

iCE65 as NAND Flash Interface

Overview

NAND Flash technology provides a cost-effective solution for applications requiring high-density, solid-state storage. This design example illustrates the implementation of a NAND Flash interface using iCE65 FPGAs. As a NAND Flash interface, the iCE65 FPGA generates all the required interfacing signals and implements the functions that are required for such a system.

Description

NAND Flash devices use a highly multiplexed 8- or 16-bit bus (I/O[7:0] or I/O[15:0]) to transfer data, addresses, and instructions. The five command pins (CLE, ALE, CE#, RE#, WE#) implement the NAND Flash command bus interface protocol. Two additional pins control hardware write protection (WP#) and monitor device status (R/B#). This hardware interface creates a low-pin-count device with a standard pinout that is the same from one density to another, supporting future upgrades to higher densities without board redesign. Table 1 lists the control signals of NAND Flash memory.

Command Write

Commands are written to the command register on the rising edge of WE# when:

- CE# and ALE are LOW, and
- CLE is HIGH, and
- The device is not busy

The exceptions to this are the READ STATUS and RESET commands. Commands are transferred to the command register on the rising edge of WE#.

Address Write

Addresses are written to the address register on the rising edge of WE# when:

- CE# and CLE are LOW, and
- ALE is HIGH, and
- The device is not busy

Addresses are input on I/O[7:0] only.

Data Write

Data is written to the data register on the rising edge of WE# when:

- CE#, CLE, and ALE are LOW, and
- The device is not busy

Date Read

After a READ command is sent to the memory device, data is transferred from the memory array to the data register in tR. Typically tR is 25µs. When data is available in the data register, it is clocked out of the part by RE# going LOW. The READ STATUS (70h) command or the R/B# signal can be used to determine when the device is ready.

Ready/Busy

The R/B# output provides a hardware method of indicating the completion of a PROGRAM/ERASE/READ operation. The signal is typically HIGH, and transitions to LOW after the appropriate command is written to the device.

Refer to data sheets of NAND Flash Memory Devices for more details on various commands, control signals and timing diagram.

Symbol	Signal	Description
ALE	Address Latch Enable	When ALE is High, addresses are latched into the NAND Flash address register on the rising edge of the WE# signal.
CE#	Chip Enable	If CE is not asserted , the NAND Flash device remains in standby mode and does not respond to any control signals.
CLE	Command Latch Enable	When CLE is high, commands are latched into the NAND Flash command register on the rising edge of the WE# signal.
RE#	Read Enable	Read data or device ID. Read enable is low for read data when r_w signal is high.
WE#	Write Enable	Write Enable is low for write data or command as well as when command latch enable is high.
WP#	Write Protect	When WPN is low, write protect enabled and when WPN is high then write protect is disabled.
SEN#	Spare Area Enable	When SEN is low, Spare area is enabled for read and write.
I/O	Input/output	Data inputs or outputs. The bidirectional I/O pins transfer address, data and instruction information. Data is output only during READ operations, at other times the I/O pins are inputs.
R/B#	Read/Busy	Obtain busy/ready status. Tristated when status not requested or chip not selected.

Table 1: Control Signal Description

Implementation

NAND Flash Interface connection diagram is shown in Figure 1. The following summarizes the operation of the NAND Flash Interface:

- A low on system reset signal rst_n brings the NAND Flash device to stand-by mode
- A high on chip select pin cs_n, de-selects NAND Flash by providing a High on NAND Flash Chip enable, nand_cen.
- A low on chip select signal cs_n generates appropriate control signal based on fn_cntrl port values and r_w signal.
- when "0000" and r_w = x then Chip Enable, nand_cen = 1
- when "0001" and r_w = 0, then Command Latch Enable, nand_cle = 1 and Write Enable, nand_wen = 0
- when "0010" and r_w = 1, then Read Enable, nand_ren = 0
- when "0011" and r_w = 0, then Write Enable, nand_wen = 0
- when "0100" and falling edge of r_w, then Address Latch Enable, nand_ale = 1
- when "0101" and falling edge of r_w, then Address Latch Enable, nand_ale = 0
- when "0110" and falling edge of r_w, then Spare Area Enable, nand_sen = 0
- when "0111" and falling edge of r_w, then Spare Area Enable, nand_sen = 1
- when "1000" and falling edge of r_w, then Write Protect Enable, nand_wpn = 0
- when "1001" and falling edge of r_w, then Write Protect Enable, nand_wpn = 1
- when "1010" and falling edge of r_w, then Chip Enable, nand_cen = 0
- when "1011" and falling edge of r_w, then Chip Enable, nand_cen = 1
- when "1100" and r_w = 1, then Interface reads nand_rdbsy pin of NAND Flash and updates it to busy_status pin of Host/CPU.

Table 2 below lists interface details of this Design Example.

Table 3 below shows the post P&R resource utilization summary of this design when implemented on an iCE65 FPGA.

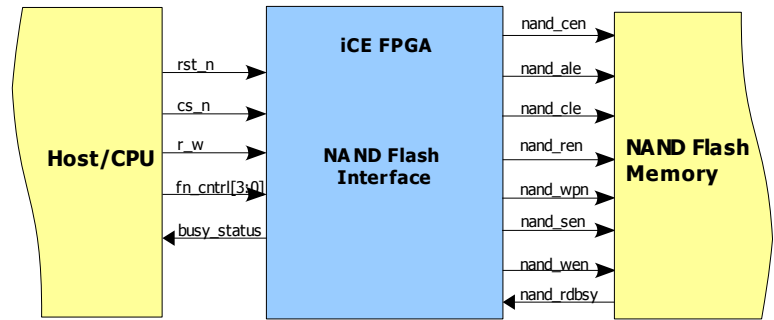


Fig 1: NAND Flash Interface

Pin	Direction	Description
rst_n	Input	Reset signal aborts the current transaction and resets IF signals
cs_n	Input	Chip select signal generated by the host
r_w	Input	Active high read,active low write signal generated by CPU/Host controller
fn_cntrl[3:0]	Input	Control signal generated by the host for the NAND flash functions
nand_rdbsy	Input	NAND Flash ready/busy signal
nand_cen	Output	Chip select
nand_ren	Output	Read enable, it controls the data and status read on i/o line
nand_wen	Output	Write enable, it controls the data and command write on i/o line
nand_cle	Output	Command latch enable, it controls the writing to the command register
nand_ale	Output	Address Latch enable, it controls the address write
nand_sen	Output	Active low spare area enable, access to the 16 byte of spare area on each page
nand_wpn	Output	Active low Write protect enable
busy_status	Output	Active low NAND flash busy status to host

Table 2: Pin Description

Device	Logic Cells	IO Cells
iCE65L04-UCB284	20	16

Table 3: Resource Utilization

Conclusion

This design example demonstrates the implementation of a NAND Flash interface using iCE FPGAs. iCE FPGA's very low power capabilities makes iCE FPGAs an obvious choice for implementing a NAND Flash interface for battery operated compact and handheld devices like PDAs, cellular phones etc..

Disclaimer

SiliconBlue is providing this document, design example, or information "as is." SiliconBlue does not ensure that this implementation or information is void of any claims of infringement. As a reader / implementer, you are responsible for obtaining any rights you may require for your implementation. SiliconBlue also does not warranty the fitness of this design example to be readily implemented in any specific context. It is your sole responsibility to validate this design example and its correctness while implementing it for your requirement.