



1. Kit Contents

The C8051F12x Development Kit contains the following items:

- C8051F12x Target Board
- Serial Adapter (RS-232 to Target Board Protocol Converter)
- Cygnal IDE and Product Information CD-ROM. CD content includes:
 - Cygnal Integrated Development Environment (IDE)
 - Keil 8051 Development Tools (macro assembler, linker, evaluation 'C' compiler)
 - Installation Utility (SETUP.EXE)
 - Source code examples and register definition file
 - Documentation
- AC to DC Power Adapter
- RS232 Serial Cable
- 7" Ribbon Cable
- Quick-start Guide
- C8051F12x Development Kit User's Guide (this document)

2. Hardware Setup

The target board is connected to a PC running the Cygnal IDE via the Serial Adapter as shown in Figure 1.

1. Connect one end of the RS232 serial cable to a Serial (COM) Port on the PC.
2. Connect the other end of the RS232 serial cable to the DB-9 connector on the Serial Adapter.
3. Connect the Serial Adapter to the JTAG connector on the target board with the 10-pin ribbon cable.
4. Connect the AC/DC power adapter to power jack P1 on the target board.

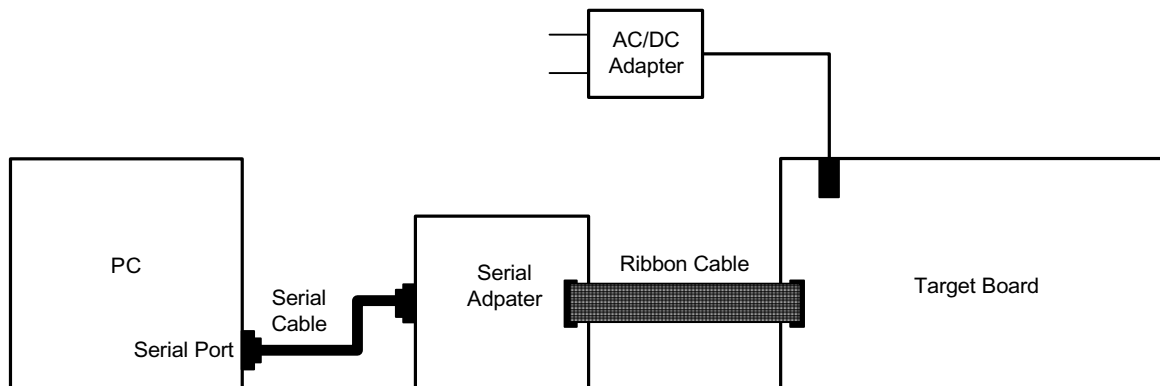


Figure 1. Hardware Setup

Note: The Reset switch on the target board is disabled when the serial adapter is connected to the target board. Use the **Reset** button in the Cygnal IDE toolbar to reset the target when connected to the Serial Adapter.



3. Software Setup

The included CD-ROM contains the Cygnal Integrated Development Environment (IDE), Keil software 8051 tools and additional documentation. Insert the CD-ROM into your PC's CD-ROM drive. An installer will automatically launch, allowing you to install the IDE software or read documentation by clicking buttons on the Installation Panel. If the installer does not automatically start when you insert the CD-ROM, run "autorun.exe" found in the root directory of the CD-ROM. Refer to the "readme.txt" file on the CD-ROM for the latest information regarding known IDE problems and restrictions.

4. CYGNAL Integrated Development Environment

The Cygnal IDE integrates a source-code editor, source-level debugger and in-system Flash programmer. The use of third-party compilers and assemblers is also supported. This development kit includes the Keil Software A51 macro assembler, BL51 linker and evaluation version C51 'C' compiler. These tools can be used from within the Cygnal IDE.

4.1. System Requirements

The Cygnal IDE requirements:

- Pentium-class host PC running Microsoft Windows 95 or later, or Microsoft Windows NT or later.
- One available COM port (1-4).
- 64 MB RAM and 40MB free HD space recommended.

4.2. Assembler and Linker

A full-version Keil A51 macro assembler and BL51 banking linker are included with the development kit and are installed during IDE installation. The complete assembler and linker reference manual can be found on-line under the **Help** menu in the IDE or in the "Cygnal/hlp" directory (A51.pdf).

4.3. Evaluation C51 'C' Compiler

An evaluation version of the Keil C51 'C' compiler is included with the development kit and is installed during IDE installation. The evaluation version of the C51 compiler is the same as the full professional version except code size is limited to 4K bytes and the floating point library is not included. The C51 compiler reference manual can be found under the **Help** menu in the IDE or in the "Cygnal/hlp" directory (C51.pdf).

4.4. Using the Keil Software 8051 Tools with the Cygnal IDE

To perform source-level debugging with the IDE, you must configure the Keil 8051 tools to generate an absolute object file in the OMF-51 format with object extensions and debug records enabled. You may build the OMF-51 absolute object file by calling the Keil 8051 tools at the command line (e.g. batch file or make file) or by using the project manager built into the IDE. The default configuration when using the Cygnal IDE project manager enables object extension and debug record generation. Refer to **Applications Note AN004** in the "Documentation" directory on the CD-ROM for additional information on using the Keil 8051 tools with the Cygnal IDE.

To build an absolute object file using the Cygnal IDE project manager, you must first create a project. A project consists of a set of files, IDE configuration, debug views, and a target build configuration (list of files and tool configurations used as input to the assembler, compiler, and linker when building an output object file).



The following sections illustrate the steps necessary to manually create a project with one or more source files, build a program and download the program to the target in preparation for debugging. (The IDE will automatically create a single-file project using the currently open and active source file if you select **Build/Make Project** before a project is defined.)

4.4.1. Creating a New Project

1. Select **Project->New Project** to open a new project and reset all configuration settings to default.
2. Select **File->New File** to open an editor window. Create your source file(s) and save the file(s) with a recognized extension, such as .c, .h, or .asm, to enable color syntax highlighting.
3. Right-click on "New Project" in the **Project Window**. Select **Add files to project**. Select files in the file browser and click Open. Continue adding files until all project files have been added.
4. For each of the files in the **Project Window** that you want assembled, compiled and linked into the target build, right-click on the file name and select **Add file to build**. Each file will be assembled or compiled as appropriate (based on file extension) and linked into the build of the absolute object file.

Note: If a project contains a large number of files, the "Group" feature of the IDE can be used to organize. Right-click on "New Project" in the **Project Window**. Select **Add Groups to project**. Add pre-defined groups or add customized groups. Right-click on the group name and choose **Add file to group**. Select files to be added. Continue adding files until all project files have been added.

4.4.2. Building and Downloading the Program for Debugging

1. Once all source files have been added to the target build, build the project by clicking on the **Build/Make Project** button in the toolbar or selecting **Project->Build/Make Project** from the menu.
2. C8051F12x family devices use the JTAG debug interface. You must select **JTAG** in the **Options->Debug Interface** menu to enable connection to 'F12x target devices. Click the **Connect** button in the toolbar or select **Debug->Connect** from the menu to connect to the device.
3. Download the project to the target by clicking the **Download Code** button in the toolbar.

Note: To enable automatic downloading if the program build is successful select **Enable automatic connect/download after build** in the **Project->Target Build Configuration** dialog. If errors occur during the build process, the IDE will not attempt the download.

4. Save the project when finished with the debug session to preserve the current target build configuration, editor settings and the location of all open debug views. To save the project, select **Project->Save Project As...** from the menu. Create a new name for the project and click on **Save**.

4.5. Example Source Code

Example source code and register-definition files are installed into the "*Examples/C8051F31x*" directory during IDE installation. The source files may be used as a template for C8051F12x code development. The *C8051F120.inc* and *C8051F120.h* files define the addresses for all C8051F12x SFR registers and bit-addressable control/status bits. The register and bit names are identical to those used in the C8051F12x datasheet. Both register definition files are installed in the default search path used by the Keil Software 8051 tools. Therefore, when using the Keil 8051 tools included with the development kit (A51, C51), it is not necessary to copy a register definition file to each project's file directory.

5. Target Board

The C8051F12x Development Kit includes a target board with a C8051F12x device pre-installed for evaluation and preliminary software development. Numerous input/output (I/O) connections are provided to facilitate prototyping using the target board. Refer to Figure 2 for the locations of the various I/O connectors.

- P1 Power connector (accepts input from 7 to 15 VDC unregulated power adapter)
- J1 Connects SW2 to C8051F02x Port 1.7 pin
- J3 Connects LED D3 to C8051F02x Port 1.7 pin
- J4 Target board JTAG interface connector
- J5 DB9 connector for UART0 RS232 interface
- J6 Jumper to connect UART0 TX to DB9
- J9 Jumper to connect UART0 RX to DB9
- J11 Analog loopback connector
- J12-J19 Port 0 - 7 connectors
- J20 Analog I/O terminal block
- J22 VREF connector
- J23 VDD Monitor Disable
- J24 96-pin Expansion I/O connector
- JTAG Used to connect the Serial Adapter to the target board via a 10-pin ribbon cable

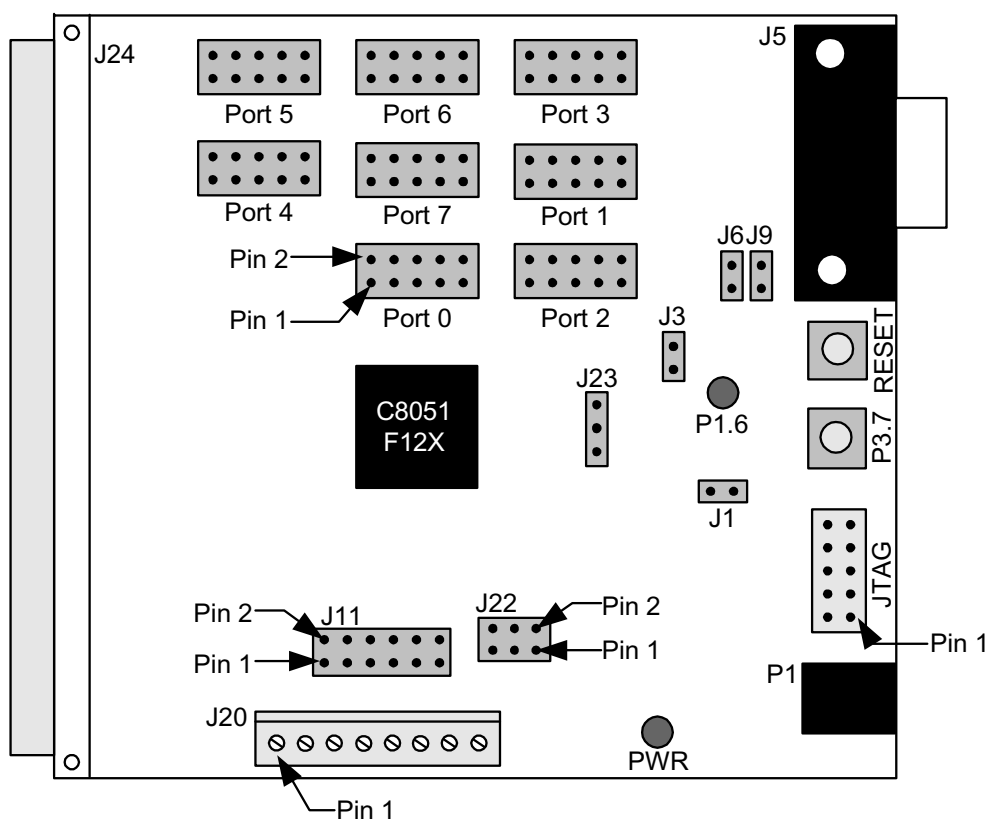


Figure 2. C8051F12x Target Board



5.1. System Clock Sources

The C8051F12x device installed on the target board features a calibrated programmable internal oscillator which is enabled as the system clock source on reset. After reset, the internal oscillator operates at a frequency of 3.0625MHz (+/-2%) by default but may be configured by software to operate at other frequencies. Therefore, in many applications an external oscillator is not required. However, if you wish to operate the C8051F31x device at a frequency not available with the internal oscillator, an external crystal may be used.

The target board is designed to facilitate the installation of an external crystal. Remove jumpers J9 and J10 and install the crystal at the pads marked Y1. Install a 10M Ω resistor at R9 and install capacitors at C14 and C15 using values appropriate for the crystal you select. Refer to the C8051F12x datasheet for more information on the configuration of the programmable internal oscillator and the use of external oscillators.

5.2. Switches and LEDs

Two switches and two LEDs are provided on the target board. Switch SW1 is connected to the RESET pin of the C8051F12x. Pressing SW1 puts the C8051F12x in its hardware-reset state. Switch SW2 is connected to the C8051F12x's Port 3.7 (P3.7) general purpose I/O (GPIO) pin. Pressing SW2 generates a logic high signal on the P3.7 pin. The P3.7 signal is also routed to a pin on the J24 I/O connector. Remove the shorting block from J1 to disconnect SW2 from P3.7.

The LED labeled PWR is used to indicate a power connection to the target board. The LED labeled P1.6 is connected to the C8051F12x's Port 1.6 (P1.6) GPIO pin through jumper J3. The P1.6 signal is also routed to a pin on the J24 I/O connector. Remove the shorting block from J3 to disconnect the LED from P1.6.

5.3. PORT I/O Connectors (J12 - J19)

In addition to all port I/O signals being routed to the 96-pin expansion connector, each of the eight parallel ports of the C8051F12x has its own 10-pin header connector. Each connector provides a pin for the corresponding port pins 0-7, +3.3VDC and digital ground. Table 1 defines the pins for the port connectors. The same pin-out order is used for all of the port connectors.

Pin #	Description
1	Pn.0
2	Pn.1
3	Pn.2
4	Pn.3
5	Pn.4
6	Pn.5
7	Pn.6
8	Pn.7
9	+3VD (+3.3VDC)
10	GND (Ground)

Table 1. J12 - J19 Port Connector Pin Descriptions



5.4. Target Board DEBUG Interface (J4)

The JTAG connector (J4) provides access to the JTAG pins of the C8051F12x. It is used to connect the Serial Adapter to the target board for in-circuit debugging and Flash programming. Table 2 shows the J4 pin definitions.

Pin #	Description
1	+3VD (+3.3VDC)
2, 3, 9	GND (Ground)
4	TCK
5	TMS
6	TDO
7	TDI
8, 10	Not Connected

Table 2. JTAG Connector Pin Descriptions

5.5. Serial Interface (J5)

A RS232 transceiver circuit and DB-9 (J5) connector are provided on the target board to facilitate serial connections to UART0 of the C8051F12x. The TX and RX signals of UART0 may be connected to the DB-9 connector and transceiver by installing jumpers on J6 and J9.

- J6 - Install shorting block to connect UART0 TX (P0.0) to transceiver.
- J9 - Install shorting block to connect UART0 RX (P0.1) to transceiver.

5.6. VREF Connector (J22)

The VREF jumper block J22 can be used to connect the VREF (Voltage Reference) output of the C8051F12x to any (or all) of its voltage reference inputs.

Install shorting blocks on J22 pins in the following manner:

- 1-2 to connect VREF to VREFD
- 3-4 to connect VREF to VREF0
- 5-6 to connect VREF to VREF1

5.7. VDD Monitor Disable Jumper (J23)

The VDD Monitor of the C8051F12x may be disabled by moving the shorting block on J23 from pins 1-2 to pins 2-3, as shown in Figure 3.

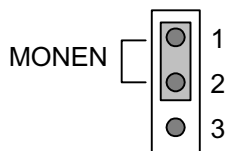


Figure 3. Hardware Setup



5.8. Analog I/O (J11 and J20)

Several C8051F12x analog signals are routed to the J20 terminal block and the J11 jumper block. Jumper block J11 provides the ability to connect DAC0 and DAC1 outputs to several different analog inputs by installing a shorting block between a DAC output and an analog input on adjacent pins of J11. Refer to Table 3 for J20 terminal block connections and Table 4 for J11 pin definitions.

Pin #	Description
1	CP0+
2	CP0-
3	DAC0
4	DAC1
5	AIN0.0
6	AIN0.1
7	VREF0
8	ADND (Analog Ground)

Table 3. J20 Terminal Block Pin Descriptions

Pin #	Description
1	CP0+
2	CP0-
3	DAC0
4	DAC1
5	CP1+
6	CP1-
7	AIN0.0
8	AIN0.1
9	DAC0
10	DAC1
11	AIN0.6
12	AIN0.7

Table 4. J11 Jumper Block Pin Descriptions



5.9. Expansion I/O Connector (J24)

The 96-pin expansion I/O connector, J24, is used to connect daughter boards to the main target board. It provides access to all signal pins of the C8051F12x device. Pins for +3V, digital ground, analog ground and the unregulated power supply (VUNREG) are also available. The VUNREG pin is connected directly to the unregulated +V pin of the P1 power connector.

Pin #	Description	Pin #	Description	Pin #	Description
A-1	+3VD2 (+3.3VDC)	B-1	DGND (Digital Gnd)	C-1	XTAL1
A-2	MONEN	B-2	P1.7	C-2	P1.6
A-3	P1.5	B-3	P1.4	C-3	P1.3
A-4	P1.2	B-4	P1.1	C-4	P1.0
A-5	P2.7	B-5	P2.6	C-5	P2.5
A-6	P2.4	B-6	P2.3	C-6	P2.2
A-7	P2.1	B-7	P2.0	C-7	P3.7
A-8	P3.6	B-8	P3.5	C-8	P3.4
A-9	P3.3	B-9	P3.2	C-9	P3.1
A-10	P3.0	B-10	P0.7	C-10	P0.6
A-11	P0.5	B-11	P0.4	C-11	P0.3
A-12	P0.2	B-12	P0.1	C-12	P0.0
A-13	P7.7	B-13	P7.6	C-13	P7.5
A-14	P7.4	B-14	P7.3	C-14	P7.2
A-15	P7.1	B-15	P7.0	C-15	P6.7
A-16	P6.6	B-16	P6.5	C-16	P6.4
A-17	P6.3	B-17	P6.2	C-17	P6.1
A-18	P6.0	B-18	P5.7	C-18	P5.6
A-19	P5.5	B-19	P5.4	C-19	P5.3
A-20	P5.2	B-20	P5.1	C-20	P5.0
A-21	P4.7	B-21	P4.6	C-21	P4.5
A-22	P4.4	B-22	P4.3	C-22	P4.2
A-23	P4.1	B-23	P4.0	C-23	TMS
A-24	TCK	B-24	TDI	C-24	TDO
A-25	/RST	B-25	DGND (Digital Gnd)	C-25	VUNREG
A-26	AGND (Analog Gnd)	B-26	DAC1	C-26	DAC0
A-27	CP1-	B-27	CP1+	C-27	CP0-
A-28	CP0+	B-28	VREF	C-28	VREFD
A-29	VREF0	B-29	VREF1	C-29	AIN0.7
A-30	AIN0.6	B-30	AIN0.5	C-30	AIN0.4
A-31	AIN0.3	B-31	AIN0.2	C-31	AIN0.1
A-32	AIN0.0	B-32	AGND (Analog Gnd)	C-32	AV+ (+3.3VDC Analog)

Table 5. J1 Pin Descriptions

6. Serial Adapter

The Serial Adapter provides the interface between the PC's RS232 serial port and the C8051F12x's in-system debug/programming circuitry. The Serial Adapter connects to the C8051F12x JTAG debug interface on the target board using the 10-pin connector on the Serial Adapter labeled "JTAG", see Figure 4. (The Serial Adapter supports both Cygnal JTAG and C2 debug interfaces.). All Serial Adapters may be powered from the target board, but the EC1 and EC2 Serial Adapter units can not provide power to the target board. Table 6 shows the pin definitions for the Serial Adapter's JTAG connector.

Notes:

1. When powering the Serial Adapter via the JTAG connector, the input voltage to the JTAG connector's power pin must be 3.0 to 3.6VDC. Otherwise, the Serial Adapter must be powered directly by connecting the AC/DC adapter to the Serial Adapter's DC power jack.
2. The Serial Adapter requires a target system clock of 32Khz or greater.

Pin #	Description
1	3.0 to 3.6VDC Input
2	GND (Ground)
4	TCK (C2DAT)
5	TMS
6	TDO
7	TDI (C2CLK)
3,8,9,10	Not Connected

Table 6. DEBUG Connector Pin Descriptions

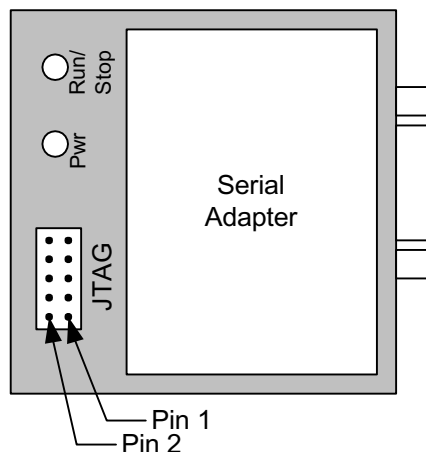


Figure 4. Serial Adapter JTAG Connector



7. Schematic

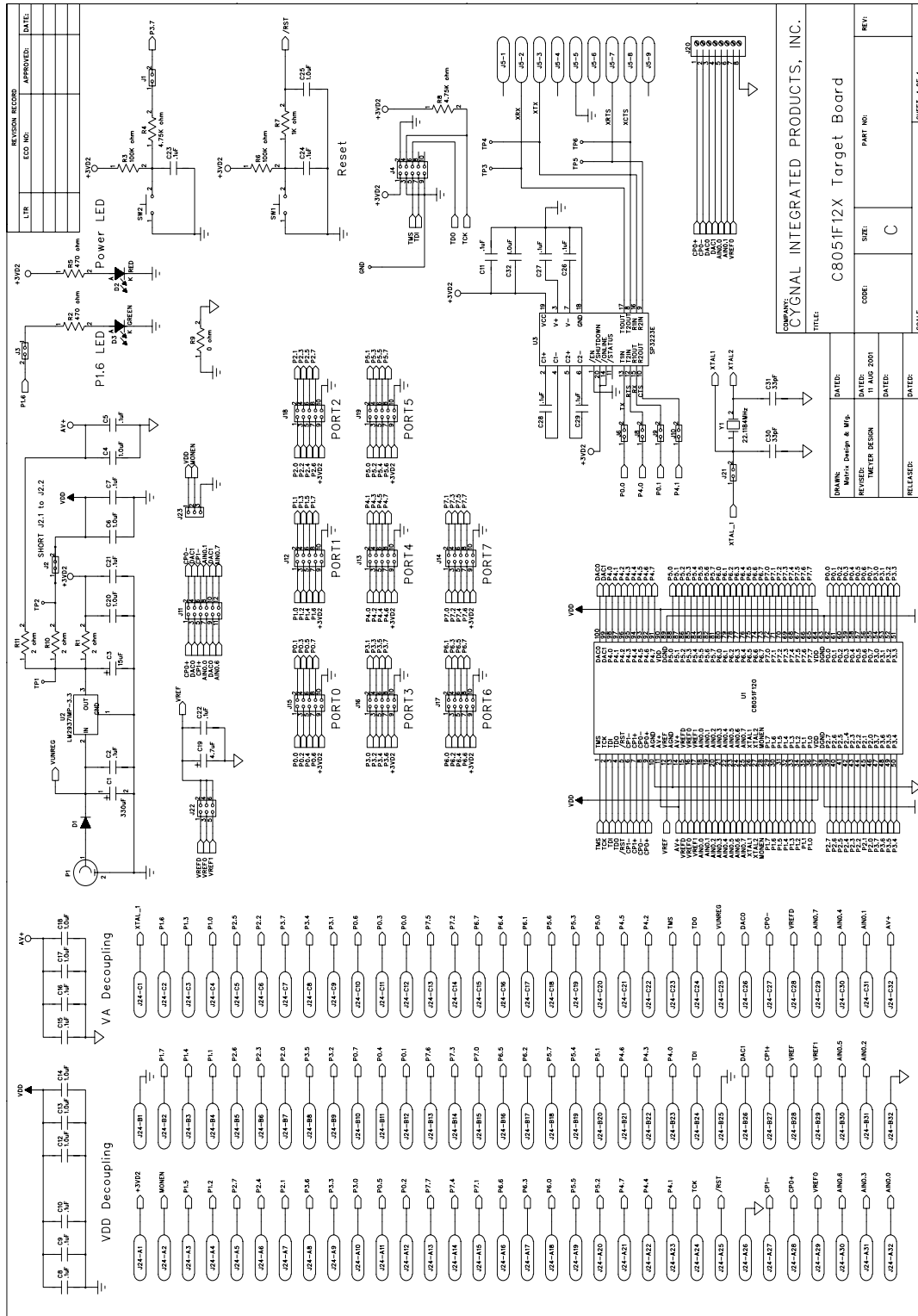


Figure 5. C8051F12x Target Board Schematic