	H	H	L	H	L	H
Page Erasure	H	L	L	L	H	H

Mode	XE	YE	SE	PROG	ERASE	NVSTR
Mode						

Note!

"H" and "L" means high level and low level.

Primitive Instantiation

The primitives can be instantiated directly, or generated by the IP Core Generator. For the details, you can refer to Chapter 4 IP Generation.

Verilog Instantiation:

```
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 Instantiation：

```
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
    );
```

Timing Parameters & Diagrams

The timing of FLASH96KA is the same as that of FLASH64KZ. For the timing parameters & diagrams, you can see 3.2 FLASH64KZ.

3.6 FLASH608K

Primitive Introduction

User Flash (FLASH608K) has a size of 608 Kbits. The width and depth of the register are fixed and cannot be configured. It has non-volatile and power-off saving functions, but without the initial value.

FLASH608K is composed of row memory and column memory cells. One row is composed of 64 column memory cells. The column memory size is 32 bits, and the row memory size is 64*32 bits=2048 bits. Page erasure is supported: 2048 bytes per page.

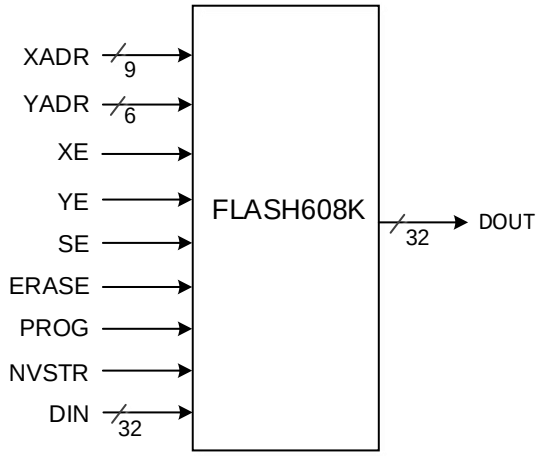
FLASH608K has following features.

- 10,000 write cycles
- Data retention for more than ten years (+85℃)
- Data Width: 32
- Size: 304 rows * 64 columns * 32 bits = 608 Kbits
- Page erase size: 2,048 bytes
- Quick page erasure/ read/write operations
- Access time: 25ns (Max)
- Program time: 16μs (Max)

- Page erasure time: 120ms (Max)
- Electric current
  - Read current/duration:2.19 mA/25 ns (V_{CC}) & 0.5 mA/25 ns (V_{CCX}) (Max)
  - Write operation/erase operations: 12/12 mA(Max)

Port Diagram

Figure 3-13 FLASH608K Ports Diagram



Port Description

Table 3-16 FLASH608K Port Description

Port	I/O	Description
DOUT[31:0]	Output	Data output bus
DIN[31:0]	Input	Data input bus
XADR[8:0]	Input	It is X address bus, which accesses the row address. XADR[n:3] is used to select a page, XADR[2:0] is used to select a row in a page. One page consists of 8 rows, and one row consists of 64 columns.
YADR[5:0]	Input	It is Y address bus, used to select one row in a memory cell, and a row consists of 64 columns.
XE	Input	It is X address enable signal. If XE is 0, all row addresses are not enabled.
YE	Input	It is Y address enable signal. If YE is 0, all column addresses are not enabled.
SE	Input	Detect amplifier enable signal, active-high.
PROG	Input	Write signal, active-high.
ERASE	Input	Erase signal, active-high.
NVSTR	Input	Flash data storage signal, active-high.

Operation Mode

Table 3-17 FLASH608K Truth Table in User Mode

Mode	XE	YE	SE	PROG	ERASE	NVSTR
Read mode	H	H	H	L	L	L
Write mode	H	H	L	H	L	H
Page Erasure Mode	H	L	L	L	H	H

Note!

"H" and "L" means high level and low level.

Primitive Instantiation

The primitives can be instantiated directly, or generated by the IP Core Generator. For the details, you can refer to Chapter 4 IP Generation.

Verilog Instantiation:

```
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 Instantiation:

```
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
        );
```

Timing Parameters & Diagrams

The timing of FLASH608K is the same as that of FLASH64KZ. For the timing parameters & diagrams, you can see 3.2 FLASH64KZ.

4 IP Generation

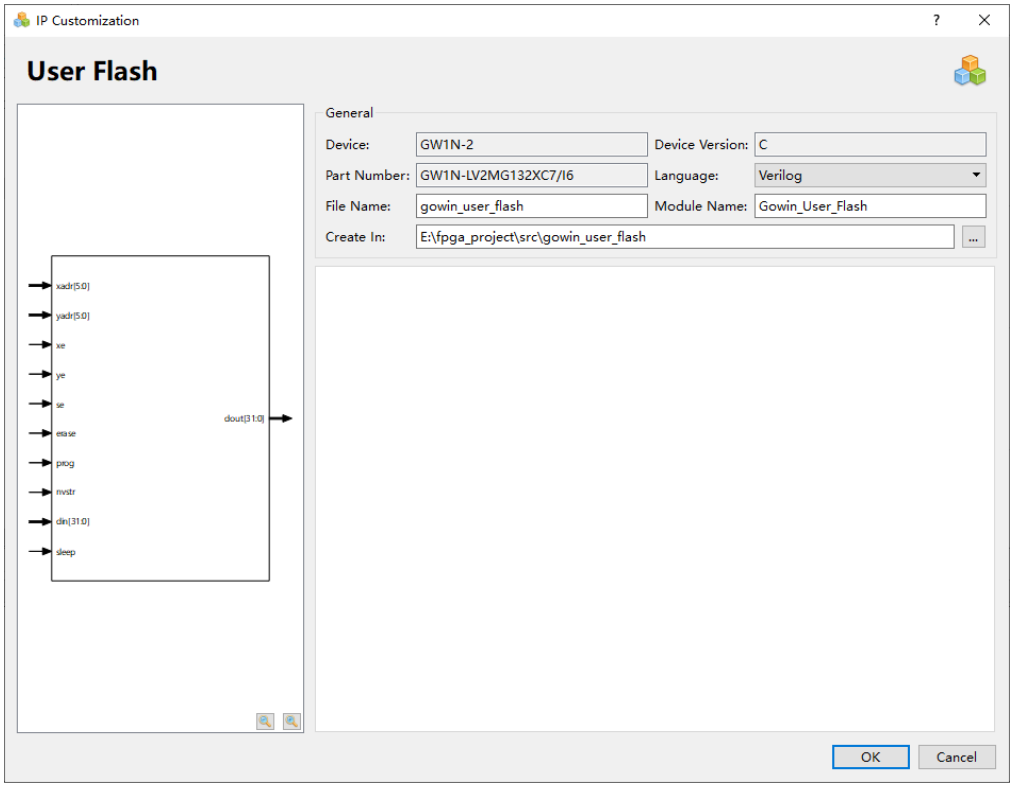
IP Core Generator can generate hard core User Flash and soft core Gowin Flash Controller IP. For the details of Gowin Flash Controller IP, refer to *IPUG901, Gowin Flash Controller IP User Guide*. Hard core User Flash can be generated by "User Flash" on GUI.

Click "User Flash" on the IP Core Generator page. A brief introduction to the User Flash will be displayed.

IP Configuration

Double-clicking "User Flash", and the "IP Customization" window will pop up. This displays the "General" configuration, and port diagram as shown in Figure 4-1.

Figure 4-1 IP Customization of User Flash



1. File: The File displays the basic information related to the DP.
   - Device: Displays information about the configured device.
   - Device Version: Displays information about the configured device version.
   - Part Number: Displays the configured Part Number.
   - Language: Hardware description language used to generate the IP design files. Click the drop-down list to select the language, including Verilog and VHDL.
   - Module Name: The module name of the generated IP design files. Enter the module name in the text box. Module name cannot be the same as the primitive name. If it is the same, an error will be reported.
   - File Name: The name of the generated IP design files. Enter the file name in the text box.
   - Create In: The path in which the generated IP files will be stored. Enter the target path in the box or select the target path by clicking the option.
2. Ports Diagram: The ports diagram displays the current IP Core configuration, and User Flash input bit-width updates in real time based on the target device, as shown in Figure 4-1.

IP Generation Files

After configuration, it will generate three files that are named after the "File Name".

- "gowin_user_flash.v" file is a complete Verilog module to generate instance User Flash, and it is generated according to the IP configuration;
- "gowin_user_flash_tmp.v" is the instance template file;
- "gowin_user_flash.ipc" file is IP configuration file. You can load the file to configure the IP.

Note!

If VHDL is selected as the hardware description language, the first two files will be named with .vhd suffix.


GOWIN
PROGRAMMING FOR THE FUTURE