Introduction

A Digital Picture Frame Interface, DiPFI, has been designed that will act as a buffer between a personal computer storing digital photographs and a VGA compatible display. JPEG images are decoded to raw .ppm files and then transferred to a Rabbit 3010 Core Module's[1] RJ-45 port. Once the data is received, the 21-bit memory address is sequentially latched into two Atmel 22V10 PLDs[2] operating at 3.3V. The PLDs' sole purpose is in Rabbit pin conservation, as only sixteen total pins are needed for both the 21-bit address bus and the 16-bit data bus. These PLDs latch the address so that the same Rabbit pins can be used to bus the data values to the Epson graphics controller[3]. The Epson has been equipped with a 4 Mb EDO DRAM memory chip[4] that it uses to make the image data available for its onboard digital-to-analog converter. The Epson also controls the RGB outputs being sent to any VGA compatible display device. The system operates using a standard unregulated 9 VDC wall wart. The box also communicates with a standard Sony IR remote control, as well as onboard pushbuttons and status LEDs that enables the user to communicate with the device.

The device is also driven using a 25.175 MHz external clock[5].

Theory of Operation

Rabbit 3010 Core Module

The Rabbit 3010 best suites the need of this project because of its Ethernet capabilities as well as its large number of I/O pins. The Rabbit has a maximum internal clock frequency of 29.4 MHz generated by a 14.7456 MHz crystal. The Rabbit boasts an internal clock doubler that allows it to achieve the maximum clock frequency of 29.4 MHz. Therefore, the graphics controller is clocked externally using a 25.175 MHz clock, while running the Rabbit at its maximum clock rate of 29.4 MHz. This clock frequency gives a ~183 kbps baud rate, which is sufficient for the needs of the project. The Rabbit takes a regulated 3.3 VDC power supply. To achieve this from a 9 V unregulated power source, a 500 mA Low-Dropout Voltage Regulator[6] is used to drop the 9 V to a regulated 3.3 V. The RCM 3010 also contains 128 kb of SRAM which is adequate for software storage and any picture buffering as needed.

Another consideration on the Rabbit is how to interpret the image data packets arriving on the Ethernet port. The PC connects to the Rabbit controller using TCP/IP and begins sending data pixel by pixel. Each packet contains address and data information. When the packet arrives at the Rabbit port, the packet is read and the data dropped into a buffer. Another routine works on forwarding any information contained to the graphics controller. Because of a lack of input pins, the address is loaded into two Atmel 22V10 PLDs. These were chosen because of their operation at 3.3V, as well as their DIP package. The PLDs take two mode pins for a total of four states. One mode latches the lower 11 of 21 address bits into one PLD. The second mode latches the rest of the address bits into PLD number two. The last mode enables the data bus and begins communications with the Epson Controller. Since there are eight control signals on the Epson, a third PLD is used to multiplex the signals as necessary.

Epson Graphics Controller

With 130+ pins, the Epson graphics controller poses a major interfacing concern. It has 21 address pins, 16 data pins, 9 memory address pins, and 16 memory data pins. It also contains 5 pins that interface to a standard VGA connector for picture display. As aforementioned, this interfacing is handled with a minimum number of PLDs and control signals. The Epson data sheets do not stress any timing constraints, so this will be a sufficient method for transmitting the image data.

Another concern on the Epson is its onboard Digital to Analog converter, which requires a separate 3.3VDC power and ground from that of the digital circuit. We accommodated this using a separate low-dropout regulator and ground. The DAC also requires a 4.6 mA current reference, IREF, which is provided by a 2n2222 NPN transistor.

Sharp IR Receiver[7] and Reynolds Electronics Decoder[8]

The Sharp IR receiver operates at 40 kHz, which is the modulation frequency of the standard Sony remote control protocol. This way, any Universal Remote can be used to control the device. An issue arrives with unwanted interference from other Sony-type remotes, in that the device is not intelligent enough to differentiate between two remotes transmitting with the same protocol. A solution has not been conceived, and it is not a priority at this time.

A Reynolds Electronics IR decoder chip was found that inputs the received IR signal and toggles a corresponding output pin. This works for buttons 0-9 and the channel and volume buttons. The channel and volume buttons are utilized in this implementation. The Rabbit monitors its input pins and watches for a toggled bit at the decoder output. It then executes the appropriate command. The Rabbit can sink up to 6 mA of current, so the output current of the decoder chip was monitored early on to ensure compatibility with the Rabbit.

An IR decoder that operates at 3.3 V could not be found. As such, a final low-dropout regulator is used to achieve a digital 5 VDC power supply. Also, the Rabbit can handle a maximum 5.5 VDC at its input pins, thus enabling the direct connection of the decoder to the input pins of the Rabbit without using level translation.

LEDs and Pushbuttons

There is little concern as far as power/current limits here. Two different colored LEDs, green and yellow, are used to indicate status signals, blinking when an image is being transferred. The source/sink limit for a Rabbit at 29.4 MHz is 6 mA, so a current limiting resistor has been used. Two Rabbit input pins are used to toggle the LEDs state. The LEDs can be shut off using the Hi-Z state of both input pins, but there was no need for this in the current implementation.

Clock

A 25.175 MHz oscillator is used as a separate means to externally clock the Epson Graphics Controller.

Reference to Devices Selected

[1] Rabbit 3010 Core Module

http://shay.ecn.purdue.edu/~477grp12/datasheets/rabbit3000_core_manual.pdf

[2] Atmel 22V10 PLD

http://shay.ecn.purdue.edu/~477grp12/datasheets/atmel_PLD.pdf

[3] Epson Graphics Controller

http://shay.ecn.purdue.edu/~477grp12/datasheets/epson_manual.pdf

[4] 4 Mb EDO DRAM

http://shay.ecn.purdue.edu/~477grp12/datasheets/dram.pdf

[5] Epson 25.175 MHz Clock

http://shay.ecn.purdue.edu/~477grp12/datasheets/clock.pdf

[6] Texas Instruments Low-Dropout Voltage Regulators

http://shay.ecn.purdue.edu/~477grp12/datasheets/ldo.pdf

[7] Sharp IR Detector

http://shay.ecn.purdue.edu/~477grp12/datasheets/sharp_ir_detector_data.pdf

[8] Reynolds Electronics IR Decoder

http://shay.ecn.purdue.edu/~477grp12/datasheets/rentron_ir_decoder.pdf