CF+ Interface Using Altera MAX Series

2014.09.22

AN-492 Subscribe Send Feedback

You can use Altera® MAX® II, MAX V, and MAX 10 devices to implement a CompactFlash+ (CF+) interface. Their low-cost, low-power, and easy power-on features make them the ideal programmable logic devices for memory device interfacing applications.

CompactFlash cards store and transport several forms of digital information (data, audio, pictures) and software between a wide span of digital systems. The CompactFlash association introduced the CF+ concept to enhance the operation of CompactFlash cards with I/O devices and magnetic disk data storage apart from flash memory. The CF+ card is a small form factor card that includes compact flash storage cards, magnetic disk cards, and various I/O cards that are available in the market, such as serial cards, ethernet cards, and wireless cards. The CF+ card includes an embedded controller that manages data storage, retrieval and error correction, power management, and clock control. CF+ cards can be used with passive adapters in PC-Card type-II or type-III sockets.

Nowadays, many consumer products such as cameras, PDAs, printers, and laptops have a socket that accepts CompactFlash and CF+ memory cards. In addition to storage devices, this socket can also be used to interface I/O devices that use the CF+ interface.

Related Information

- **Design Example for MAX II**
  Provides the MAX II design files for this application note (AN 492)
- **Design Example for MAX 10**
  Provides the MAX 10 design files for this application note (AN 492)
- **Power Management in Portable Systems Using Altera Devices**
  Provides more information about power management in portable systems using Altera devices
- **MAX II Device Design Guidelines**
  Provides more information about MAX II device design guidelines

Using the CF+ Interface with Altera Devices

The CF+ card interface is enabled by the host by asserting the `H_ENABLE` signal. When the CompactFlash card is inserted in the socket, the two pins (`CD_1 [1:0]`) go low, indicating to the interface that the card has been inserted properly. In response to this action, an interrupt signal `H_INT` is generated by the interface, depending on the status of CD_1 pins and the chip enable signal (`H_ENABLE`).

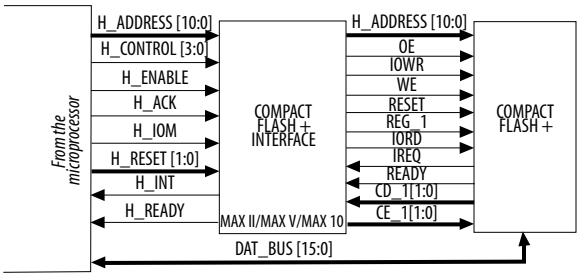
The `H_READY` signal is also asserted whenever the required conditions are met. This signal indicates to the processor that the interface is ready to accept the data from the processor. The 16-bit data bus to the CF+ card is connected directly to the host. When the host receives an interrupt signal, it responds to it by generating an acknowledgment signal, `H_ACK`, for the interface to indicate that it has received the interrupt

and is ready to perform further functions. This signal acts as an impetus; all operations of the interface, host, or the processor and CompactFlash card are synchronized to this signal. The interface also checks for H_RESET signal; this signal is generated by the host to indicate that all the initial conditions must be reset. The interface in turn generates the RESET signal to the CompactFlash card indicating to it to reset all its control signals to its default condition. The H_RESET signal can either be hardware or software generated. The software reset is indicated by the MSB of the Configuration Option Register within the CF+ card. The host generates a 4-bit control signal H_CONTROL to indicate the desired function of the CF+ card to the CF + interface. The interface decodes the H_CONTROL signal and issues various control signals to read and write data, and configuration information. Every card operation is synchronized to the H_ACK signal. At the positive edge of the H_ACK, the supported Altera device checks for the reset signal, and correspondingly issues the HOST_ADDRESS, chip enable (CE_1), output enable (OE), write enable (WE), REG_1, and RESET signals. Each of these signals have a predefined value for all the operations mentioned above. These are standard protocols, as defined by the CompactFlash association.

The H_IOM signal is held low in common memory mode and high in I/O mode. The common memory mode allows writing and reading of both 8-bit and 16-bit data. Also, the Configuration Registers in the CF + card configuration option register, Card Status Register, and Pin Replacement Register are read from and written into. A 4-bit wide H_CONTROL [3:0] signal issued by the host differentiates between all these operations. The CF+ interface decodes H_CONTROL and issues the control signals to the CF+ card according to the CF+ specifications. Data is made available on the 16-bit data bus after the control signals are issued. In the I/O mode, the software reset (generated by making the MSB of the Configuration Option Register in the CF+ card high) is checked. Byte and word access operations are executed by the interface in a manner similar to those in the memory mode detailed above.

Figure 1: The Different Interfacing Signals of the CF+ Interface and the CF+ Device

This figure shows the basic block diagram for implementing the CF+ interface.



Signals

Table 1: CF+ Interface Signals

This table lists the CF+ card interfacing signals.

Signal	Direction	Description
HOST_ADDRESS [10:0]	Output	These address lines select the following: the I/O port address registers, the memory mapped port address registers, its configuration control and status registers.
CE_1 [1:0]	Output	This is a 2-bit active-low card select signal.

Signal	Direction	Description
IORD	Output	This is an I/O read strobe generated by the host interface to gate the I/O data on the bus from the CF+ card.
IOWR	Output	This is an I/O write pulse strobe used to clock the I/O data on the card data bus on the CF+ card.
OE	Output	Active-low output enable strobe.
READY	Input	In memory mode, this signal is kept high when the CF+ card is ready to accept a new data transfer operation and low when the card is busy.
IREQ	Input	In the I/O mode operation, this signal is used as an interrupt request. It is strobed low.
REG_1	Output	This signal is used to distinguish between common memory and attribute memory accesses. High for common memory and low for attribute memory. In I/O mode, this signal should be active-low when the I/O address is on the bus.
WE	Output	Active-low signal for writing into the card configuration registers.
RESET	Output	This signal resets or initializes all registers in the CF+ card.
CD_1 [1:0]	Input	This is a 2-bit active-low card detect signal.

Table 2: Host Interface Signals

This table lists the signals that form the host interface.

Signal	Direction	Description
H_INT	Output	Active-low interrupt signal from interface to host indicating insertion of card.
H_READY	Output	Ready signal from interface to host indicating CF+ is ready to accept new data.
H_ENABLE	Input	Chip enable
H_ACK	Input	Acknowledgment to the interrupt request made by the interface.
H_CONTROL [3:0]	Input	A 4-bit signal selecting between I/O and memory READ/WRITE operations.
H_RESET [1:0]	Input	A 2-bit signal for hardware and software reset.
H_IOM	Input	Differentiates memory mode and I/O mode.

Implementation

These designs may be implemented using MAX II, MAX V, and MAX 10 devices. The provided design source codes target the MAX II (EPM240) and MAX 10 (10M08) respectively. These design source codes are compiled and can be programmed directly to the MAX devices.

For the MAX II design example, map the host and CF+ interfacing ports to suitable GPIOs. This design utilizes about 54% of the total LEs in an EPM240 device and uses 45 I/O pins.

The MAX II design example uses a CF+ device, which functions in two modes: PC Card ATA using I/O mode and PC Card ATA using memory mode. The third optional mode, True IDE mode, is not considered. The MAX II device operates as the host controller and acts as a bridge between the host and the CF+ card.

Source Code

These design examples are implemented in Verilog.

Acknowledgments

Design example adapted for Altera MAX 10 FPGAs by:

Orchid Technologies Engineering and Consulting, Inc.

Maynard, Massachusetts 01754

TEL: 978-461-2000

WEB: **www.orchid-tech.com**

EMAIL: **info@orchid-tech.com**

Document Revision History

Table 3: Document Revision History

Date	Version	Changes
September 2014	2014.09.22	Added MAX 10 information.
December 2007, V1.0	1.0	Initial release.