XESS Corporation

XSA-200 Board V1.3 User Manual

How to install, test, and use your new XSA-200 Board

RELEASE DATE: 6/23/2005

Table of Contents

1 Preliminaries

Getting Help!

Here are some places to get help if you encounter problems:

- If you can't get the XSA-200 Board hardware to work, send an e-mail message describing your problem to help@xess.com or submit a problem report at http://www.xess.com/help.html. Our web site also has
  - answers to frequently-asked-questions,
  - example designs, application notes and tutorials for the XS Boards,
  - a place to sign-up for our email forum where you can post questions to other XS Board users.
- If you can't get your XILINX WebPACK software tools installed properly, send an e-mail message describing your problem to hotline@XILINX.com or check their web site at http://www.xilinx.com/support/support.htm.
- If you need help using the WebPACK software to create designs for your XSA-200 Board, then check out this tutorial.

Take notice!!

- The XSA-200 Board requires an external power supply to operate! It does not draw power through the downloading cable from the PC parallel port.
- If you are connecting a 9V DC power supply to your XSA-200 Board, please make sure the center terminal of the plug is positive and the outer sleeve is negative.
- Do not power your XSA-200 Board with a battery! This will not provide enough current to insure reliable operation of the XSA-200 Board.

Packing List

Here is what you should have received in your package:

- an XSA-200 Board;
- a 6' cable with a 25-pin male connector on each end;
- an XSTOOLS CDROM with software utilities and documentation for using the XSA-200 Board.

2 Installation

Installing the XSTOOLS Utilities and Documentation

XILINX currently provides the WebPACK tools for programming their CPLDs and Spartan-series FPGAs. The XESS CDROM contains a version of WebPACK that will generate bitstream configuration files compatible with your XSA-200 Board. You can also download the most current version of the WebPACK tools from the XILINX website.

In addition, XESS Corp. provides the XSTOOLS utilities for interfacing a PC to your XSA-200 Board. These utilities should be installed automatically when you insert the XSTOOLS CDROM into your CDROM drive. If not, then manually run the SETUP.EXE installation program on the CDROM.

Applying Power to Your XSA-200 Board

You can use your XSA-200 Board in three ways, distinguished by the method you use to apply power to the board. **Only use one of these methods to power your XSA-200 Board!** Supplying power from multiple sources can damage the board and/or power supplies.

Using a 9V DC wall-mount power supply

You can use your XSA-200 Board all by itself to experiment with logic designs. Just place the XSA-200 Board on a non-conducting surface as shown in Figure 1. Then apply power to the XSA-200 Board from a 9V DC wall-mount power supply with a 2.1 mm female, center-positive plug. (See Figure 2 for the location of the 9V DC power jack on your XSA-200 Board.) The on-board voltage regulation circuitry will create the voltages required by the rest of the XSA-200 Board circuitry. **Be careful!! The voltage regulators on the XSA-200 Board can become hot.** Attach a heat sink to them if necessary.

Powering Through the PS/2 Connector

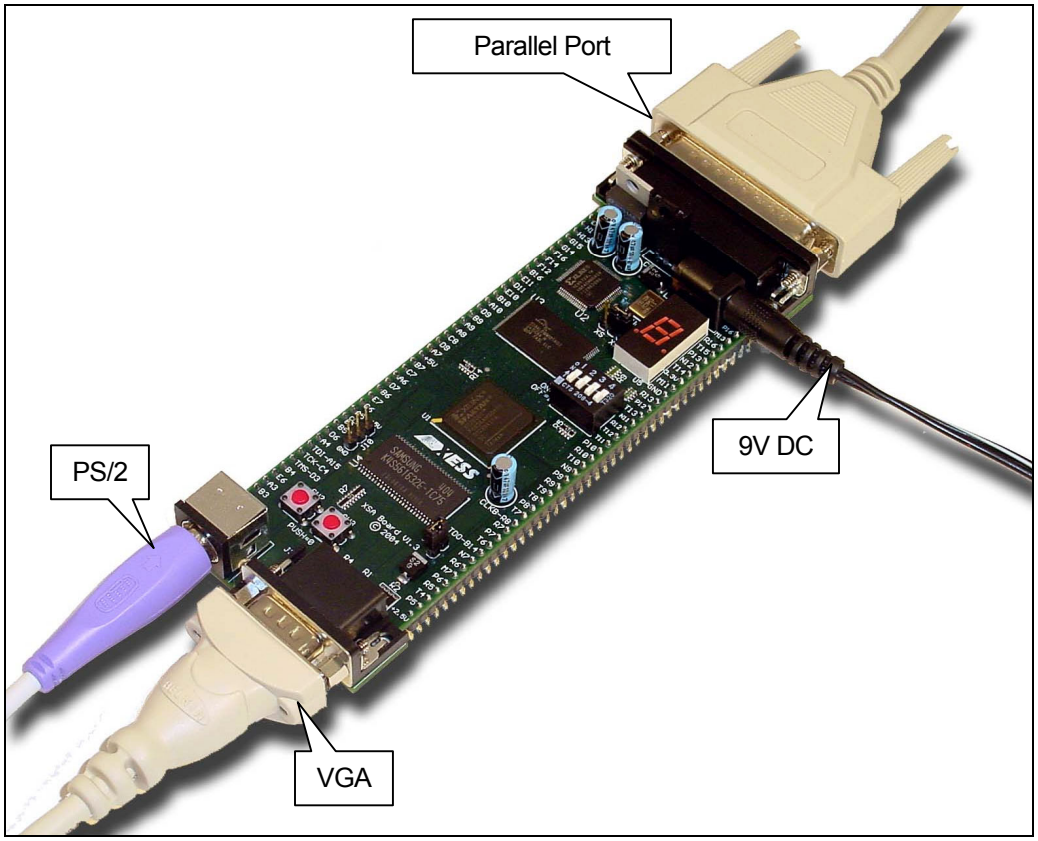
You can use your XSA-200 Board with a laptop PC by connecting a PS/2 male-to-male cable between the PS/2 ports of the laptop and the board. The on-board voltage regulation circuitry will create the voltages required by the rest of the XSA-200 Board circuitry. **Many PS/2 ports cannot supply more than 0.5A so large, high-frequency FPGA designs may not work when using this power source!**

Solderless Protoboard Installation

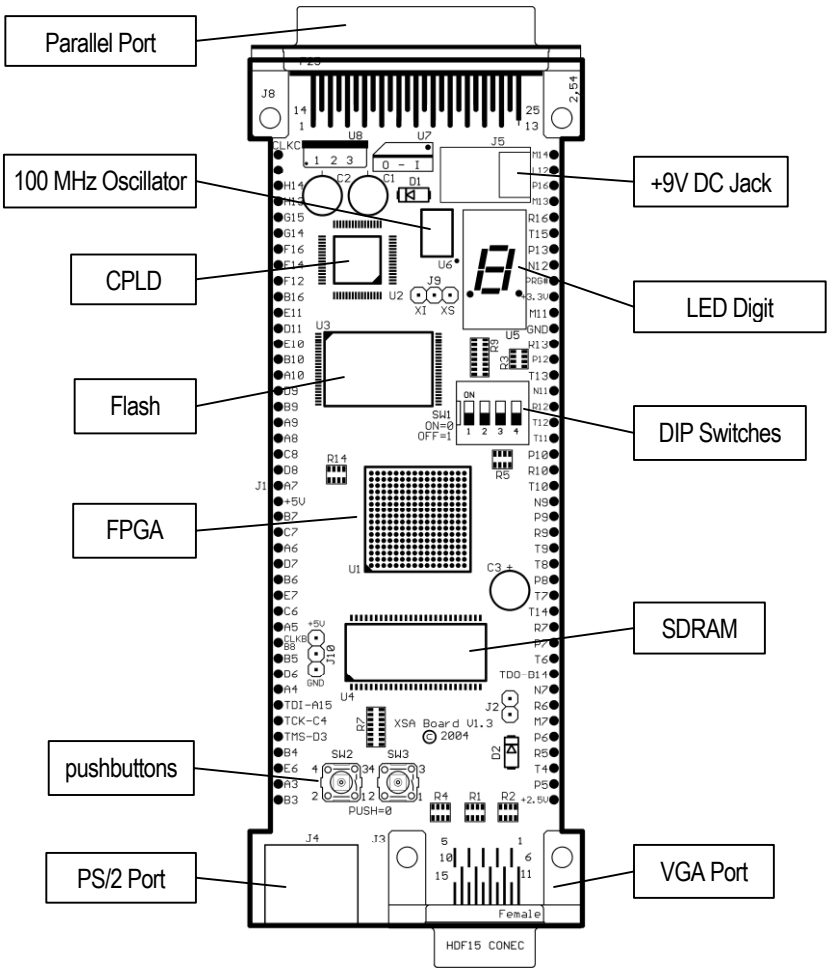
The two rows of pins from your XSA-200 Board can be plugged into a solderless protoboard with holes spaced at 0.1" intervals. (One of the A.C.E. protoboards from 3M is a good choice.) Once plugged in, many of the pins of the FPGA are accessible to other circuits on the protoboard. (The labels printed next to the rows of pins on your XSA-200 Board correspond to the pin numbers of the FPGA.) Power can still be supplied to your XSA-200 Board though the 9V DC jack, or power can be applied directly through several pins on the underside of the board. Just connect +5V, +3.3V, +2.5V and ground to the pins of your XSA-200 Board listed in Table 1. (Remove the shunt on jumper J2 if you supply +2.5V from an external source.)

- Table 1: Power supply pins for the XSA-200 Board.

Voltage	Pin	Note
+5V	2	This pin is labeled "+5V".
+3.3V	54	This pin is labeled "+3.3V".
+2.5V	22	This pin is labeled "+2.5V".
GND	52	This pin is labeled "GND".



- Figure 1: External connections to the XSA-200 Board.



• Figure 2: Arrangement of components on the XSA-200 Board.

Connecting a PC to Your XSA-200 Board

The 6' DB25 male-to-male cable included with your XSA-200 Board connects it to a PC. One end of the cable attaches to the parallel port on the PC and the other connects to the female DB-25 connector at the top of the XSA-200 Board as shown in Figure 1.

Connecting a VGA Monitor to Your XSA-200 Board

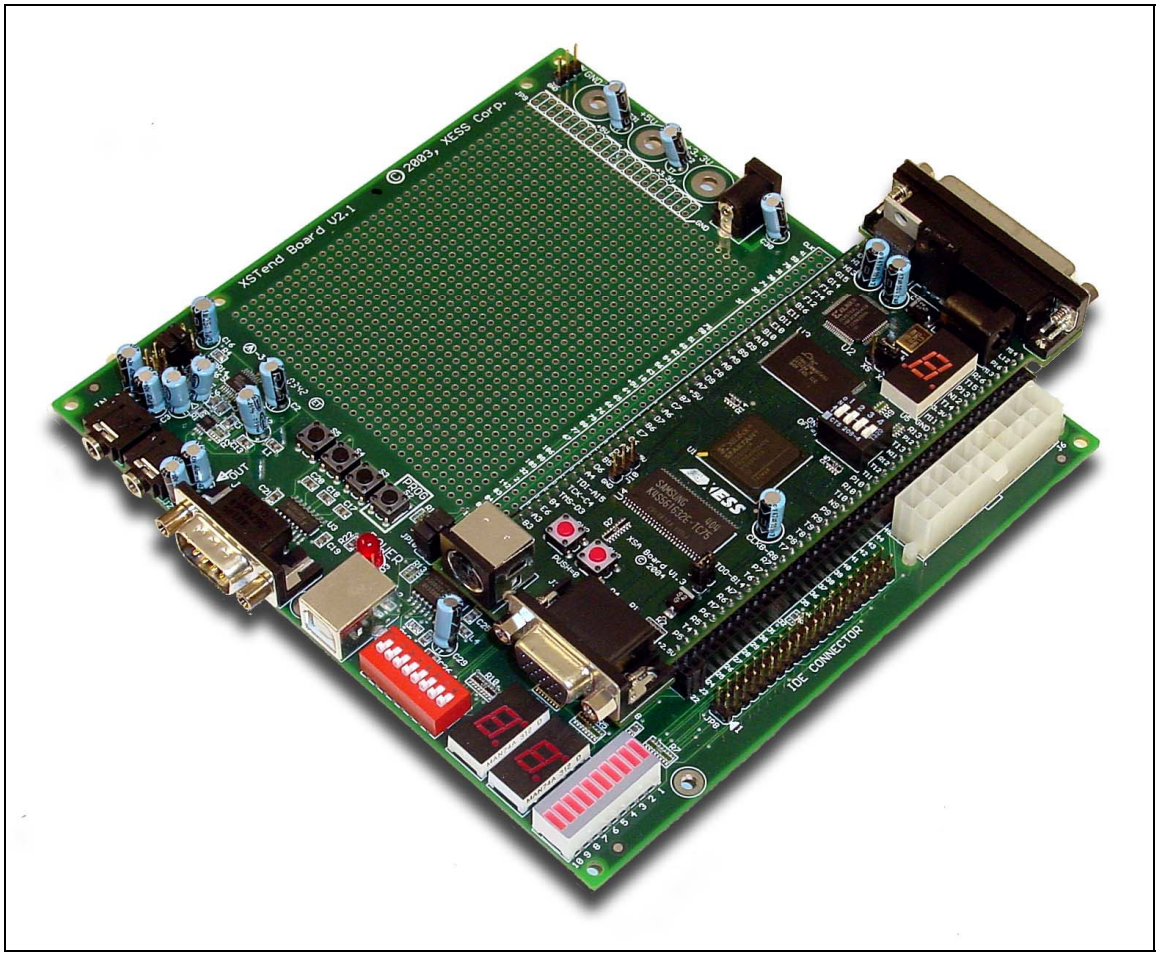
You can display images on a VGA monitor by connecting it to the VGA port at the bottom of your XSA-200 Board (see Figure 1). You will have to create a VGA display circuit for your XSA-200 Board to actually display an image. See this section for details on the VGA port circuitry and creating a VGA display circuit.

Connecting a Mouse or Keyboard to Your XSA-200 Board

You can accept inputs from a keyboard or mouse by connecting it to the PS/2 port at the bottom of your XSA-200 Board (see Figure 1). You will have to create a keyboard or mouse interface circuit to actually receive information on keystrokes or mouse movements. See this section for details on the PS/2 port circuitry and creating a keyboard interface.

Inserting the XSA-200 Board into an XStend Board

If you have the optional XST-2.*x* Board, then the XSA-200 Board is inserted as shown below. Refer to the XST-2.*x* Board Manual for more details.



Setting the Jumpers on Your XSA-200 Board

The default jumper settings shown in Table 2 configure your XSA-200 Board for use in a logic design environment. You will need to change the jumper settings only if you are:

- downloading FPGA bitstreams to your XSA-200 Board using the XILINX iMPACT software;
- changing the power sources for the XSA-200 supply voltages.

• Table 2: Jumper settings for XSA-200 Boards.

Jumper	Setting	Purpose
J2	On (default)	A shunt should be installed if the +2.5V supply voltage is derived from the +3.3V supply.
	Off	The shunt should be removed if the +2.5V supply voltage is applied from an external source through pin 22 of the XSA-200 Board (labeled "+2.5V" at the lower right-hand corner of the board).
J9	1-2 (XI)	The shunt should be installed on pins 1 and 2 (XI) if the XSA-200 Board is to be downloaded using the XILINX iMPACT software.
	2-3 (XS) (default)	The shunt should be installed on pins 2 and 3 (XS) if the XSA-200 Board is to be downloaded using the XESS GXSLOAD software.
J10	N/A	This is a header that provides access to the +5V and GND references on the board. No shunt should be placed on this header.

Testing Your XSA-200 Board

Once your XSA-200 Board is installed and the jumpers are in their default configuration, you can test the board using the GUI-based GXSTEST utility as follows.

You start GXSTEST by clicking on the 

icon placed on the desktop during the XSTOOLS installation. This brings up the window shown below.

Next you select the parallel port that your XSA-200 Board is connected to from the Port pulldown list. GXSTEST starts with parallel port LPT1 as the default, but you can also select LPT2 or LPT3 depending upon the configuration of your PC.

After selecting the parallel port, pick the XSA-200 item in the Board Type pulldown list. Then click on the TEST button to start the testing procedure. GXSTEST will configure the FPGA to perform a test procedure on your XSA-200 Board. Within thirty seconds you will see a **O** displayed on the LED digit if the test completes successfully. Otherwise an **E** will

be displayed if the test fails. A status window will also appear on your PC screen informing you of the success or failure of the test.

If your XSA-200 Board fails the test, you will be shown a checklist of common causes for failure. If none of these causes applies to your situation, then try some of the solutions listed in the XSTOOLS\README.TXT file. If you cannot get your board to pass the test even after taking these steps, then contact XESS Corp for further assistance.

As a result of testing the XSA-200 Board, the CPLD is programmed with the standard parallel port interface found in the XSTOOLS\XSA\200\dwnldpar.svf bitstream file. This is the interface that should be loaded into the CPLD when you want to use it with the GXSLOAD utility.

Setting the XSA-200 Board Clock Oscillator Frequency

Unlike previous versions of the XSA Board, your XSA-200 Board has a fixed-frequency oscillator of 100 MHz. The GXSSETCLK utility cannot be used to change the frequency of the clock sent to the FPGA and CPLD. You can lower the clock frequency by placing a clock-divider circuit in the FPGA or CPLD. See the section on the XSA-200 Board clock circuitry for more details.

3 Programming

This section will show you how to download logic designs into the FPGA and CPLD of your XSA-200 Board and how to download and upload data to and from the SDRAM and Flash devices on the board.

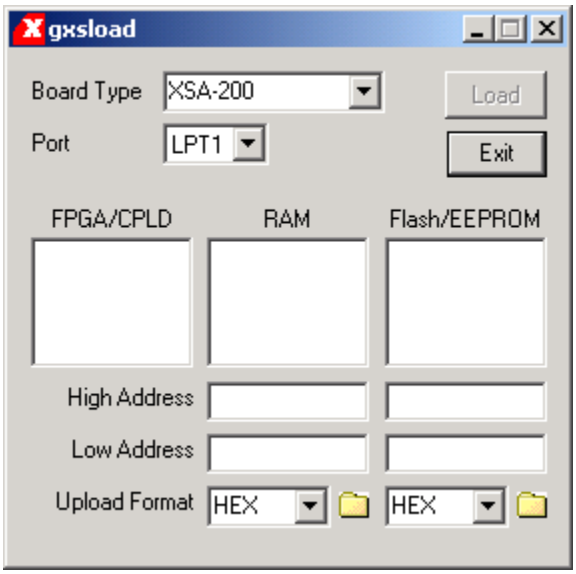
Downloading Bitstreams into the FPGA and CPLD

Downloading Using GXSLOAD

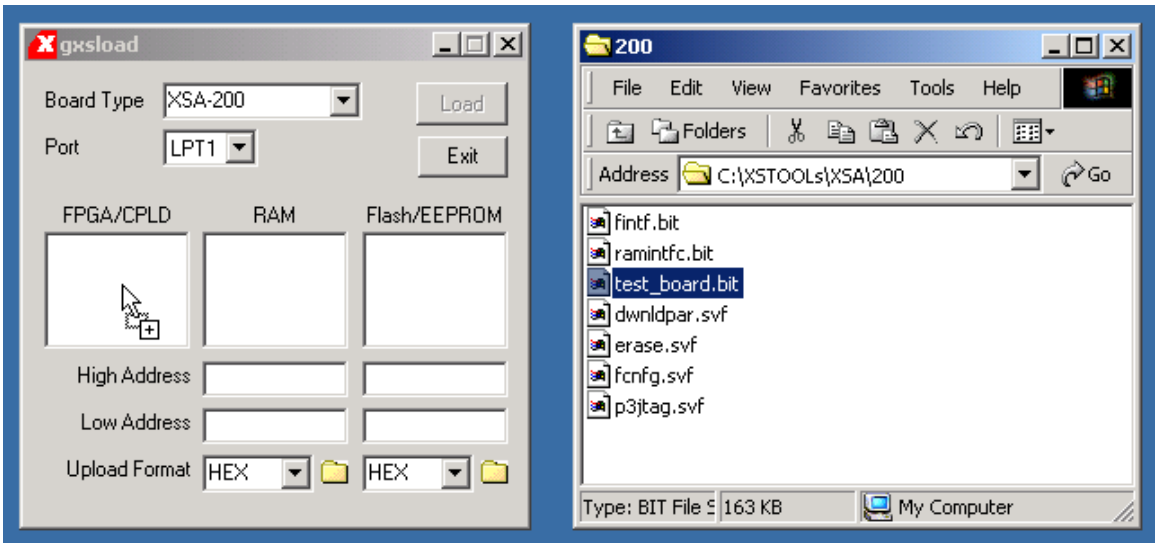
As you develop and test a logic design, you will usually connect the XSA-200 Board to the parallel port of a PC and download the configuration bitstream each time you make changes. You can download a bitstream into your XSA-200 Board using the GXSLOAD utility.

You start GXSLOAD by clicking on the 

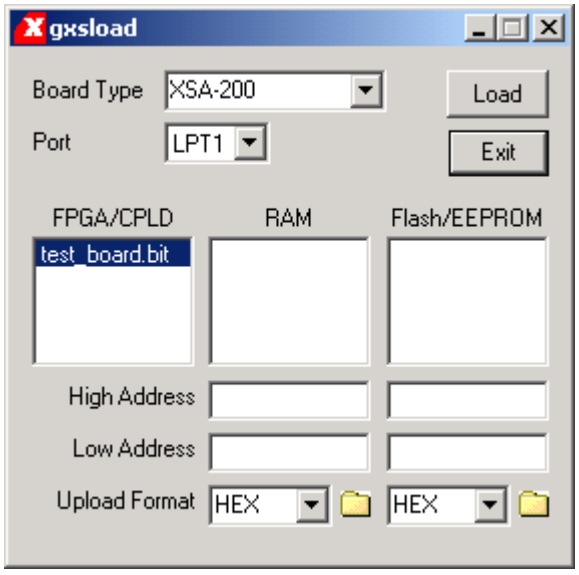
icon placed on the desktop during the XSTOOLS installation. This brings up the window shown below. Select the XSA-200 Board and the parallel port to which it is connected as indicated below.



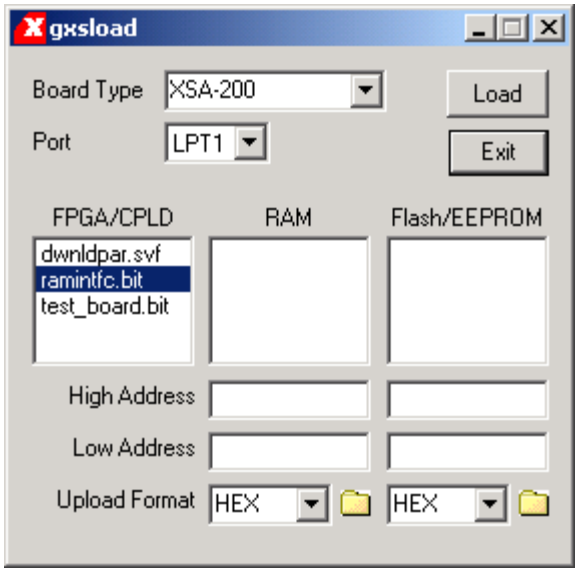
Now you can download bitstream files to the FPGA or CPLD simply by dragging them from their folder and dropping them into the FPGA/CPLD pane of the GXSLOAD window as shown below.



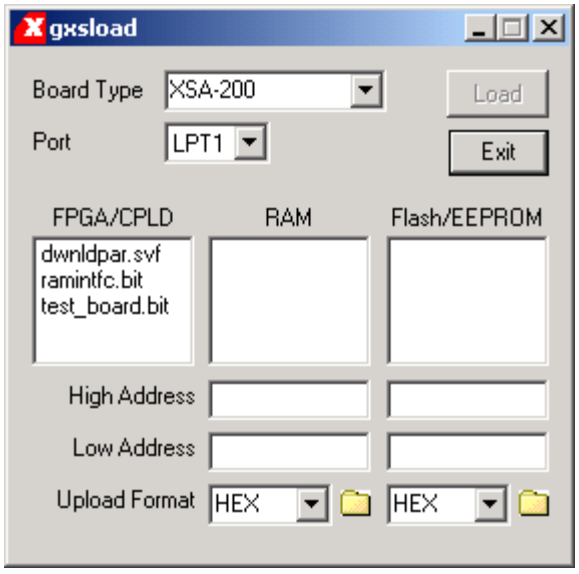
Once you drop the file, the highlighted file name appears in the FPGA/CPLD pane and the Load button in the GXSLOAD window is enabled. Clicking on the Load button will begin sending the bitstream in the file to the XSA-200 Board through the parallel port connection. .BIT files contain configuration bitstreams that are loaded into the FPGA while .SVF files will go to the CPLD. GXSLOAD will reject any non-downloadable files (ones with a suffix other than .BIT or .SVF). During the downloading process, GXSLOAD will display the name of the file and the progress of the current download.



You can drag & drop multiple files into the FPGA/CPLD area. Clicking your mouse on a filename will highlight the name and select it for downloading. Only one file at a time can be selected for downloading.



Double-clicking the highlighted file will deselect it so no file will be downloaded. Doing this disables the Load button.



Downloading Using Xilinx iMPACT

You can use the Xilinx iMPACT software to download bitstreams to the XSA-200 Board. The iMPACT programming tool downloads bitstreams through the JTAG interface of the FPGA so we need to change the parallel port interface by reprogramming the CPLD. Drag & drop the p3jtag.svf file from the XSTOOLS\XSA\200 folder into the FPGA/CPLD pane of the GXSLOAD window. Then click on the Load button and the CPLD will be reprogrammed in less than a minute. Then move the shunt on jumper J9 from the XS to the XI position. At this point you can start iMPACT and it will believe it is connected to the

XSA-200 Board through a Xilinx Parallel Cable III in boundary-scan mode. Follow the instructions for iMPACT to download bitstreams to the FPGA.

Note that the CPLD only needs to be reprogrammed once to support iMPACT because it retains its configuration even when power is removed from the board. (If you want to go back to using the GXSLOAD programming utility, just must move the shunt on J9 back to the XS position and download the XSTOOLS\XSA\200\dwnldpar.svf file into the CPLD.)

Storing Non-Volatile Bitstreams in the Flash

The FPGA on the XSA-200 Board stores its configuration in an on-chip SRAM which is erased whenever power is removed. Once your design is finished, you may want to store the bitstream in the 16 Mbit Flash device on the XSA-200 Board from which the FPGA will be configured each time power is applied.

The Flash is partitioned into four quadrants, each of which can hold a bitstream for the FPGA. Before a bitstream can be downloaded into a quadrant of the Flash, the .BIT file must be converted into an .EXO or .MCS format using one of the following commands:

Quadrant	Address Range	Conversion Command	DIP Switch Setting	
			SW1-1	SW1-2
0	0x000000 – 0x07FFFF	promgen –u 0 file.bit –p exo -w promgen –u 0 file.bit –p mcs –w	ON	ON
1	0x080000 – 0x0FFFFF	promgen –u 80000 file.bit –p exo -w promgen –u 80000 file.bit –p mcs –w	ON	OFF
2	0x100000 – 0x17FFFF	promgen –u 100000 file.bit –p exo -w promgen –u 100000 file.bit –p mcs –w	OFF	ON
3	0x180000 – 0x1FFFFF	promgen –u 180000 file.bit –p exo -w promgen –u 180000 file.bit –p mcs -w	OFF	OFF

In the commands shown above, the bitstream in file.bit is transformed into an .EXO or .MCS formatted file starting at the first address in each quadrant and proceeding upward.

The .EXO or .MCS file is downloaded into the Flash device by dragging it into the Flash/EEPROM pane and clicking on the Load button. This activates the following sequence of steps:

1. The FPGA and CPLD on the XSA-200 Board are reprogrammed to create an interface between the Flash device and the PC parallel port.

2. The entire Flash device is erased.

3. The contents of the .EXO or .MCS file are downloaded into the Flash through the parallel port.

4. The CPLD is reprogrammed with a circuit that configures the FPGA with the contents of the Flash whenever power is applied to the XSA-200 Board. (This configuration loader is stored in the XSTOOLS\XSA\200\fcnfg.svf file.)

Once the Flash download is complete, you must set the DIP switches to select the Flash quadrant containing the FPGA bitstream (see the switch settings in the table above). The

FPGA will be configured with the bitstream in that quadrant whenever power is applied to the board. You can download multiple bitstreams to the Flash and use the switches to select the one to be loaded into the FPGA on power-up.

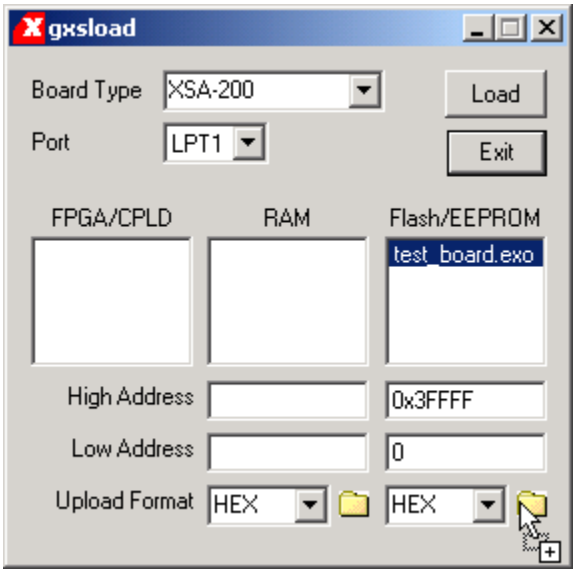
Multiple files can be stored in the Flash device just by dragging them into the Flash/EEPROM area, highlighting the files to be downloaded and clicking the Load button. (Note that anything previously stored in the Flash will be erased by each new download.) This is useful if you need to store information in the Flash in addition to the FPGA bitstream. Files are selected and de-selected for downloading just by clicking on their names in the Flash/EEPROM area. **The address ranges of the data in each file should not overlap or this will corrupt the data stored in the Flash device!**

You can also examine the contents of the Flash device by uploading it to the PC. To upload data from an address range in the Flash, type the upper and lower bounds of the range into the High Address and Low Address fields located below the Flash/EEPROM pane, and select the format for the uploaded data from the Upload Format pulldown list. Then click on the file icon and drag & drop it into any folder. This activates the following sequence of steps:

1. The CPLD and FPGA on the XSA-200 Board are reprogrammed to create an interface between the Flash device and the PC parallel port.

2. The Flash data between the high and low addresses (inclusive) is uploaded through the parallel port.

3. The uploaded data is stored in a file named FLSHUPLD with an extension that reflects the file format.



The uploaded data can be stored in the following formats:

MCS: Intel hexadecimal file format. This is the same format generated by the promgen utility with the –p mcs option.

HEX: Identical to MCS format.

EXO-16: Motorola S-record format with 16-bit addresses (suitable for 64 KByte uploads only).

EXO-24: Motorola S-record format with 24-bit addresses. This is the same format generated by the promgen utility with the –p exo option.

EXO-32: Motorola S-record format with 32-bit addresses.

XESS-16: XESS hexadecimal format with 16-bit addresses. (This is a simplified file format that does not use checksums.)

XESS-24: XESS hexadecimal format with 24-bit addresses.

XESS-32: XESS hexadecimal format with 32-bit addresses.

After the data is uploaded from the Flash, the default parallel port interface remains in the CPLD. You will need to reprogram the CPLD with the configuration loader bitstream in XSTOOLS\XSA\200\fcnfg.svf if you want the FPGA to be configured from Flash whenever power is applied.

Downloading and Uploading Data to the SDRAM

The XSA-200 Board contains a 256 Mbit, synchronous DRAM (16M x 16 SDRAM) whose contents can be downloaded and uploaded by GXSLOAD. This is useful for initializing the SDRAM with data for use by the FPGA and then reading the SDRAM contents after the FPGA has operated upon it. The SDRAM is loaded with data by dragging & dropping one or more .EXO, .MCS, .HEX, and/or .XES files into the RAM pane of the GXSLOAD window and then clicking on the Load button. This activates the following sequence of steps:

1. The FPGA is reprogrammed to create an interface between the SDRAM and the PC parallel port. (This interface is stored in the XSTOOLS\XSA\200\ramintfc.bit bitstream file. **The CPLD must have previously been loaded with the dwnldpar.svf file found in the same folder.**)

2. The contents of the .EXO, .MCS, .HEX or .XES files are downloaded into the SDRAM through the parallel port. **The data in the files will overwrite each other if their address ranges overlap.**

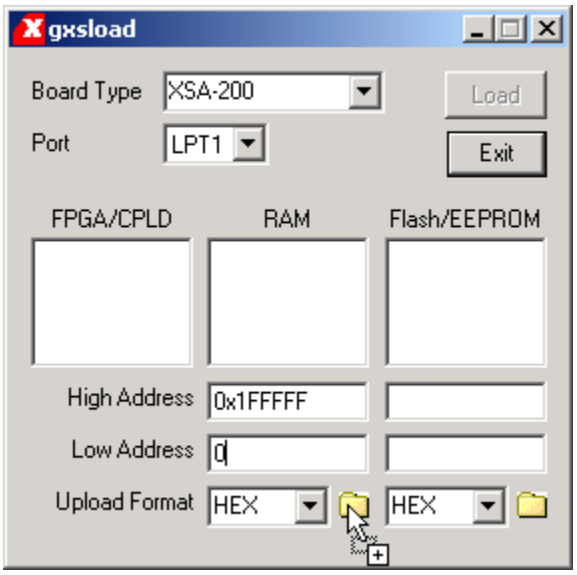
3. If any file is highlighted in the FPGA/CPLD pane, then this bitstream is loaded into the FPGA or CPLD on the XSA-200 Board. Otherwise the FPGA remains configured as an interface between the PC and the SDRAM.

You can also examine the contents of the SDRAM device by uploading it to the PC. To upload data from an address range in the SDRAM, type the upper and lower bounds of the range into the High Address and Low Address fields below the RAM pane, and select the format for the uploaded data from the Upload Format pulldown list. Then click on the file icon and drag & drop it into any folder. This activates the following sequence of steps:

1. The FPGA is reprogrammed to create an interface between the SDRAM device and the PC parallel port.

2. The SDRAM data between the high and low addresses (inclusive) is uploaded through the parallel port.

3. The uploaded data is stored in a file named RAMUPLD with an extension that reflects the file format.



The 16-bit data words in the SDRAM are mapped into the eight-bit data format of the .HEX, .MCS, .EXO and .XES files using a Big Endian style. That is, the 16-bit word at location *N* in the SDRAM is stored in the eight-bit file with the upper eight bits at address *2N* and the lower eight bits at address *2N+1*. This byte-ordering applies for both RAM uploads and downloads.

4

Programmer's Models

This section describes the various sections of the XSA-200 Board and shows how the I/O of the FPGA and CPLD are connected to the rest of the circuitry. The schematics which follow are less detailed so as to simplify the descriptions. For more information, you can find a table of FPGA and CPLD pin connections and detailed schematics at the end of this manual.

XSA-200 Board Organization

The XSA-200 Board contains the following components:

FPGA: This is the main repository of programmable logic on the XSA-200 Board.

CPLD: This manages the interface between the PC parallel port and the rest of the XSA-200 Board. It can also configure the FPGA with a bitstream from Flash.

Oscillator: A fixed-frequency oscillator generates the master clock for the XSA-200 Board.

SDRAM: A 256 Mbit SDRAM provides volatile data storage accessible by the FPGA.

Flash: A 16 Mbit Flash device provides non-volatile storage for data and FPGA configuration bitstreams.

LED: A seven-segment LED allows visible feedback as the XSA-200 Board operates.

DIP switch: A four-position DIP switch passes settings to the XSA-200 Board and controls the upper address bits of the Flash device.

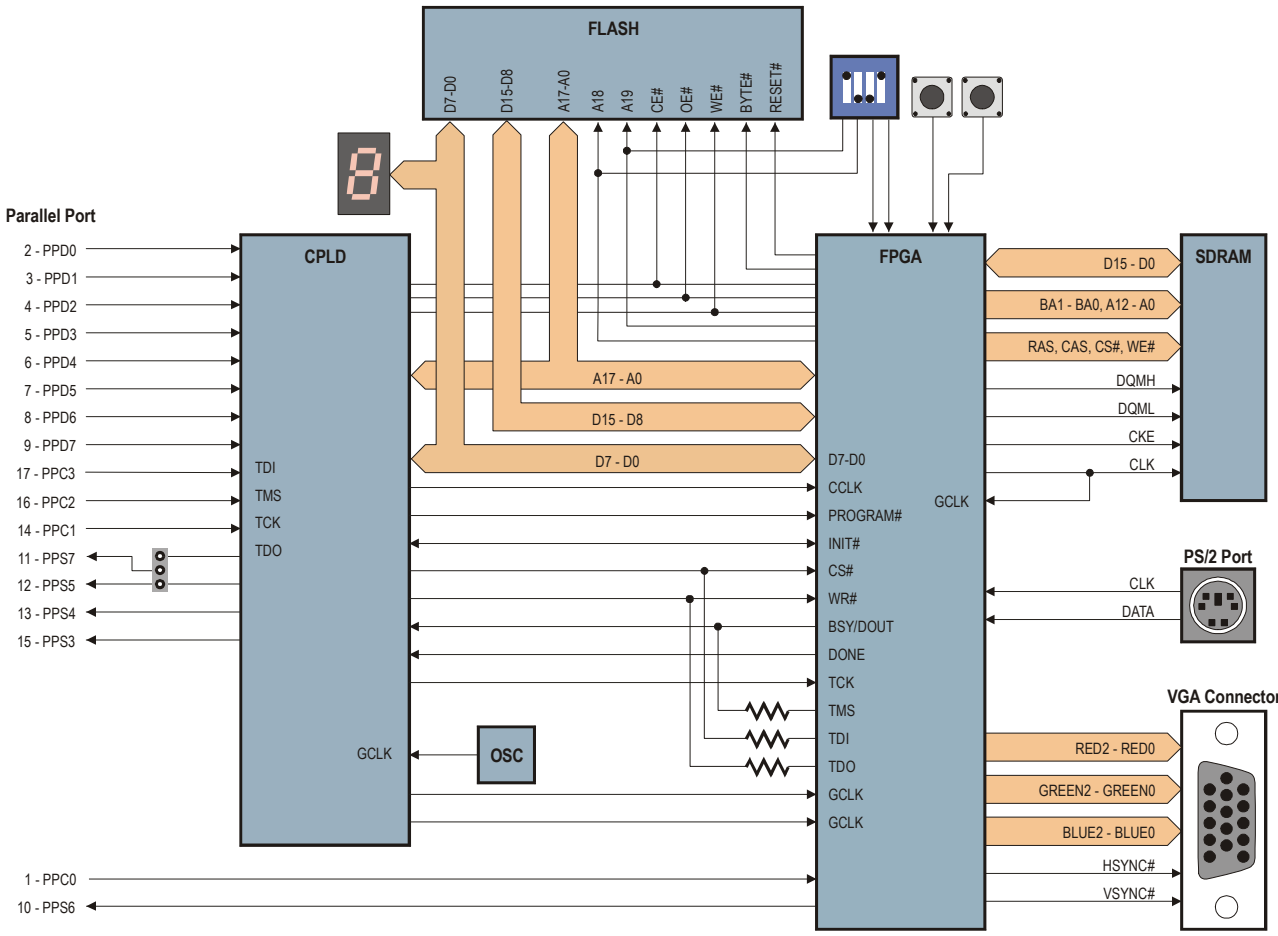
Pushbuttons: Two pushbuttons send momentary contact information to the FPGA.

PS/2 Port: A keyboard or mouse can interface to the XSA-200 Board through this port.

VGA Port: The XSA-200 Board can send signals to display graphics on a VGA monitor through this port.

Parallel Port: This is the main interface for passing configuration bitstreams and data to and from the XSA-200 Board.

Prototyping Header: Many of the FPGA I/O pins are connected to the 84 pins on the bottom of the XSA-200 Board that are meant to mate with solderless breadboards or an XST-2 Board.



• Figure 3: XSA-200 Board programmer's model.

Programmable logic: FPGA and CPLD

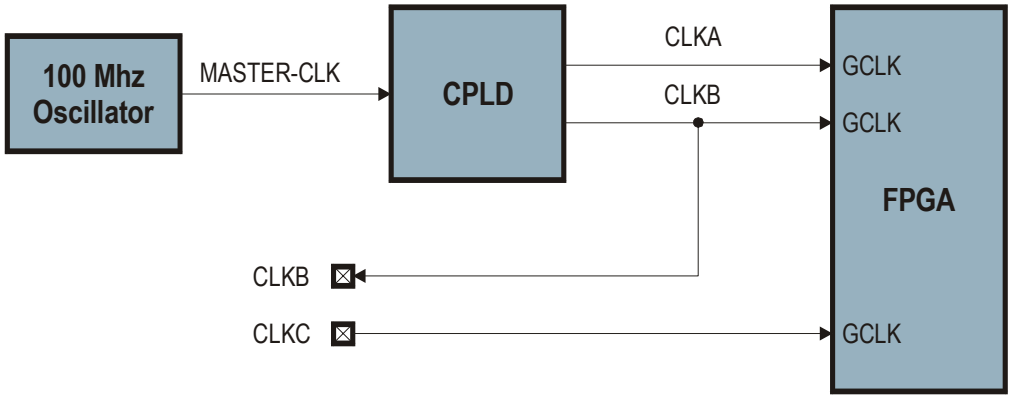
The XSA-200 Board contains two programmable logic chips:

- A 200-Kgate XILINX Spartan-II FPGA in a 256-pin BGA package (XC2S200-5FG256) is the main repository of programmable logic on the XSA-200 Board.
- A XILINX XC9500XL CPLD (XC9572XL-10VQ64) is used to manage the configuration of the FPGA via the parallel port. In stand-alone mode, the CPLD also configures the FPGA with a bitstream from the Flash RAM.

100 MHz Fixed-Frequency Oscillator

An oscillator provides a fixed, 100 MHz clock signal to a dedicated clock input of the CPLD. From this clock, the CPLD generates two clock signals, CLKA and CLKB, that go to dedicated clock inputs of the FPGA. This allows the CPLD to control the FPGA clocks. By default, the CPLD outputs 100 MHz and 50 MHz clocks on CLKA and CLKB, respectively. The clock-divider circuit in the CPLD can be reprogrammed to send lower-frequency clocks to the FPGA if desired.

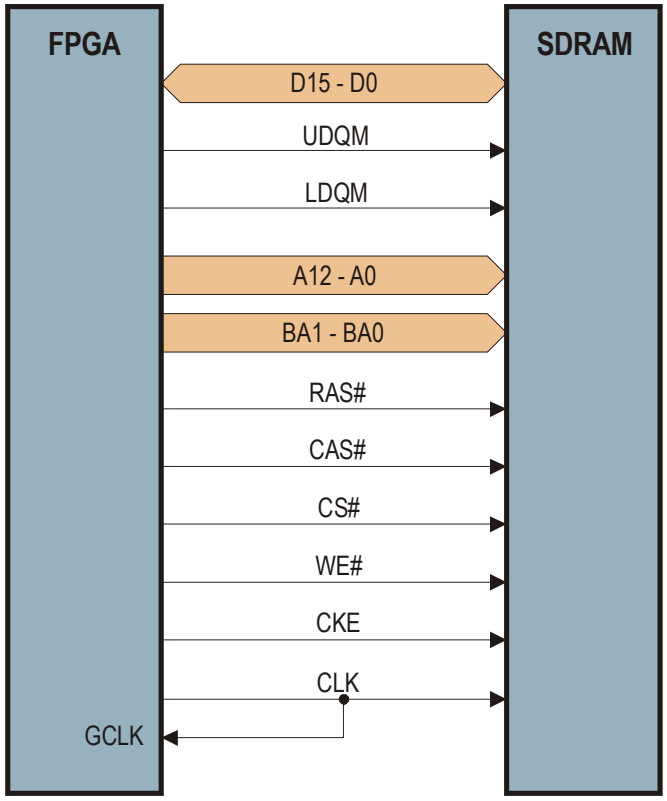
The CLKB signal also exits through a pin on the prototyping header, so it can be used as a clock for an external system connected to the XSA-200 Board. Or the external system can send a clock directly to the FPGA through the dedicated CLKC pin of the prototyping header.



Synchronous DRAM

The XSA-200 Board incorporates a 16M x 16 SDRAM (K4S561632ETC75) that connects solely to the FPGA as shown below. Note that the clock signal is re-routed back to a dedicated clock input of the FPGA to compensate for clock delays to the SDRAM, thus allowing synchronization of the FPGA's internal operations with the SDRAM operations.

This application note describes an SDRAM controller that makes the SDRAM appear like a simple static RAM to the rest of the circuitry in the FPGA.



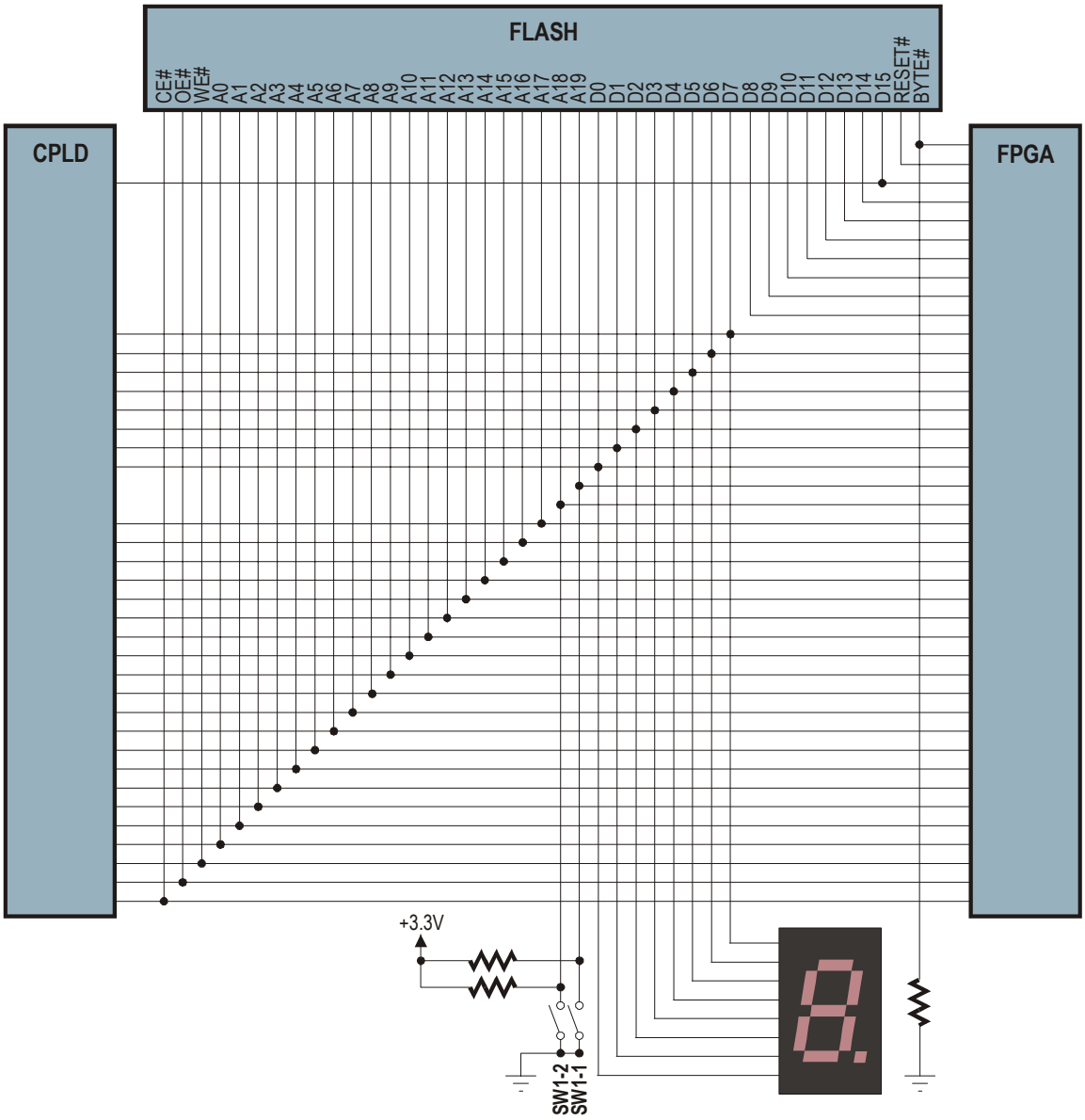
Flash RAM

The CPLD and FPGA connect to a 16 Mbit Flash RAM (S29AL016M10TAI020) that operates in either byte mode (2M x 8) or word mode (1M x 16). The CPLD uses the byte mode as it only has access to the lower eight bits of the Flash data bus, while the FPGA connects to the entire 16-bit data bus and can select either mode using the BYTE# control line.

The FPGA has access to the entire Flash address bus so it can read or write any location. For this reason, the FPGA is used to pass data between the parallel port and the Flash when GXSLOAD downloads/uploads files to/from the Flash. The CPLD, however, is not connected to the upper two address lines so it can only access a quadrant of the Flash. The quadrant is selected by two DIP switches connected to the upper address lines. On power-up in stand-alone mode, the CPLD configures the FPGA with a bitstream retrieved from the selected quadrant, so the DIP switches can be used to select between four

separate bitstreams stored in the Flash. (See the application note on the XSA Board Flash configuration circuit for more details on this.)

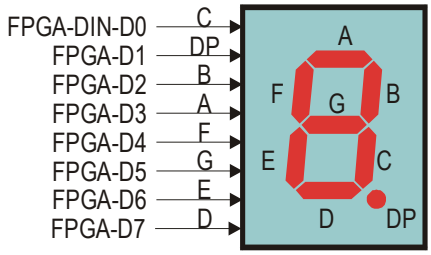
After power-up, any application circuit loaded into the FPGA can read and/or write the Flash. To avoid contention, the CPLD is programmed to release control of all Flash address/data/control lines whenever the FPGA lowers the Flash CE# line. When the Flash is disabled by raising CE#, the I/O lines connected to the Flash are available for general-purpose communication between the FPGA and the CPLD.

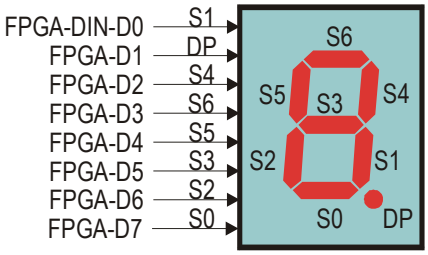


Seven-Segment LED

The XSA-200 Board has a 7-segment LED digit for use by the FPGA or the CPLD. Segments of the LED glow when a logic-high level is applied to them.

The LED shares the same eight-bit data bus that interconnects the CPLD, the FPGA configuration port and the lower-byte of the Flash RAM data bus. The connections between the LED segments and the data bus are shown below. (We use two distinct labelings of the LED segments in our documentation and design examples, so we show the connections for both.)





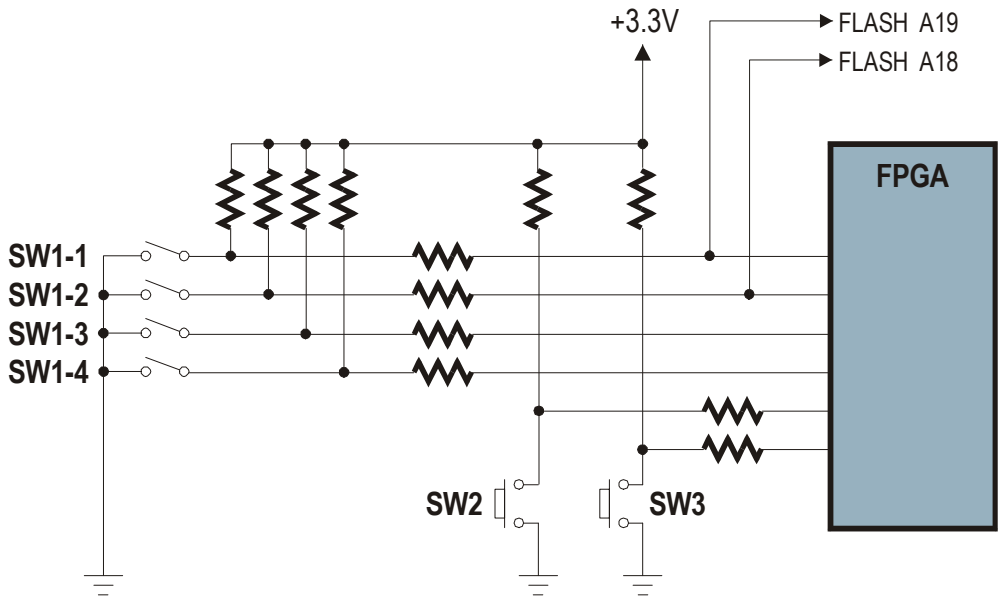
DIP Switches and Pushbuttons

Four DIP switches are attached to the FPGA. When closed (ON), each switch pulls the connected pin of the FPGA to ground. The pin is pulled high through a resistor when the switch is open (OFF).

Two of the DIP switches also connect to the upper two bits of the Flash address bus. These DIP switches are used to select the Flash quadrant holding a bitstream that will be loaded into the FPGA by the CPLD on power-up.

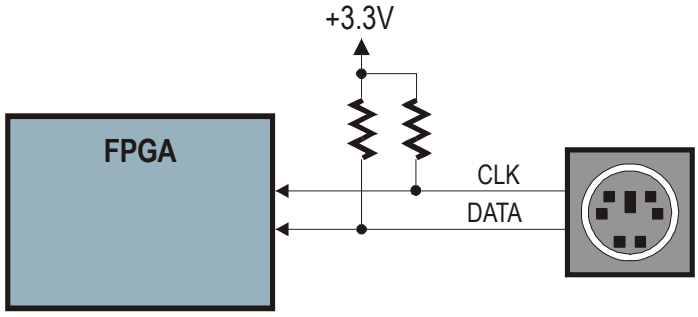
The FPGA also connects to two pushbuttons. Each pushbutton applies a low level to its FPGA pin when pressed and a resistor pulls the pin to a high level when the pushbutton is released.

Small resistors are placed in series between the FPGA and the switches and pushbuttons to prevent damage if the FPGA tries to drive a pin that is being pulled low.



PS/2 Port

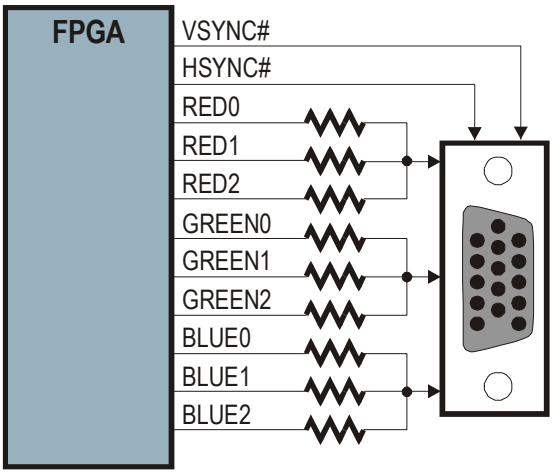
A PS/2 port provides the FPGA with an interface to either a keyboard or a mouse. The FPGA receives two signals from the PS/2 interface: a clock signal and a serial data stream that is synchronized with the falling edge of the clock. (For more details on using the PS/2 port and a simple circuit for receiving keystroke information from a keyboard, see this application note.)



VGA Port

The FPGA can generate a video signal for display on a VGA monitor. The FPGA outputs three bits each of red, green, and blue color information to a simple resistor-ladder DAC. This provides a palette of $2^3 \times 2^3 \times 2^3 = 512$ colors. The outputs of the DAC are sent to the

RGB inputs of a VGA monitor. The FPGA also generates the horizontal and vertical sync pulses (HSYNC#, VSYNC#). (See this application note for more details on a simple circuit for generating VGA signals that displays an image stored in SDRAM.)



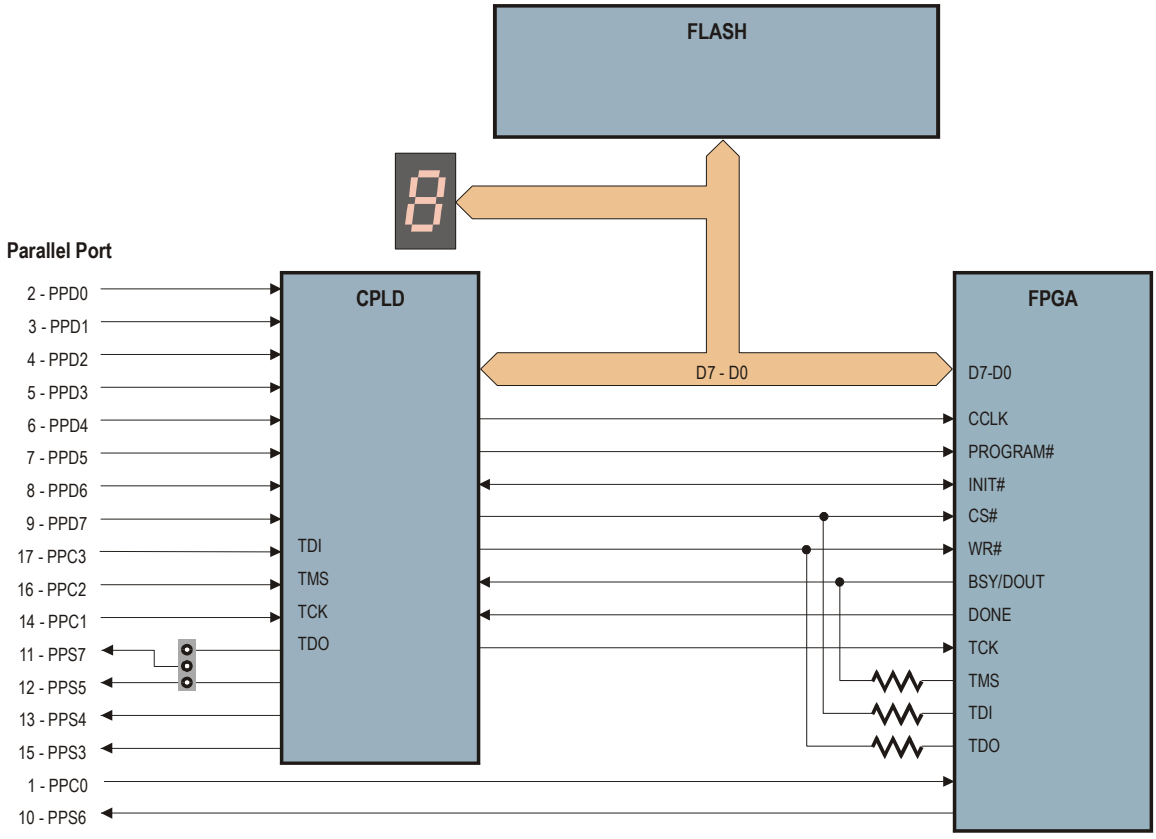
Parallel Port

The parallel port is the main interface for communicating between the XSA-200 Board and a PC. Control line C0 and status line S6 connect directly to the FPGA and can be used for bidirectional communication between the FPGA and PC. The CPLD handles the fifteen remaining active lines of the parallel port as follows.

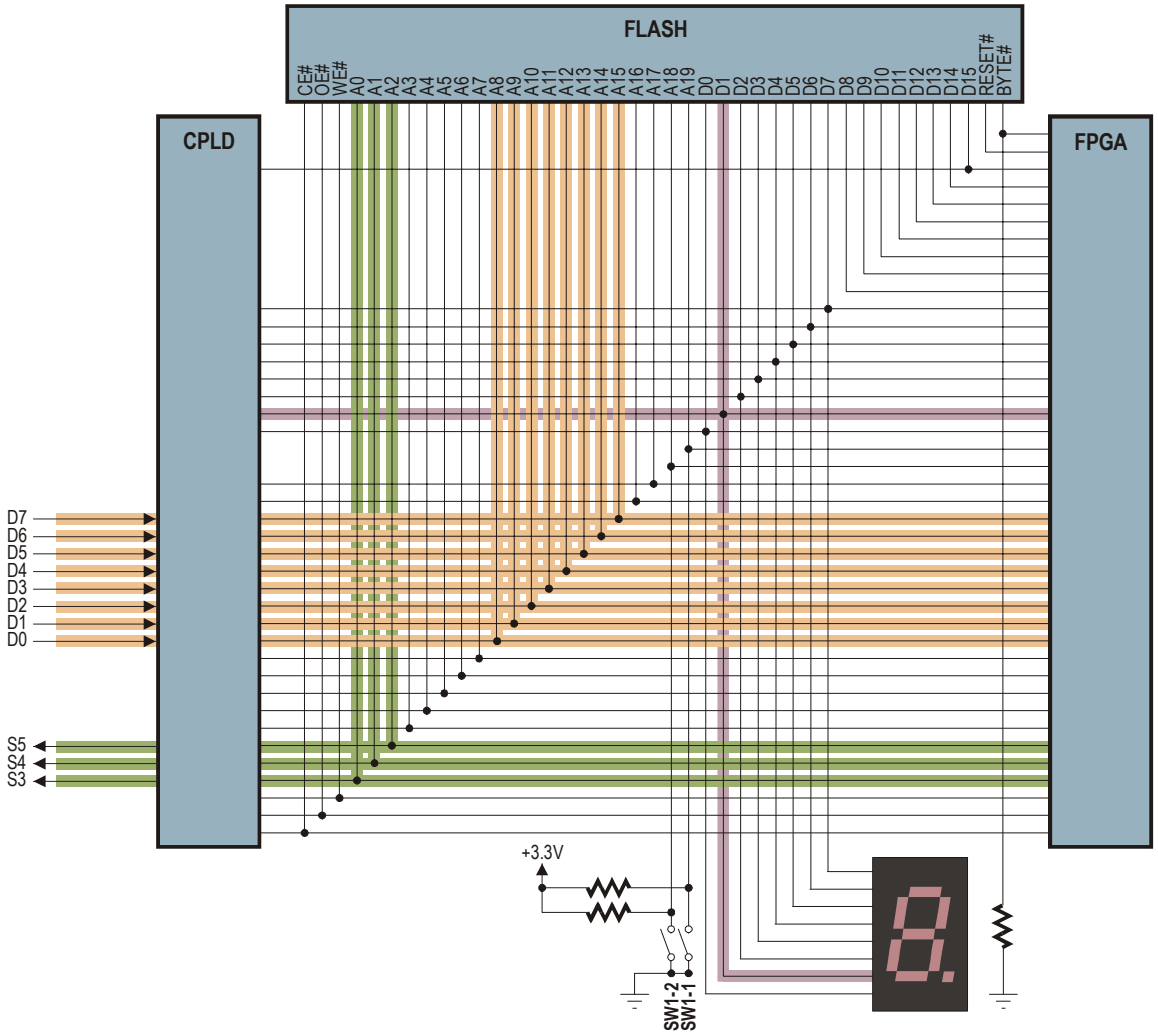
Three of the parallel port control lines, C1–C3, connect to the JTAG pins through which the CPLD is programmed. The C1 control line clocks configuration data presented on the C3 line into the CPLD while the C2 signal steers the actions of the CPLD programming state machine. Meanwhile, information from the CPLD returns to the PC through status line S7.

The eight data lines, D0–D7, and the remaining three status lines, S3–S5, connect to general-purpose pins of the CPLD. The CPLD can be programmed to act as an interface between the FPGA and the parallel port. The CPLD connects to the FPGA configuration pins so it can pass bitstreams from the parallel port to the FPGA. The actual configuration data is presented to the FPGA on the same 8-bit bus that also connects to the Flash and seven-segment LED. The CPLD also drives the configuration pins (CCLK, PROGRAM#, CS#, and WR#) that sequence the loading of a bitstream into the FPGA. The CPLD can monitor the status of the bitstream download through the INIT#, DONE, and BSY/DOUT pins.

The CPLD also has access to the FPGA's JTAG pins: TCK, TMS, TDI, and TDO. The TMS, TDI, and TDO pins share the connections with the BSY/DOUT, CS#, and WR# pins. The CPLD can be programmed with an interface that allows configuration of the FPGA through the JTAG pins using the XILINX iMPACT software (see this application note for more details). Jumper J9 allows the connection of status pin S7 to the general-purpose CPLD pin that also drives status pin S5. This is required by the iMPACT software so it can check for the presence of the downloading cable.



After the FPGA is configured with a bitstream and the DONE pin goes high, the CPLD switches into a mode that connects the parallel port data and status pins to the FPGA. This lets the PC pass data to the FPGA over the parallel port data lines while receiving data from the FPGA over the status lines. The active connections between the FPGA, CPLD and the parallel port after configuration are shown below.



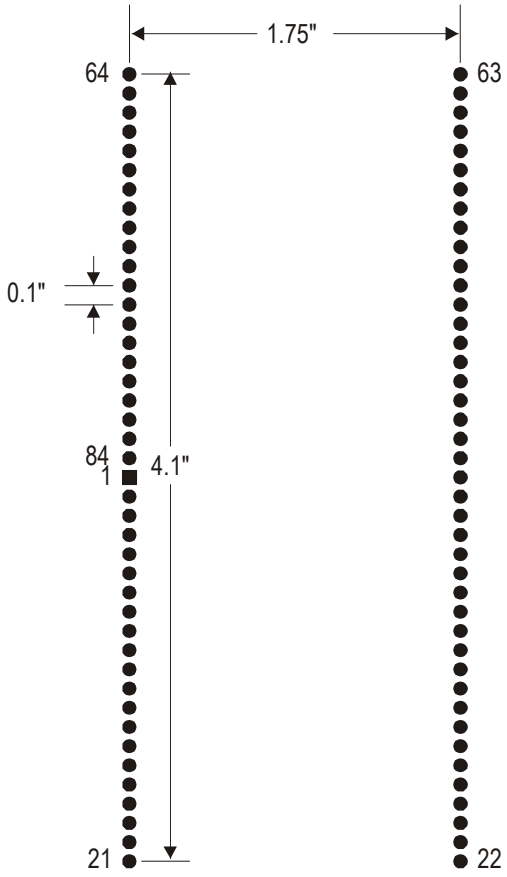
The FPGA sends data to the PC by driving logic levels onto the A0, A1 and A2 Flash address lines which pass through the CPLD and onto the parallel port status lines S3, S4 and S5, respectively. Conversely, the PC sends data to the FPGA on parallel port data lines D7–D0 and the data passes through the CPLD and ends up on the A15–A8 Flash address lines, respectively. The FPGA should never drive A15–A8 unless it is accessing the Flash, otherwise the CPLD and/or FPGA could be damaged. The CPLD can sense when the FPGA lowers the Flash CE# and it will release the Flash address lines so the FPGA can drive them without contention.

The CPLD also drives the decimal-point of the LED display (connected to Flash data line D1) to give a visual indication when the FPGA is configured with a valid bitstream. Unless it is accessing the Flash RAM, the FPGA should never drive Flash data line D1 to a low logic level or it may damage itself or the CPLD. But when the FPGA lowers the Flash CE#, the CPLD will stop driving the LED decimal-point to allow the FPGA access to data line D1 of the Flash.

For more details on how the CPLD manages the interface between the parallel port and the FPGA both before and after device configuration, see this application note.

Prototyping Header

The pins of the FPGA are accessible through the 84-pin prototyping header on the underside of the XSA-200 Board. Pin 1 of the header (denoted by a square pad) is located in the middle of the left-hand edge of the board and the remaining 83 pins are arranged counter-clockwise around the periphery. The physical dimensions of the prototyping header and the pin arrangement are shown below.



A subset of the FPGA pins connects to the prototyping header. These pins are not connected to any of the other components on the XSA-200 Board so they are completely free to use for I/O operations with external systems without any restrictions.

The number of the FPGA pin connected to a given header pin is printed next to the header pin on the board. This makes it easier to find a given FPGA pin when you want to connect it to an external system.

A

XSA-200 Pin Connections

The following tables list the pin numbers of the FPGA and CPLD along with the pin names of the other chips that they connect to on the XSA-200 Board and the XST-2.x Board.

Connections Between the FPGA, CPLD and other Components on the XSA-200 Board …												and the XST-2.x Board						
Net Name	FPGA	CPLD	LEDs	Switch / Buttons	SDRAM	Flash	Osc	Parallel Port	PS/2 Port	VGA Port	Proto. Header	LEDs	Switch Button	SRAM	IDE Intfc.	Stereo Codec	USB	Serial Port
+2.5V											22							
+3.3V											54							
+5V											2							
CLKA	R8 (GCK1)	60																
CLKB	B8 (GCK3)	57									11							
CLKC	C9 (GCK2)										64							
CPLD-TDO		53 (TDO)																
FLASH-A0	L13	64				A0		S3										
FLASH-A1	C10	50				A1		S4										
FLASH-A2	D10	49				A2		S5										
FLASH-A3	A11	48				A3												
FLASH-A4	B11	47				A4												
FLASH-A5	C11	46				A5												
FLASH-A6	A12	45				A6												
FLASH-A7	B12	44				A7												
FLASH-A8	E13	40				A8		D0										
FLASH-A9	C16	39				A9		D1										
FLASH-A10	E14	38				A10		D2										
FLASH-A11	D16	36				A11		D3										
FLASH-A12	F13	35				A12		D4										
FLASH-A13	E15	34				A13		D5										
FLASH-A14	G12	33				A14		D6										
FLASH-A15	G13	32				A15		D7										
FLASH-A16	H16	13				A16												
FLASH-A17	A13	43				A17												
FLASH-A18	M10			SW1-2		A18												
FLASH-A19	P11			SW1-1		A19												
FLASH-D8	K12					DQ8												
FLASH-D9	L15					DQ9												
FLASH-D10	K13					DQ10												
FLASH-D11	L16					DQ11												
FLASH-D12	K14					DQ12												
FLASH-D13	K15					DQ13												
FLASH-D14	K16					DQ14												
FLASH-D15	J14	12				DQ15/A-1												
FLASH-BYTE#	J15					BYTE#												
FLASH-CE#	M15	1				CE#												
FLASH-OE#	L14	2				OE#												
FLASH-RDY	C12					RY/BY#												
FLASH-RESET#	A14					RESET#												
FLASH-WE#	D12	42				WE#												
FPGA-CCLK	D15 (CCLK)	59																
FPGA-CS#	B13 (CS#)	52																
FPGA-DIN-D0	D14 (DIN/D0)	4	LED-C (S1)			DQ0												
FPGA-D1	E16 (D1)	5	LED-DP			DQ1												
FPGA-D2	F15 (D2)	6	LED-B (S4)			DQ2												
FPGA-D3	G16 (D3)	7	LED-A (S6)			DQ3												
FPGA-D4	J16 (D4)	8	LED-F (S5)			DQ4												
FPGA-D5	M16 (D5)	9	LED-G (S3)			DQ5												
FPGA-D6	N16 (D6)	10	LED-E (S2)			DQ6												
FPGA-D7	N14 (D7)	11	LED-D (S0)			DQ7												
FPGA-DONE	R14 (DONE)	63																
FPGA-DOUT-BSY	C15 (DOUT/BUSY)	51																
FPGA-INIT#	N15 (INIT#)	61																
FPGA-PROG#	P15 (PROGRAM#)	62									55		PUSHB1					
FPGA-TCK	C4 (TCK)	58									16							
FPGA-TDI	A15 (TDI)	52									15							
FPGA-TDO	B14 (TDO)	56									30							
FPGA-TMS	D3 (TMS)	51									17							
FPGA-WR#	C13 (WRITE#)	56																
GND											52							
MASTER-CLK		17					OUT											
PP-C0	H15							C0										
PP-C1		30 (TCK)						C1										
PP-C2		29 (TMS)						C2										
PP-C3		28 (TDI)						C3										
PP-D0		31						D0										
PP-D1		25						D1										
PP-D2		24						D2										
PP-D3		23						D3										

Connections Between the FPGA, CPLD and other Components on the XSA-200 Board …												and the XST-2.x Board						
Net Name	FPGA	CPLD	LEDs	Switch / Buttons	SDRAM	Flash	Osc	Parallel Port	PS/2 Port	VGA Port	Proto. Header	LEDs	Switch Button	SRAM	IDE Intfc.	Stereo Codec	USB	Serial Port
PP-D4		22						D4										
PP-D5		20						D5										
PP-D6		19						D6										
PP-D7		18						D7										
PP-S3		27						S3										
PP-S4		16						S4										
PP-S5		15						S5										
PP-S6	J13							S6										
PP-S7								S7										
PS2-CLK	F4								CLK									
PS2-DATA	E1								DATA									
SDRAM-A0	L4				A0													
SDRAM-A1	N1				A1													
SDRAM-A2	N2				A2													
SDRAM-A3	M4				A3													
SDRAM-A4	T5				A4													
SDRAM-A5	N6				A5													
SDRAM-A6	M6				A6													
SDRAM-A7	T3				A7													
SDRAM-A8	N5				A8													
SDRAM-A9	P1				A9													
SDRAM-A10	L3				A10													
SDRAM-A11	M3				A11													
SDRAM-A12	M2				A12													
SDRAM-BA0	K2				BA0													
SDRAM-BA1	L2				BA1													
SDRAM-D0	C2				DQ0													
SDRAM-D1	C1				DQ1													
SDRAM-D2	F5				DQ2													
SDRAM-D3	D1				DQ3													
SDRAM-D4	F3				DQ4													
SDRAM-D5	F2				DQ5													
SDRAM-D6	F1				DQ6													
SDRAM-D7	G3				DQ7													
SDRAM-D8	G4				DQ8													
SDRAM-D9	G5				DQ9													
SDRAM-D10	E2				DQ10													
SDRAM-D11	E4				DQ11													
SDRAM-D12	B1				DQ12													
SDRAM-D13	A2				DQ13													
SDRAM-D14	D5				DQ14													
SDRAM-D15	C5				DQ15													
SDRAM-CKE	L1				CKE													
SDRAM-CLK	J4				CLK													
SDRAM-CLKFB	N8 (GCK0)				CLK													
SDRAM-CS#	J3				CS#													
SDRAM-CAS#	H3				CAS#													
SDRAM-RAS#	J2				RAS#													
SDRAM-WE#	G1				WE#													
SDRAM-LDQM	G2				LDQM													
SDRAM-UDQM	H1				UDQM													
SW1-3	R11			SW1-3														
SW1-4	N10			SW1-4														
SW2	E3			SW2														
SW3	D2			SW3														
VGA-BLUE0	H4									BLUE0								
VGA-BLUE1	K3									BLUE1								
VGA-BLUE2	L5									BLUE2								
VGA-GREEN0	H2									GREEN0								
VGA-GREEN1	K5									GREEN1								
VGA-GREEN2	R1									GREEN2								
VGA-RED0	J1									RED0								
VGA-RED1	M1									RED1								
VGA-RED2	T2									RED2								
VGA-HSYNC#	K4									HSYNC#								
VGA-VSYNC#	K1									VSYNC#								
PROTO1	A7										1							
PROTO3	B7										3	LED1-D		RAM-A4	IDE-DA0			
PROTO4	C7										4	LED1-C		RAM-A3	IDE-D15			
PROTO5	A6										5		DIPSW5	RAM-A2	IDE-DA1			

Connections Between the FPGA, CPLD and other Components on the XSA-200 Board …												and the XST-2.x Board						
Net Name	FPGA	CPLD	LEDs	Switch / Buttons	SDRAM	Flash	Osc	Parallel Port	PS/2 Port	VGA Port	Proto. Header	LEDs	Switch Button	SRAM	IDE Intfc.	Stereo Codec	USB	Serial Port
PROTO6	D7										6	LED1-E				AUDIO-SDTO		
PROTO7	B6										7		DIPSW8	RAM-CE#				
PROTO8	E7										8		DIPSW7					RS232-RT
PROTO9	C6										9		DIPSW6			AUDIO-MCLK		
PROTO10	A5										10	BARLED8		RAM-D2	IDE-D7			
PROTO12	B5										12							
PROTO13	D6										13							
PROTO14	A4										14							
PROTO18	B4										18							RS232-TD
PROTO19	E6										19						USB-SCL	
PROTO20	A3										20						USB-SDA	
PROTO21	B3										21							
PROTO23	P5										23				IDE-DIOR#			
PROTO24	T4										24				IDE-DIOW#			
PROTO25	R5										25							
PROTO26	P6										26							
PROTO27	M7										27							
PROTO28	R6										28							
PROTO29	N7										29							
PROTO31	T6										31							
PROTO32	P7										32							
PROTO33	R7										33							
PROTO34	T14										34							
PROTO35	T7										35	BARLED5		RAM-D3	IDE-D4			
PROTO36	P8										36		PUSHB4					
PROTO37	T8										37		PUSHB3					
PROTO38	T9										38	BARLED4		RAM-D4	IDE-D3			
PROTO39	R9										39	BARLED3		RAM-D5	IDE-D2			
PROTO40	P9										40	BARLED2		RAM-D6	IDE-D1			
PROTO41	N9										41	BARLED1		RAM-D7	IDE-D0			
PROTO42	T10										42							
PROTO43	R10										43							
PROTO44	P10										44							
PROTO45	T11										45	LED2-DP		RAM-A1	IDE-DMACK#			
PROTO46	T12										46							
PROTO47	R12										47							
PROTO48	N11										48							
PROTO49	T13										49							
PROTO50	P12										50	LED2-B		RAM-A0	IDE-DMARQ			
PROTO51	R13										51	LED2-E		RAM-A10			USB-INT#	
PROTO53	M11										53							
PROTO56	N12										56	LED2-G		RAM-A11			USB-SUSPEND	
PROTO57	P13										57	LED2-C		RAM-A9	IDE-INTRQ			
PROTO58	T15										58	LED2-F		RAM-A8	IDE-D8			
PROTO59	R16										59	LED2-D		RAM-A13	IDE-D10			
PROTO60	M13										60	LED2-A		RAM-A15	IDE-D11			
PROTO61	P16										61			RAM-OE#	IDE-D9			
PROTO62	L12										62		DIPSW2	RAM-WE#	IDE-D14			
PROTO63	M14										63							
PROTO65											65							
PROTO66	H14										66	BARLED10				AUDIO-LRCK		
PROTO67	H13										67		PUSHB2					
PROTO68	G15										68				IDE-RESET#			
PROTO69	G14										69		DIPSW1					
PROTO70	F16										70		DIPSW3			AUDIO-SDTI		
PROTO71	F14										71	BARLED9		RAM-A16	IDE-IORDY			
PROTO72	F12										72							
PROTO73	B16										73							
PROTO74	E11										74							
PROTO75	D11										75							
PROTO76	E10										76							
PROTO77	B10										77		DIPSW4			AUDIO-SCLK		
PROTO78	A10										78	LED1-G		RAM-A14	IDE-D12			
PROTO79	D9										79	LED1-B		RAM-A12	IDE-D13			
PROTO80	B9										80	BARLED7		RAM-D0	IDE-D6			RS232-RD
PROTO81	A9										81	BARLED6		RAM-D1	IDE-D5			RS232-CT
PROTO82	A8										82	LED1-F		RAM-A7	IDE-CS0#			
PROTO83	C8										83	LED1-A		RAM-A6	IDE-CS1#			
PROTO84	D8										84	LED1-DP		RAM-A5	IDE-DA2			

B

XSA-200 Schematics

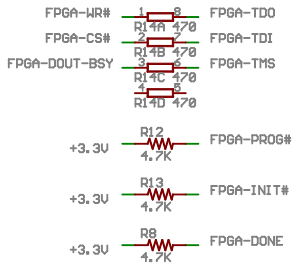
The following pages show the detailed schematics for the XSA-200 Board.

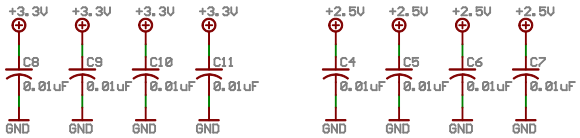
U1
XC2S200-5FG256C

GND	A1	GND@A1	SDRAM-D13	A2	I/O@A2	PROTO20	A3	IO_VREF_@@A3	PROTO14	A4	I/O@A4	PROTO10	A5	I/O@A5	PROTO5	A6	I/O@A6
SDRAM-D12	B1	IO_VREF_7@B1	GND	B2	GND@B2	PROTO21	B3	I/O@B3	PROTO18	B4	IO_VREF_@@B4	PROTO12	B5	I/O@B5	PROTO7	B6	I/O@B6
SDRAM-D1	C1	IO_VREF_7@C1	SDRAM-D0	C2	I/O@C2	+2.5V	C3	VCCINT@C3	FPGA-TCK	C4	TCK	SDRAM-D15	C5	I/O@C5	PROTO9	C6	IO_VREF_@@C6
SDRAM-D3	D1	IO_VREF_7@D1	SW3	D2	I/O@D2	FPGA-TMS	D3	TMS	+2.5V	D4	VCCINT@D4	SDRAM-D14	D5	I/O@D5	PROTO13	D6	I/O@D6
PS2-DATA	E1	I/O@E1	SDRAM-D10	E2	I/O@E2	SW2	E3	I/O@E3	SDRAM-D11	E4	I/O@E4	+2.5V	E5	VCCINT@E5	PROTO19	E6	I/O@E6
SDRAM-D6	F1	I/O@F1	SDRAM-D5	F2	I/O@F2	SDRAM-D4	F3	I/O@F3	PS2-CLK	F4	I/O@F4	SDRAM-D2	F5	I/O@F5	GND	F6	GND@F6
SDRAM-WE#	G1	IO_IRDY@G1	SDRAM-LDQM	G2	I/O@G2	SDRAM-D7	G3	I/O@G3	SDRAM-D8	G4	I/O@G4	SDRAM-D9	G5	I/O@G5	GND	G6	GND@G6
SDRAM-UDQM	H1	I/O@H1	VGA-GREEN0	H2	I/O@H2	SDRAM-CAS#	H3	IO_VREF_7@H3	VGA-BLUE0	H4	I/O@H4	+3.3V	H5	VCCO_7@H5	+3.3V	H6	VCCO_7@H6
VGA-RED0	J1	I/O@J1	SDRAM-RAS#	J2	IO_TRDY@J2	SDRAM-CS#	J3	IO_VREF_6@J3	SDRAM-CLK	J4	I/O@J4	+3.3V	J5	VCCO_6@J5	+3.3V	J6	VCCO_6@J6
VGA-VSYNC#	K1	I/O@K1	SDRAM-BA0	K2	I/O@K2	VGA-BLUE1	K3	I/O@K3	VGA-HSYNC#	K4	I/O@K4	VGA-GREEN1	K5	I/O@K5	GND	K6	GND@K6
SDRAM-CKE	L1	I/O@L1	SDRAM-BA1	L2	I/O@L2	SDRAM-A10	L3	I/O@L3	SDRAM-A0	L4	I/O@L4	VGA-BLUE2	L5	I/O@L5	GND	L6	GND@L6
VGA-RED1	M1	IO_VREF_6@M1	SDRAM-A12	M2	I/O@M2	SDRAM-A11	M3	I/O@M3	SDRAM-A3	M4	I/O@M4	+2.5V	M5	VCCINT@M5	SDRAM-A6	M6	I/O@M6
SDRAM-A1	N1	IO_VREF_6@N1	SDRAM-A2	N2	IO_VREF_6@N2	GND	N3	M0	+2.5V	N4	VCCINT@N4	SDRAM-A8	N5	I/O@N5	SDRAM-A5	N6	I/O@N6
SDRAM-A9	P1	I/O@P1	+3.3V	P2	M1_	+2.5V	P3	VCCINT@P3		P4	NC@P4	PROTO23	P5	I/O@P5	PROTO26	P6	I/O@P6
VGA-GREEN2	R1	I/O@R1	GND	R2	GND@R2	GND	R3	M2_		R4	NC@R4	PROTO25	R5	IO_VREF_5@R5	PROTO28	R6	I/O@R6
GND	T1	GND@T1	VGA-RED2	T2	IO_VREF_5@T2	SDRAM-A7	T3	I/O@T3	PROTO24	T4	IO_VREF_5@T4	SDRAM-A4	T5	I/O@T5	PROTO31	T6	I/O@T6

PROTO1	A7	I/O@A7	PROTO82	A8	I/O@A8	PROTO81	A9	I/O@A9	PROTO78	A10	I/O@A10	FLASH-A3	A11	I/O@A11	FLASH-A6	A12	I/O@A12
PROTO3	B7	IO_VREF_0@B7	CLKB	B8	GCK3	PROTO80	B9	IO_VREF_1@B9	PROTO77	B10	I/O@B10	FLASH-A4	B11	I/O@B11	FLASH-A7	B12	I/O@B12
PROTO4	C7	I/O@C7	PROTO83	C8	I/O@C8	CLKC	C9	GCK2	FLASH-A1	C10	I/O@C10	FLASH-A5	C11	IO_VREF_1@C11	FLASH-RDY	C12	I/O@C12
PROTO6	D7	I/O@D7	PROTO84	D8	I/O@D8	PROTO79	D9	I/O@D9	FLASH-A2	D10	I/O@D10	PROTO75	D11	I/O@D11	FLASH-WE#	D12	I/O@D12
PROTO8	E7	I/O@E7	+3.3V	E8	VCCO_0@E8	+3.3V	E9	VCCO_1@E9	PROTO76	E10	I/O@E10	PROTO74	E11	IO_VREF_1@E11	+2.5V	E12	VCCINT@E12
GND	F7	GND@F7	+3.3V	F8	VCCO_0@F8	+3.3V	F9	VCCO_1@F9	GND	F10	GND@F10	GND	F11	GND@F11	PROTO72	F12	I/O@F12
GND	G7	GND@G7	GND	G8	GND@G8	GND	G9	GND@G9	GND	G10	GND@G10	GND	G11	GND@G11	FLASH-A14	G12	I/O@G12
GND	H7	GND@H7	GND	H8	GND@H8	GND	H9	GND@H9	GND	H10	GND@H10	+3.3V	H11	VCCO_2@H11	+3.3V	H12	VCCO_2@H12
GND	J7	GND@J7	GND	J8	GND@J8	GND	J9	GND@J9	GND	J10	GND@J10	+3.3V	J11	VCCO_3@J11	+3.3V	J12	VCCO_3@J12
GND	K7	GND@K7	GND	K8	GND@K8	GND	K9	GND@K9	GND	K10	GND@K10	GND	K11	GND@K11	FLASH-D8	K12	I/O@K12
GND	L7	GND@L7	+3.3V	L8	VCCO_5@L8	+3.3V	L9	VCCO_4@L9	GND	L10	GND@L10	GND	L11	GND@L11	PROTO62	L12	I/O@L12
PROTO27	M7	I/O@M7	+3.3V	M8	VCCO_5@M8	+3.3V	M9	VCCO_4@M9	FLASH-A18	M10	I/O@M10	PROTO53	M11	I/O@M11	+2.5V	M12	VCCINT@M12
PROTO29	N7	I/O@N7	SDRAM-CLK	N8	GCK0	PROTO41	N9	I/O@N9	SW1-4	N10	I/O@N10	PROTO48	N11	I/O@N11	PROTO56	N12	I/O@N12
PROTO32	P7	I/O@P7	PROTO36	P8	IO_VREF_5@P8	PROTO40	P9	IO_VREF_4@P9	PROTO44	P10	I/O@P10	FLASH-A19	P11	I/O@P11	PROTO50	P12	I/O@P12
PROTO33	R7	I/O@R7	CLKA	R8	GCK1	PROTO39	R9	I/O@R9	PROTO43	R10	I/O@R10	SW1-3	R11	I/O@R11	PROTO47	R12	I/O@R12
PROTO35	T7	I/O@T7	PROTO37	T8	I/O@T8	PROTO38	T9	I/O@T9	PROTO42	T10	I/O@T10	PROTO45	T11	IO_VREF_4@T11	PROTO46	T12	IO_VREF_4@T12

FLASH-A17	A13	I/O@A13	FLASH-RESET#	A14	IO_VREF_1@A14	FPGA-TDI	A15	TDI	GND	A16	GND@A16						
FPGA-CS#	B13	IO_CS	FPGA-TDO	B14	TDO	GND	B15	GND@B15	PROTO73	B16	I/O@B16						
FPGA-WR#	C13	IO_WRITE	+2.5V	C14	VCCINT@C14	FPGA-DOUT-BSY	C15	IO_DOUT_BUSY	FLASH-A9	C16	I/O@C16						
+2.5V	D13	VCCINT@D13	FPGA-DIN-D0	D14	IO_DIN_D0	FPGA-CCLK	D15	CCLK	FLASH-A11	D16	I/O@D16						
FLASH-A8	E13	IO_VREF_2@E13	FLASH-A10	E14	I/O@E14	FLASH-A13	E15	I/O@E15	FPGA-D1	E16	IO_D1						
FLASH-A12	F13	IO_VREF_2@F13	PROTO71	F14	IO_VREF_2@F14	FPGA-D2	F15	IO_D2	PROTO70	F16	I/O@F16						
FLASH-A15	G13	I/O@G13	PROTO69	G14	I/O@G14	PROTO68	G15	I/O@G15	FPGA-D3	G16	IO_D3						
PROTO67	H13	IO_VREF_2@H13	PROTO66	H14	I/O@H14	PP-CS	H15	I/O@H15	FLASH-A16	H16	IO_IRDY@H16						
PP-S6	J13	I/O@J13	FLASH-D15	J14	I/O@J14	FLASH-BYTE#	J15	IO_TRDY@J15	FPGA-D4	J16	IO_D4						
FLASH-D10	K13	I/O@K13	FLASH-D12	K14	I/O@K14	FLASH-D13	K15	I/O@K15	FLASH-D14	K16	IO_VREF_3@K16						
FLASH-A0	L13	IO_VREF_3@L13	FLASH-OE#	L14	IO_VREF_3@L14	FLASH-D9	L15	I/O@L15	FLASH-D11	L16	I/O@L16						
PROTO60	M13	IO_VREF_3@M13	PROTO63	M14	I/O@M14	FLASH-CE#	M15	I/O@M15	FPGA-D5	M16	IO_D5						
+2.5V	N13	VCCINT@N13	FPGA-D7	N14	IO_D7	FPGA-INIT#	N15	IO_INIT	FPGA-D6	N16	IO_D6						
PROTO57	P13	I/O@P13	+2.5V	P14	VCCINT@P14	FPGA-PROG#	P15	PROGRAM	PROTO61	P16	I/O@P16						
PROTO51	R13	IO_VREF_4@R13	FPGA-DONE	R14	DONE	GND	R15	GND@R15	PROTO59	R16	I/O@R16						
PROTO49	T13	I/O@T13	PROTO34	T14	I/O@T14	PROTO58	T15	I/O@T15	GND	T16	GND@T16						



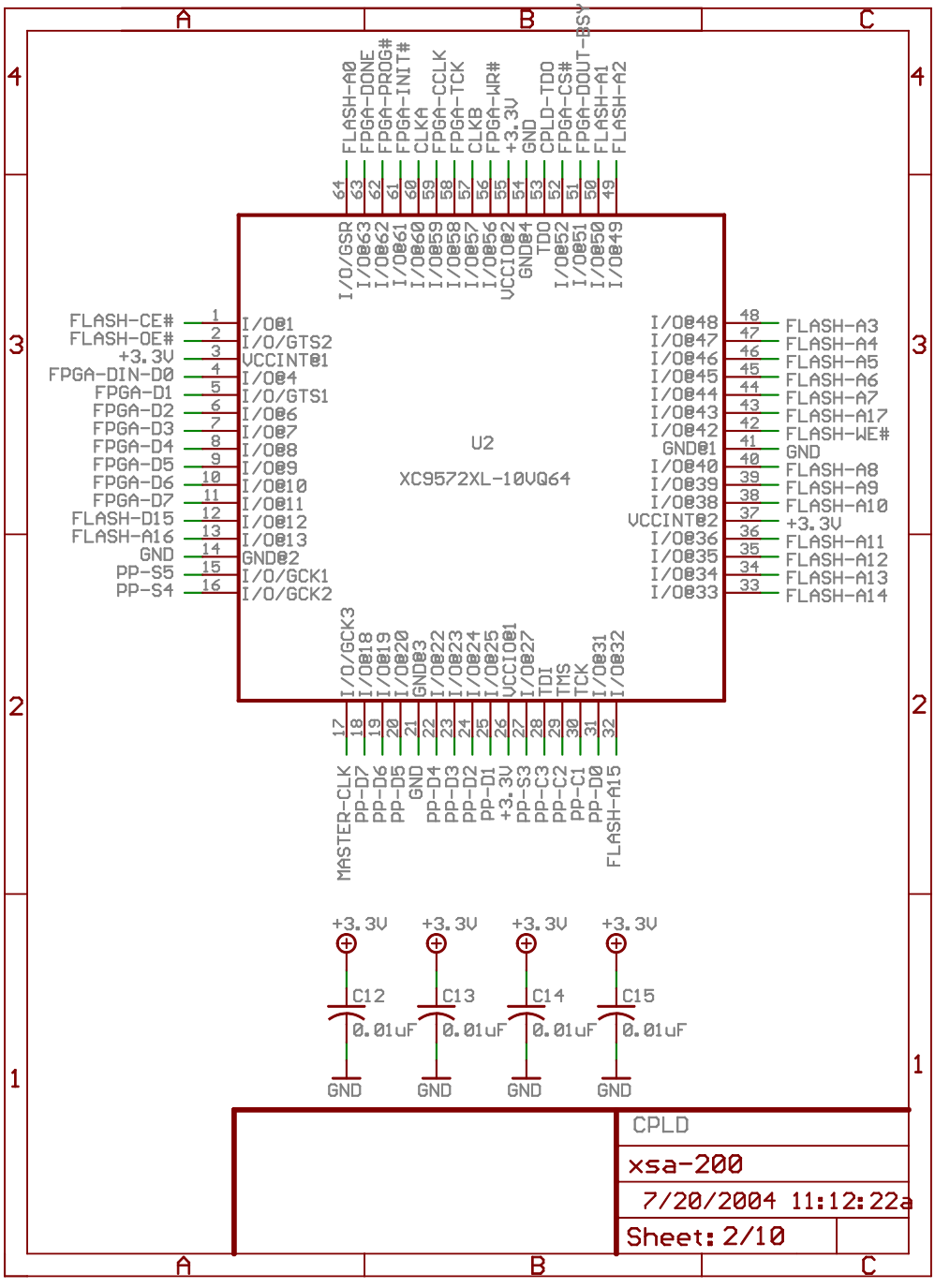


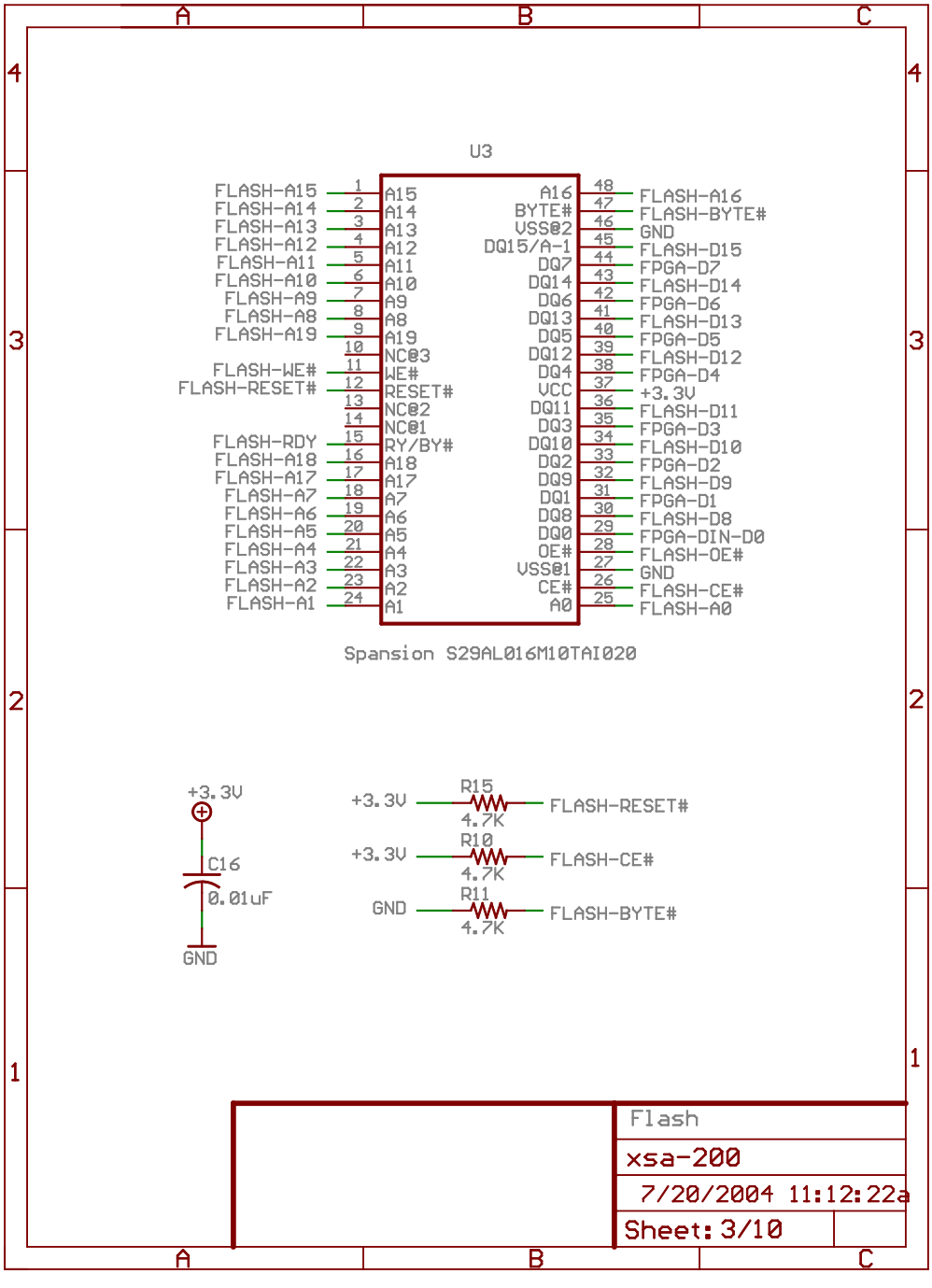
FPGA

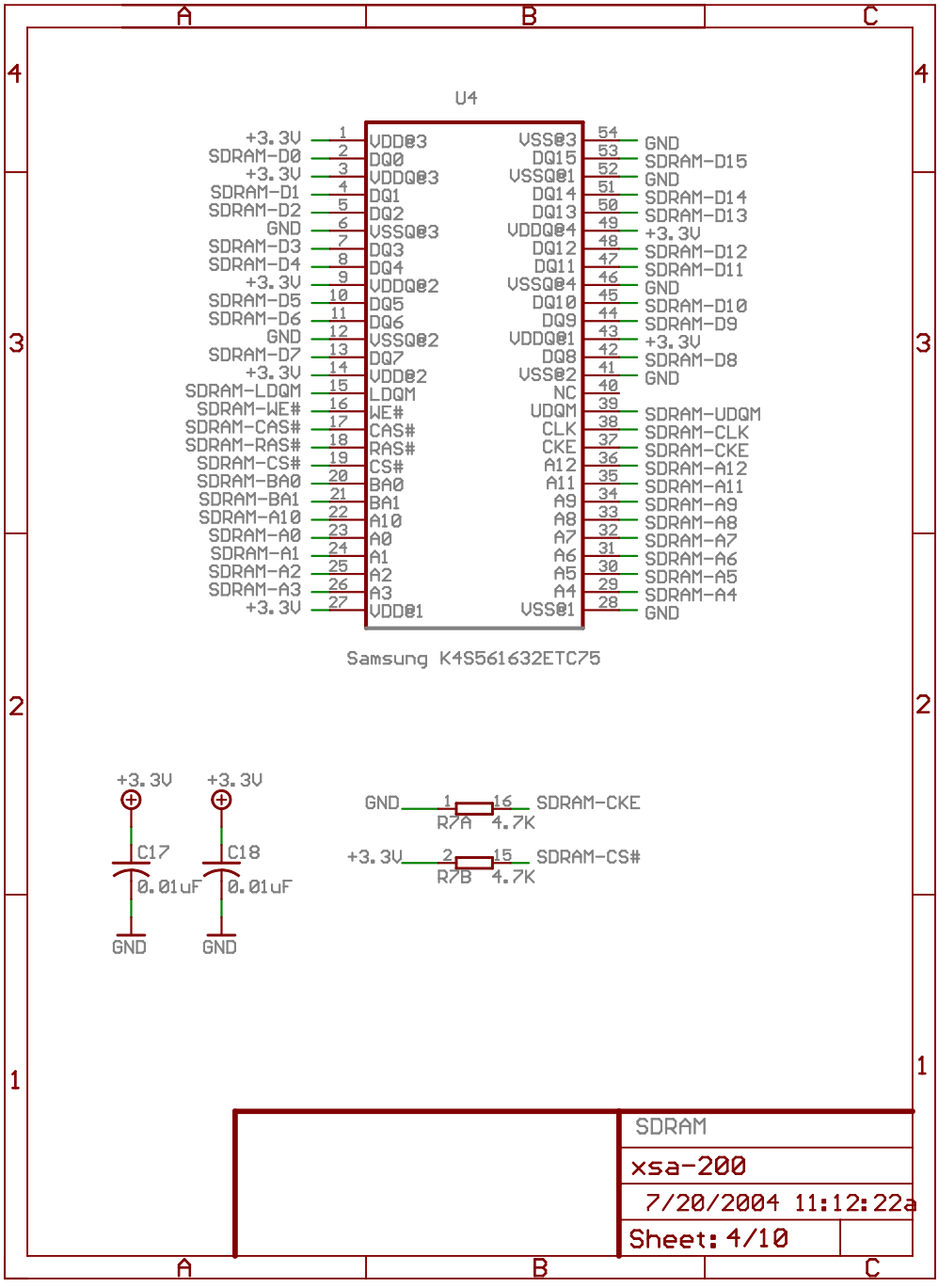
TITLE: xsa-200

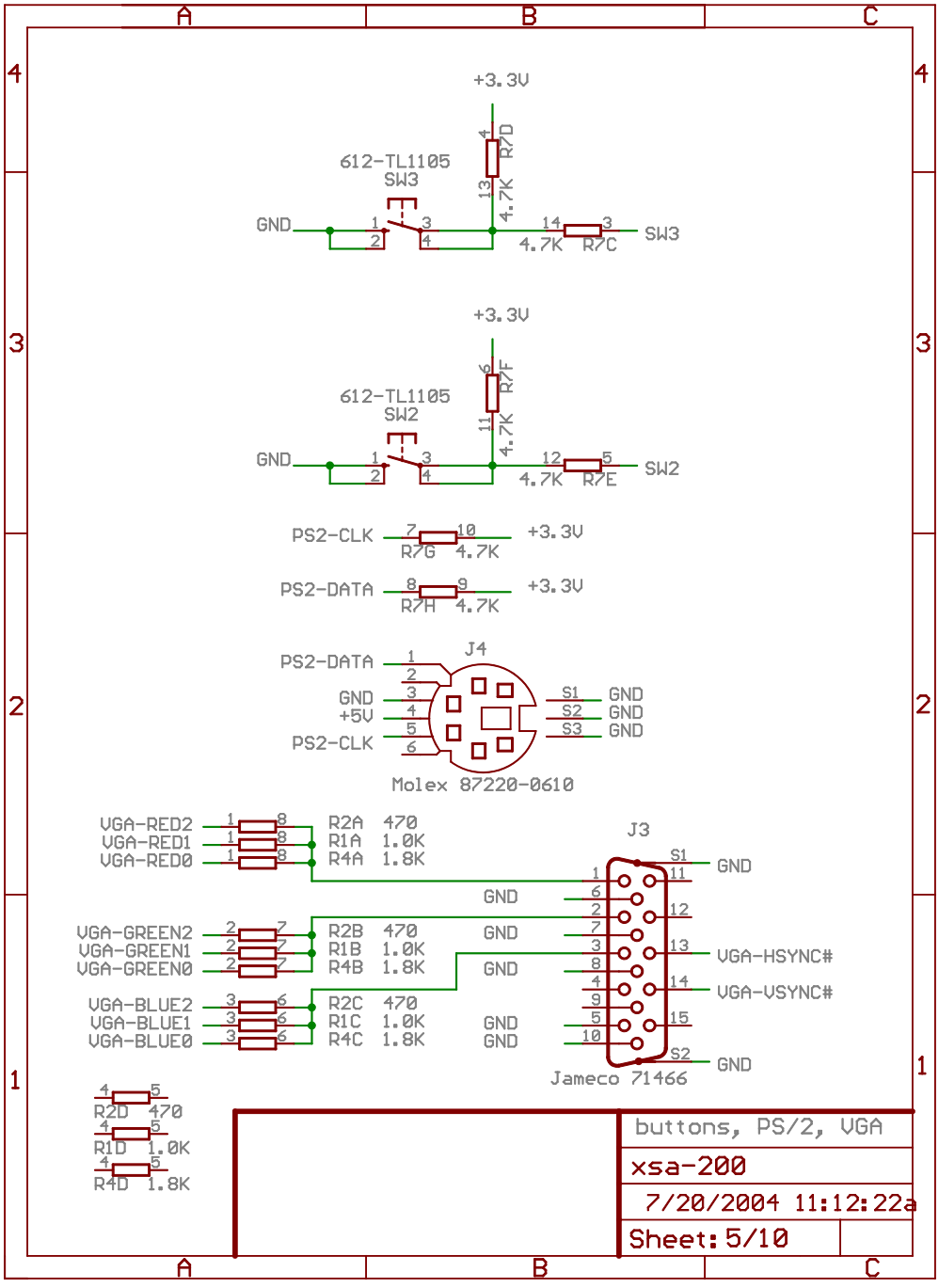
Document Number: REV:

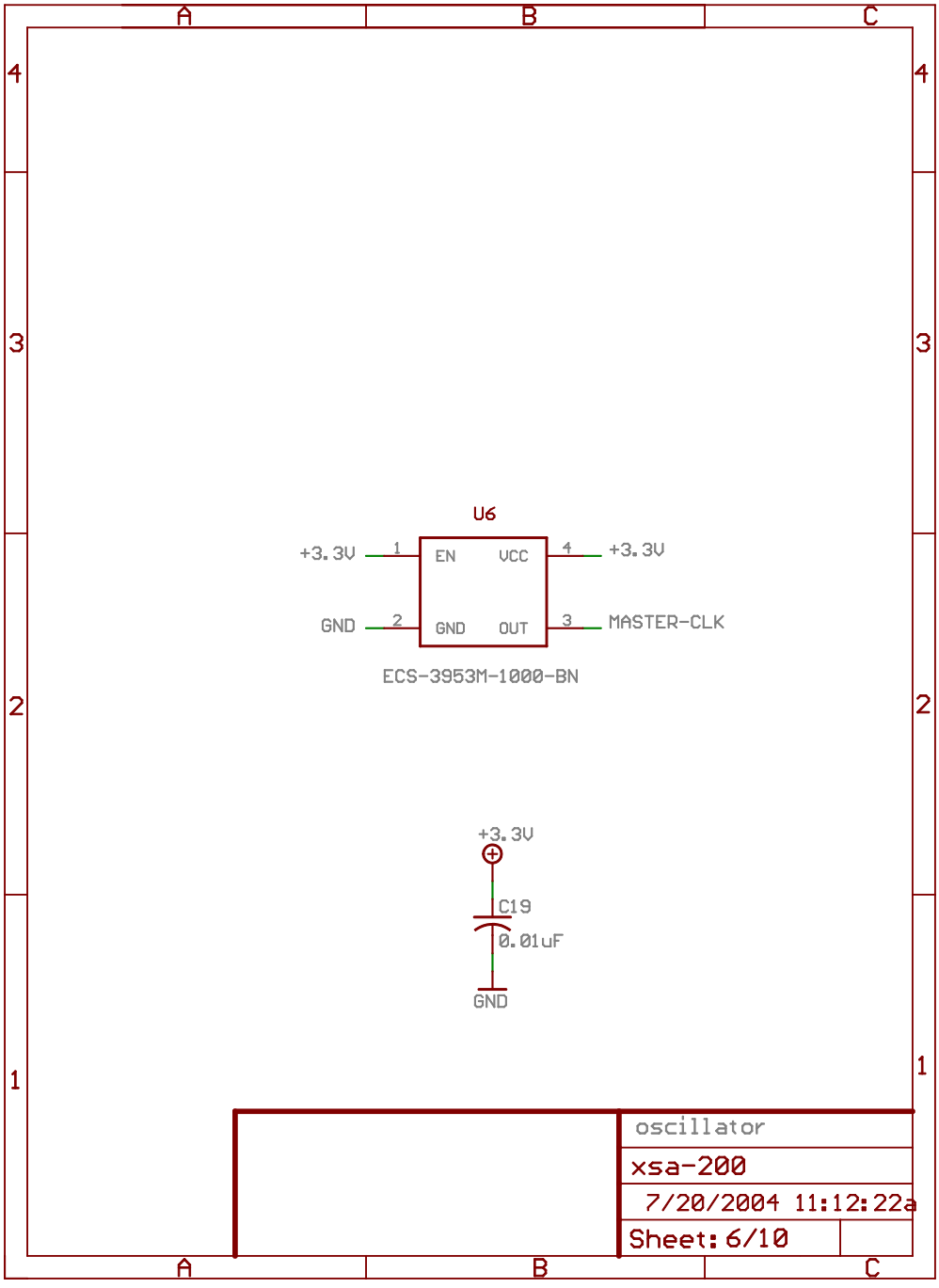
Date: 7/20/2004 11:12:22a Sheet: 1/10

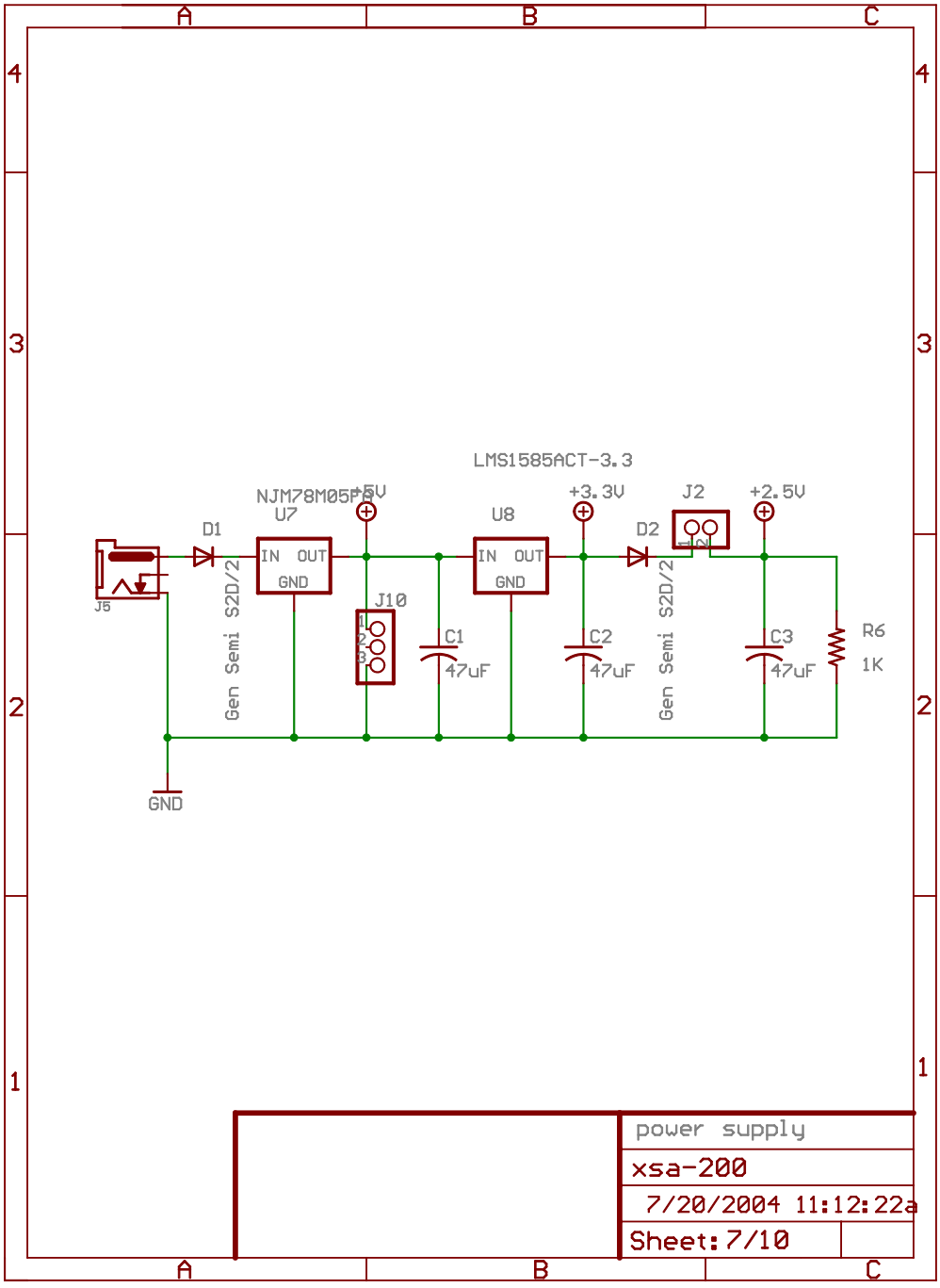

A
B
C
1
2
3
4
U2
XC9572XL-10VQ64
I/O/GSR
I/O@63
I/O@62
I/O@61
I/O@60
I/O@59
I/O@58
I/O@57
I/O@56
VCCIO@2
GND@4
TDO
I/O@52
I/O@51
I/O@50
I/O@49
I/O@48
I/O@47
I/O@46
I/O@45
I/O@44
I/O@43
I/O@42
GND@1
I/O@40
I/O@39
I/O@38
VCCINT@2
I/O@36
I/O@35
I/O@34
I/O@33
I/O@1
I/O/GTS2
VCCINT@1
I/O@4
I/O/GTS1
I/O@6
I/O@7
I/O@8
I/O@9
I/O@10
I/O@11
I/O@12
I/O@13
GND@2
I/O/GCK1
I/O/GCK2
I/O/GCK3
I/O@18
I/O@19
I/O@20
GND@3
I/O@22
I/O@23
I/O@24
I/O@25
VCCIO@1
I/O@27
TDI
TMS
TCK
I/O@31
I/O@32
FLASH-A0
FPGA-DONE
FPGA-PROG#
FPGA-INIT#
CLKA
FPGA-CCLK
FPGA-TCK
CLKB
FPGA-WR#
+3.3V
GND
CPLD-TDO
FPGA-CS#
FPGA-DOUT-BSY
FLASH-A1
FLASH-A2
FLASH-A3
FLASH-A4
FLASH-A5
FLASH-A6
FLASH-A7
FLASH-A17
FLASH-WE#
GND
FLASH-A8
FLASH-A9
FLASH-A10
+3.3V
FLASH-A11
FLASH-A12
FLASH-A13
FLASH-A14
FLASH-CE#
FLASH-OE#
+3.3V
FPGA-DIN-D0
FPGA-D1
FPGA-D2
FPGA-D3
FPGA-D4
FPGA-D5
FPGA-D6
FPGA-D7
FLASH-D15
FLASH-A16
GND
PP-S5
PP-S4
MASTER-CLK
PP-D7
PP-D6
PP-D5
GND
PP-D4
PP-D3
PP-D2
PP-D1
+3.3V
PP-S3
PP-C3
PP-C2
PP-C1
PP-D0
FLASH-A15
+3.3V
+3.3V
+3.3V
+3.3V
C12
C13
C14
C15
0.01uF
0.01uF
0.01uF
0.01uF
GND
GND
GND
GND
CPLD
xsa-200
7/20/2004 11:12:22a
Sheet: 2/10

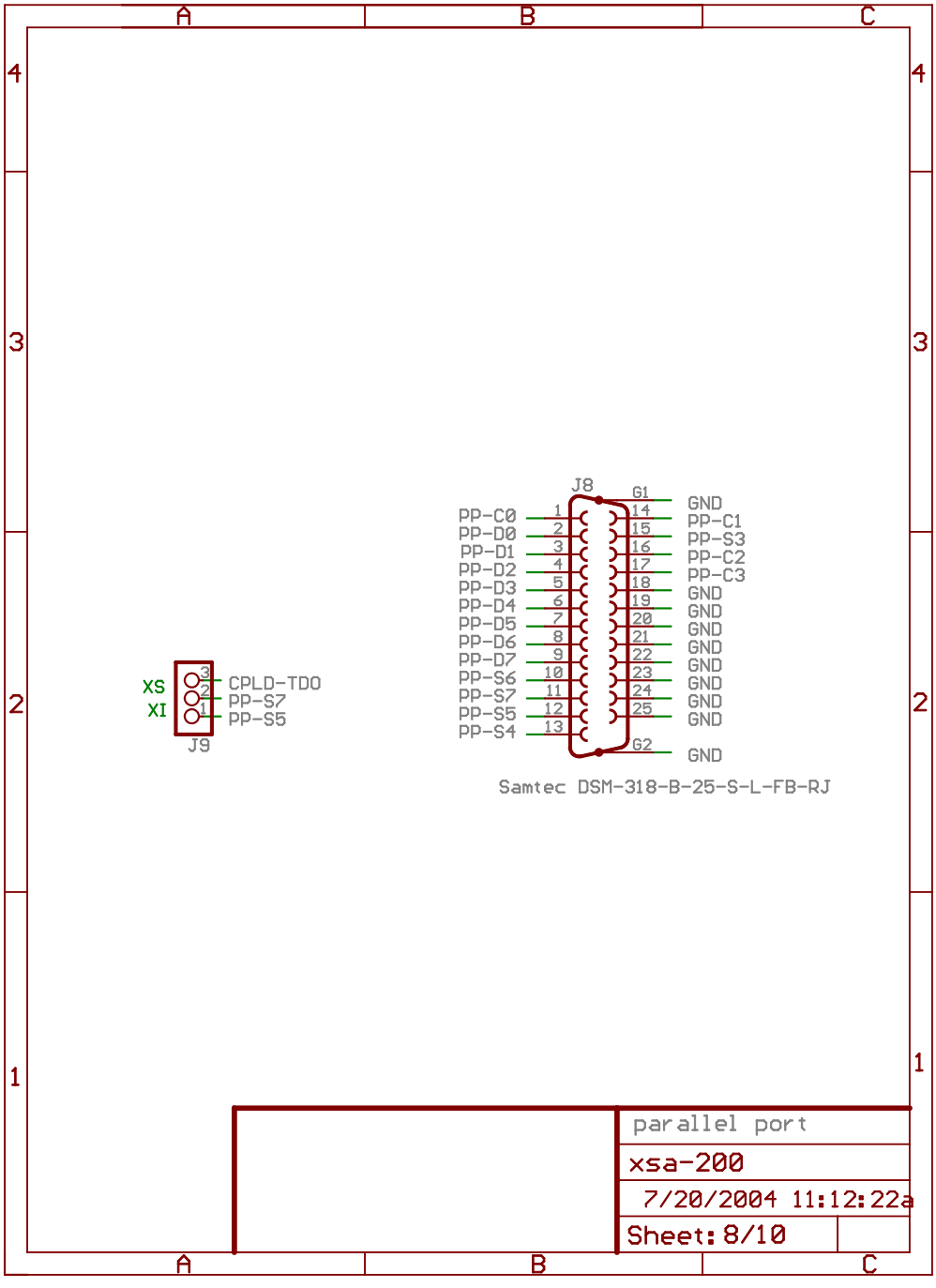

A
B
C
4
3
2
1
U3
FLASH-A15 1 A15
FLASH-A14 2 A14
FLASH-A13 3 A13
FLASH-A12 4 A12
FLASH-A11 5 A11
FLASH-A10 6 A10
FLASH-A9 7 A9
FLASH-A8 8 A8
FLASH-A19 9 A19
10 NC@3
FLASH-WE# 11 WE#
FLASH-RESET# 12 RESET#
13 NC@2
14 NC@1
FLASH-RDY 15 RY/BY#
FLASH-A18 16 A18
FLASH-A17 17 A17
FLASH-A7 18 A7
FLASH-A6 19 A6
FLASH-A5 20 A5
FLASH-A4 21 A4
FLASH-A3 22 A3
FLASH-A2 23 A2
FLASH-A1 24 A1
A16 48 FLASH-A16
BYTE# 47 FLASH-BYTE#
VSS@2 46 GND
DQ15/A-1 45 FLASH-D15
DQ7 44 FPGA-D7
DQ14 43 FLASH-D14
DQ6 42 FPGA-D6
DQ13 41 FLASH-D13
DQ5 40 FPGA-D5
DQ12 39 FLASH-D12
DQ4 38 FPGA-D4
VCC 37 +3.3V
DQ11 36 FLASH-D11
DQ3 35 FPGA-D3
DQ10 34 FLASH-D10
DQ2 33 FPGA-D2
DQ9 32 FLASH-D9
DQ1 31 FPGA-D1
DQ8 30 FLASH-D8
DQ0 29 FPGA-DIN-D0
OE# 28 FLASH-OE#
VSS@1 27 GND
CE# 26 FLASH-CE#
A0 25 FLASH-A0
Spansion S29AL016M10TAI020
+3.3V
C16
0.01uF
GND
+3.3V
R15
4.7K
FLASH-RESET#
+3.3V
R10
4.7K
FLASH-CE#
GND
R11
4.7K
FLASH-BYTE#
Flash
xsa-200
7/20/2004 11:12:22a
Sheet: 3/10

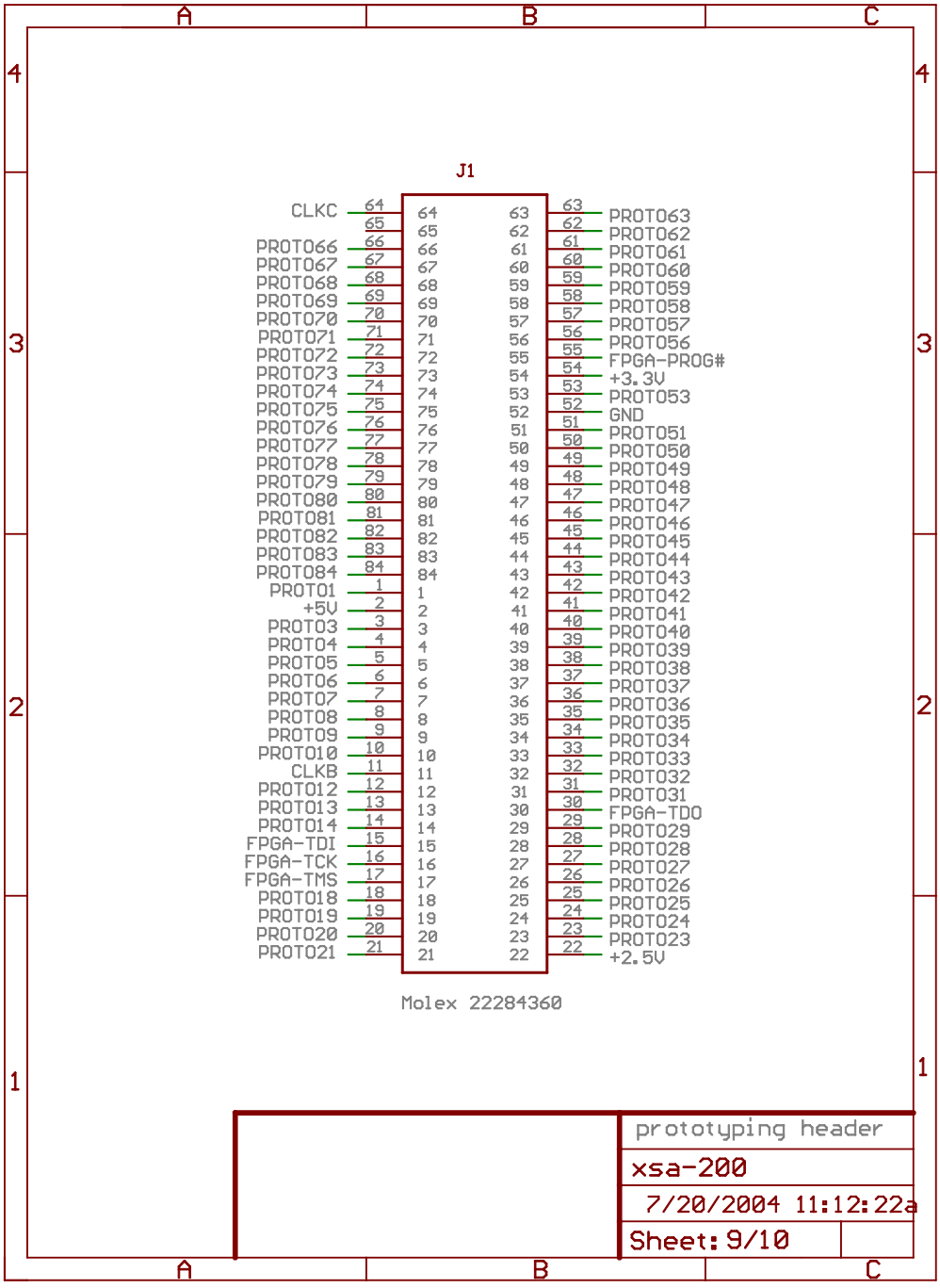

A
B
C
4
3
2
1
U4
+3.3V 1 VDD@3
SDRAM-D0 2 DQ0
+3.3V 3 VDDQ@3
SDRAM-D1 4 DQ1
SDRAM-D2 5 DQ2
GND 6 VSSQ@3
SDRAM-D3 7 DQ3
SDRAM-D4 8 DQ4
+3.3V 9 VDDQ@2
SDRAM-D5 10 DQ5
SDRAM-D6 11 DQ6
GND 12 VSSQ@2
SDRAM-D7 13 DQ7
+3.3V 14 VDD@2
SDRAM-LDQM 15 LDQM
SDRAM-WE# 16 WE#
SDRAM-CAS# 17 CAS#
SDRAM-RAS# 18 RAS#
SDRAM-CS# 19 CS#
SDRAM-BA0 20 BA0
SDRAM-BA1 21 BA1
SDRAM-A10 22 A10
SDRAM-A0 23 A0
SDRAM-A1 24 A1
SDRAM-A2 25 A2
SDRAM-A3 26 A3
+3.3V 27 VDD@1
VSS@3 54 GND
DQ15 53 SDRAM-D15
VSSQ@1 52 GND
DQ14 51 SDRAM-D14
DQ13 50 SDRAM-D13
VDDQ@4 49 +3.3V
DQ12 48 SDRAM-D12
DQ11 47 SDRAM-D11
VSSQ@4 46 GND
DQ10 45 SDRAM-D10
DQ9 44 SDRAM-D9
VDDQ@1 43 +3.3V
DQ8 42 SDRAM-D8
VSS@2 41 GND
NC 40
UDQM 39 SDRAM-UDQM
CLK 38 SDRAM-CLK
CKE 37 SDRAM-CKE
A12 36 SDRAM-A12
A11 35 SDRAM-A11
A9 34 SDRAM-A9
A8 33 SDRAM-A8
A7 32 SDRAM-A7
A6 31 SDRAM-A6
A5 30 SDRAM-A5
A4 29 SDRAM-A4
VSS@1 28 GND
Samsung K4S561632ETC75
+3.3V
+3.3V
C17
0.01uF
GND
C18
0.01uF
GND
GND 1 R7A 16 SDRAM-CKE
4.7K
+3.3V 2 R7B 15 SDRAM-CS#
4.7K
SDRAM
xsa-200
7/20/2004 11:12:22a
Sheet: 4/10

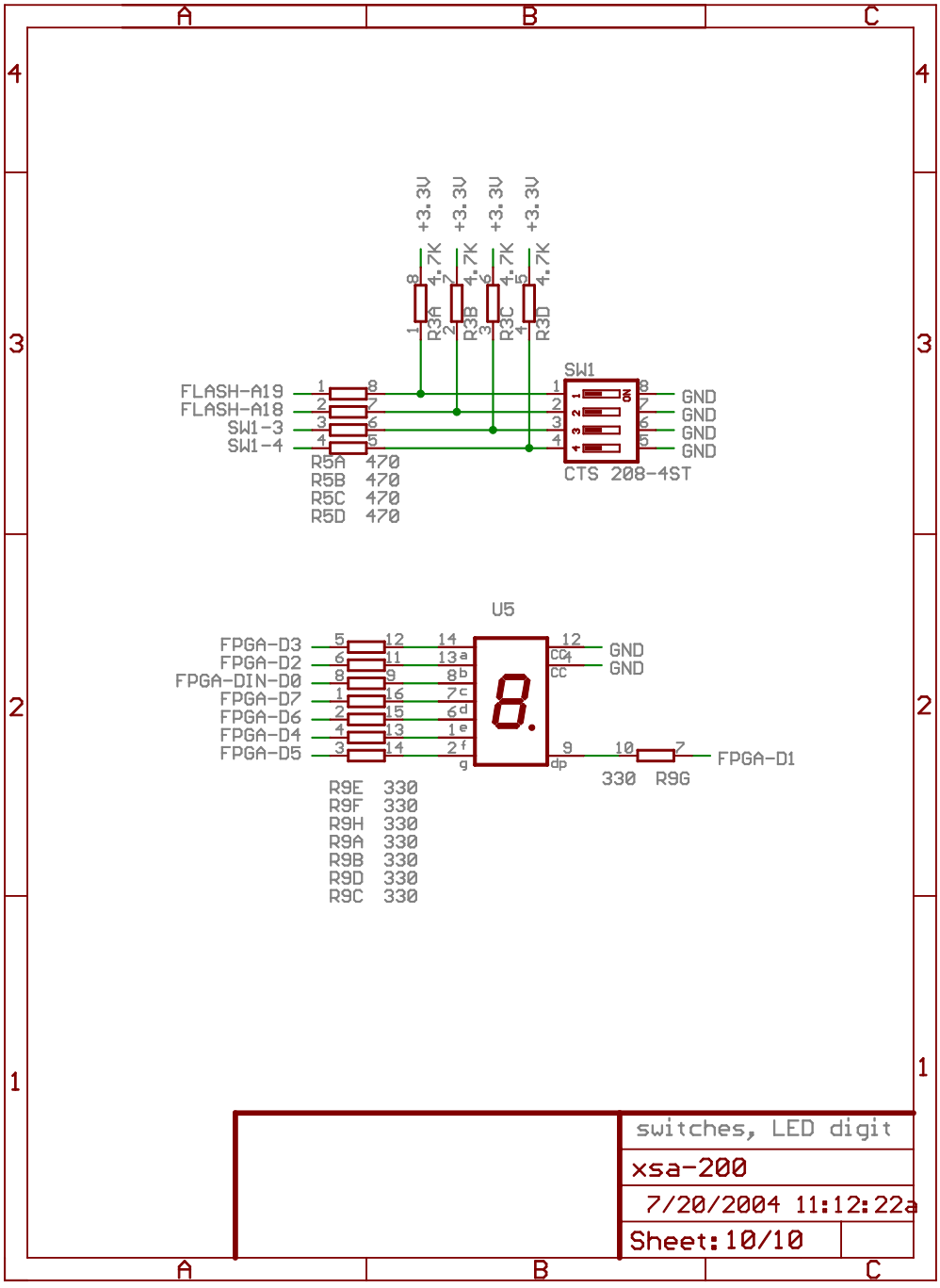

A
B
C
4
3
2
1
+3.3V
612-TL1105
SW3
R7D
4.7K
GND
1
2
3
4
14
4.7K
R7C
SW3
+3.3V
612-TL1105
SW2
R7F
4.7K
GND
12
4.7K
R7E
SW2
PS2-CLK
7
10
R7G
4.7K
+3.3V
PS2-DATA
8
9
R7H
4.7K
+3.3V
J4
PS2-DATA
GND
+5V
PS2-CLK
S1
S2
S3
GND
GND
GND
Molex 87220-0610
VGA-RED2
VGA-RED1
VGA-RED0
R2A 470
R1A 1.0K
R4A 1.8K
VGA-GREEN2
VGA-GREEN1
VGA-GREEN0
R2B 470
R1B 1.0K
R4B 1.8K
VGA-BLUE2
VGA-BLUE1
VGA-BLUE0
R2C 470
R1C 1.0K
R4C 1.8K
R2D 470
R1D 1.0K
R4D 1.8K
J3
GND
GND
GND
GND
GND
VGA-HSYNC#
VGA-VSYNC#
GND
Jameco 71466
buttons, PS/2, VGA
xsa-200
7/20/2004 11:12:22a
Sheet: 5/10


A
B
C
4
3
2
1
U6
+3.3V
1
EN
VCC
4
+3.3V
GND
2
GND
OUT
3
MASTER-CLK
ECS-3953M-1000-BN
+3.3V
C19
0.01uF
GND
oscillator
xsa-200
7/20/2004 11:12:22a
Sheet: 6/10


A
B
C
4
3
2
1
LMS1585ACT-3.3
NJM78M05FA
U7
U8
D1
D2
J2
J5
J10
IN OUT
GND
+5V
+3.3V
+2.5V
Gen Semi S2D/2
C1
47uF
C2
47uF
C3
47uF
R6
1K
GND
power supply
xsa-200
7/20/2004 11:12:22a
Sheet:7/10


A
B
C
4
3
2
1
XS
XI
CPLD-TDO
PP-S7
PP-S5
J9
J8
PP-C0
PP-D0
PP-D1
PP-D2
PP-D3
PP-D4
PP-D5
PP-D6
PP-D7
PP-S6
PP-S7
PP-S5
PP-S4
GND
PP-C1
PP-S3
PP-C2
PP-C3
Samtec DSM-318-B-25-S-L-FB-RJ
parallel port
xsa-200
7/20/2004 11:12:22a
Sheet: 8/10


A
B
C
4
3
2
1
J1
CLKC 64 64 63 63 PROTO63
65 65 62 62 PROTO62
PROTO66 66 66 61 61 PROTO61
PROTO67 67 67 60 60 PROTO60
PROTO68 68 68 59 59 PROTO59
PROTO69 69 69 58 58 PROTO58
PROTO70 70 70 57 57 PROTO57
PROTO71 71 71 56 56 PROTO56
PROTO72 72 72 55 55 FPGA-PROG#
PROTO73 73 73 54 54 +3.3V
PROTO74 74 74 53 53 PROTO53
PROTO75 75 75 52 52 GND
PROTO76 76 76 51 51 PROTO51
PROTO77 77 77 50 50 PROTO50
PROTO78 78 78 49 49 PROTO49
PROTO79 79 79 48 48 PROTO48
PROTO80 80 80 47 47 PROTO47
PROTO81 81 81 46 46 PROTO46
PROTO82 82 82 45 45 PROTO45
PROTO83 83 83 44 44 PROTO44
PROTO84 84 84 43 43 PROTO43
PROTO1 1 1 42 42 PROTO42
+5V 2 2 41 41 PROTO41
PROTO3 3 3 40 40 PROTO40
PROTO4 4 4 39 39 PROTO39
PROTO5 5 5 38 38 PROTO38
PROTO6 6 6 37 37 PROTO37
PROTO7 7 7 36 36 PROTO36
PROTO8 8 8 35 35 PROTO35
PROTO9 9 9 34 34 PROTO34
PROTO10 10 10 33 33 PROTO33
CLKB 11 11 32 32 PROTO32
PROTO12 12 12 31 31 PROTO31
PROTO13 13 13 30 30 FPGA-TDO
PROTO14 14 14 29 29 PROTO29
FPGA-TDI 15 15 28 28 PROTO28
FPGA-TCK 16 16 27 27 PROTO27
FPGA-TMS 17 17 26 26 PROTO26
PROTO18 18 18 25 25 PROTO25
PROTO19 19 19 24 24 PROTO24
PROTO20 20 20 23 23 PROTO23
PROTO21 21 21 22 22 +2.5V
Molex 22284360
prototyping header
xsa-200
7/20/2004 11:12:22a
Sheet: 9/10


A
B
C
4
3
2
1
+3.3V
+3.3V
+3.3V
+3.3V
4.7K
4.7K
4.7K
4.7K
R3A
R3B
R3C
R3D
SW1
FLASH-A19
FLASH-A18
SW1-3
SW1-4
GND
GND
GND
GND
CTS 208-4ST
R5A 470
R5B 470
R5C 470
R5D 470
U5
FPGA-D3
FPGA-D2
FPGA-DIN-D0
FPGA-D7
FPGA-D6
FPGA-D4
FPGA-D5
GND
GND
FPGA-D1
330 R9G
R9E 330
R9F 330
R9H 330
R9A 330
R9B 330
R9D 330
R9C 330
switches, LED digit
xsa-200
7/20/2004 11:12:22a
Sheet:10/10