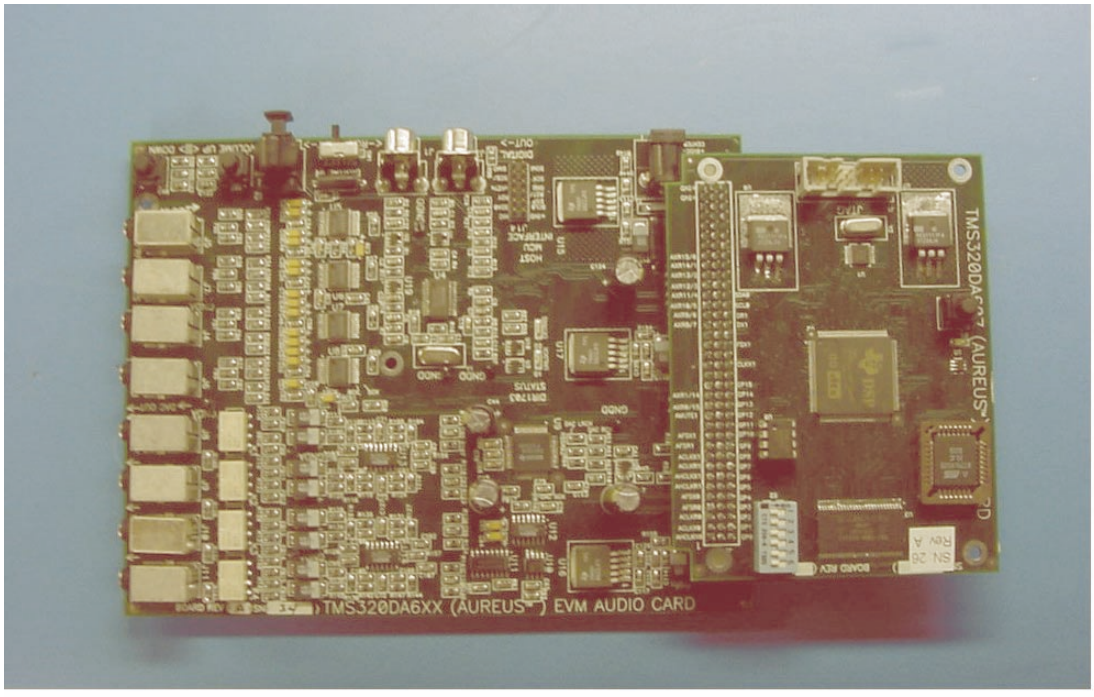
MOMENTUM DATA SYSTEMS

DA6XX Evaluation Module Hardware Manual

DA607 Evaluation Module (DA607EVM)

DA6XX EVM Audio Card (AC6XX) / DA607 Processor Card (PC607)



April 28, 2004 **MN0228 - Version 1.1**

Digital Audio Research and Development *Momentum Data Systems, Inc.*

IMPORTANT NOTICE

Momentum Data Systems and its subsidiaries (MDS) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgment, including those pertaining to warranty, patent infringement, and limitation of liability.

MDS warrants performance of its products to the specifications applicable at the time of sale in accordance with MDS's standard warranty. Testing and other quality control techniques are utilized to the extent MDS deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Customers are responsible for their applications using MDS components.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

MDS assumes no liability for applications assistance or customer product design. MDS does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of MDS covering or relating to any combination, machine, or process in which such products or services might be or are used. MDS's publication of information regarding any third party's products or services does not constitute MDS's approval, license, warranty or endorsement thereof.

Reproduction of information in MDS data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations and notices. Representation or reproduction of this information with alteration voids all warranties provided for an associated MDS product or service, is an unfair and deceptive business practice, and MDS is not responsible nor liable for any such use.

Resale of MDS's products or services with *statements different from or beyond the parameters* stated by MDS for that product or service voids all express and any implied warranties for the associated MDS product or service, is an unfair and deceptive business practice, and MDS is not responsible nor liable for any such use.

Mailing Addresses:

Momentum Data Systems, Inc.

17330 Brookhurst St., Ste 230
Fountain Valley, CA 92708
714-378-5805

6151 Powers Ferry Road, Ste. 130
Atlanta, GA 30339-2959
770-818-9180

Email: support-pa@mds.com

PREFACE Read This First

About This Manual

This is the Hardware Manual for the DA6XX Evaluation Module (DA6XXEVM). This manual describes how to setup and operate the DA6XXEVM. For those eager to get started, a Quick Start Guide is provided immediately following this Preface to help you "Plug in and go" with a quick "Listening Test."

The DA6XXEVM helps you evaluate characteristics of the Aureus™ TMS320DA601/607 digital signal processor (DSP) to determine if it meets your application requirements. The DA6XXEVM is intended to be operated on a desktop or lab bench with the use of a Spectrum Digital XDS510PP Plus or equivalent JTAG emulator. The DA6XXEVM also supports "Flash Boot" for operation without an emulator. The DA6XXEVM comes equipped with a power adapter.

This manual assumes that you are familiar with working in a Windows environment and understand general and technical PC terminology.

How to Use This Manual

This users guide provides the following types of information about the DA6XXEVM:

- The Quick Start Guide, immediately following this Preface, provides procedures for:
  - Setup Hardware: DA6XXEVM
  - Setup Hardware: SPI/I^{2}C Card: Debug Mode
  - Setup Hardware: SPI/I^{2}C Card: Parallel Debug Mode
  - Setup Software: Run From Flash
  - Setup Software: Run Test Programs From Emulator
  - Building and Running Performance Audio System Code
  - Communicating With Performance Audio System Code
- Chapter 1 provides general information about the features of the DA6XXEVM, including a hardware overview of the DA6XXEVM.
- Chapter 2 describes details of the DA6XX EVM Audio Card (AC6XX).

- Chapter 3 describes details of the DA601 Processor Card (PC601).
- Chapter 4 describes details of the DA607 Processor Card (PC607).
- Chapter 5 describes details of the SPI/I^2C Card (SPI Dongle).
- Chapter 6 describes test and example software for the DA6XXEVM hardware.
- Chapter 7 describes Performance Audio software and firmware releases.

FCC Warning

This equipment is intended for use in a laboratory test environment only. It generates, uses, and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to subpart J of part 15 of FCC rules, which are designed to provide reasonable protection against radio frequency interference. Operation of this equipment in other environments may cause interference with radio communications, in which case the user at his own expense will be required to take whatever measures may be required to correct this interference.

Trademarks

Aureus is a registered trademark of Texas Instruments.

Windows and *MS-DOS* are registered trademarks of the Microsoft Corporation.

Related Documents from Texas Instruments

See TI's web site at http://dspvillage.ti.com.

Quick Start Guide

Note: The following items are not included in the DA6XXEVM but are required. Requirements and installation for use of these items are described in detail in Appendix B:

- *Texas Instruments C6000 PC Code Composer Studio IDE, Version 2.2 (DSP4387U). Texas Instruments Code Composer Studio (CCS) is a fully integrated development environment supporting the Texas Instruments TMS320DA6XX platform.*
- *Texas Instruments FlashBurn Flash Memory Programmer Utility, version 2.71 or later.*
- *Spectrum Digital XDS510PP Plus or equivalent JTAG emulator.*
- *The Hex Conversion Utility (`coff2hex`) contained in the Performance Audio PA17 Software Development Kit Release*

This Quick Start Guide describes procedures for setup and operation of your DA6XXEVM.

As a first step, you can easily setup and run a "Listening Test" (factory-programmed into flash memory on the DA6XXEVM processor card) by performing procedures [1] and [4] listed below. The remainder of procedures are described in following paragraphs.

Procedures provided in this Quick Start Guide:

1. Setup Hardware: DA6XXEVM, page QSG-2.
2. Setup Hardware: SPI/I^{2}C Card: Debug Mode, page QSG-2.
3. Setup Hardware: SPI/I^{2}C Card: Parallel Debug Mode, page QSG-4.
4. Setup Software: Run From Flash, page QSG-6.
5. Setup Software: Run Test Programs From Emulator, page QSG-6.

   Note: Setup Software for SPI/I^{2}C Card is not yet available.

6. Building and Running Performance Audio System Code, page QSG-7.
7. Communicating With Performance Audio System Code, page QSG-8.

The "Listening Test" factory-programmed into flash memory is actually a Performance Audio PA17 release. As described above, performing procedures [1] and [4] runs PA17 via Flash Boot. Alternately, Performance Audio releases can be built then run via JTAG Boot by performing procedures [1] and [6].

Memory diagnostic and audio "loopback" test programs are provided on "DA6XXEVM Hardware Distribution CD" in file "`EVM_Hardware_Dist_CD_YYMMDD.zip`," where "`YYMMDD`" is the date of the release. After unzipping this file on your hard drive, the memory and loopback test programs are located in directories "`\tests\memory\`" and "`\tests\loopback\`." These programs can be setup and run by performing procedures [1] and [5]. General description of the functionality of these programs is also provided in these procedures.

Hardware setup procedure [1] does not include the optional DA6XX SPI/I^{2}C Card (SPI Dongle). The SPI Dongle can be used to communicate with Performance Audio System Code by performing procedures [1], [2] (or [3]), and [7].[6]

Setup Hardware: DA6XXEVM

The connectors on the Audio Card (motherboard) are identified by the silkscreen on the board. For a detailed description of connectors, see Figure 2-1 and Table 2-1 in Chapter 2.

1. Connect all desired analog inputs.
2. Connect all desired analog outputs.
3. Connect all desired digital input(s).
4. Set the digital input selector switch (SW1) to the appropriate position. The silkscreen on the Audio Card indicates the appropriate position for coaxial or optical input.
5. If desired, connect the digital output.

Note: *If using the optional SPI/I^{2}C Card, skip to Setup Hardware: SPI/I^{2}C Card: Debug Mode or Setup Hardware: SPI/I^{2}C Card: Parallel Debug Mode.*

6. Connect your JTAG emulator to the JTAG header (J1) on the Processor Card (daughterboard).
7. Connect the Power Adapter to the power jack on the Audio Card.

Note: *If not using the SPI/I^{2}C Card, hardware setup is complete.*

Setup Hardware: SPI/I^{2}C Card: Debug Mode

Refer to Figure 0-1 and Table 0-1 for performing the following setup procedure.

1. On the SPI/I^{2}C Card, set jumper JP1 to connect pins 1 and 2.
   Note: Pin 1 is indicated by the silkscreen on the board.
2. Remove the serial EEPROM (U8) from the Processor Card.
3. Using the SPI/I^{2}C power cable (the jumper cable, not the DA6XXEVM Power Adapter), connect power jack (J4) on the SPI/I^{2}C Card to the power jack on the Audio Card.
4. Connect your JTAG emulator to the JTAG header (J1) on the Processor Card.
5. Connect the 14-pin ribbon cable (keyed) from Target MCU Interface header (J14) on the SPI/I^{2}C Card to the Host MCU Interface header (J14) on the Audio Card.
6. Connect the RS-232 pass-through cable from your PC's serial port to the RS-232 connector (J2) on the SPI/I^{2}C Card.
7. Connect the Power Adapter to power jack (J12) on the SPI/I^{2}C Card.

6. Documentation for software setup of DA6XXEVM for use with SPI/I^{2}C Card in Debug or Parallel Debug mode is not yet available.

FIGURE 0-1 **Setup SPI/I2C Card: Debug Mode**

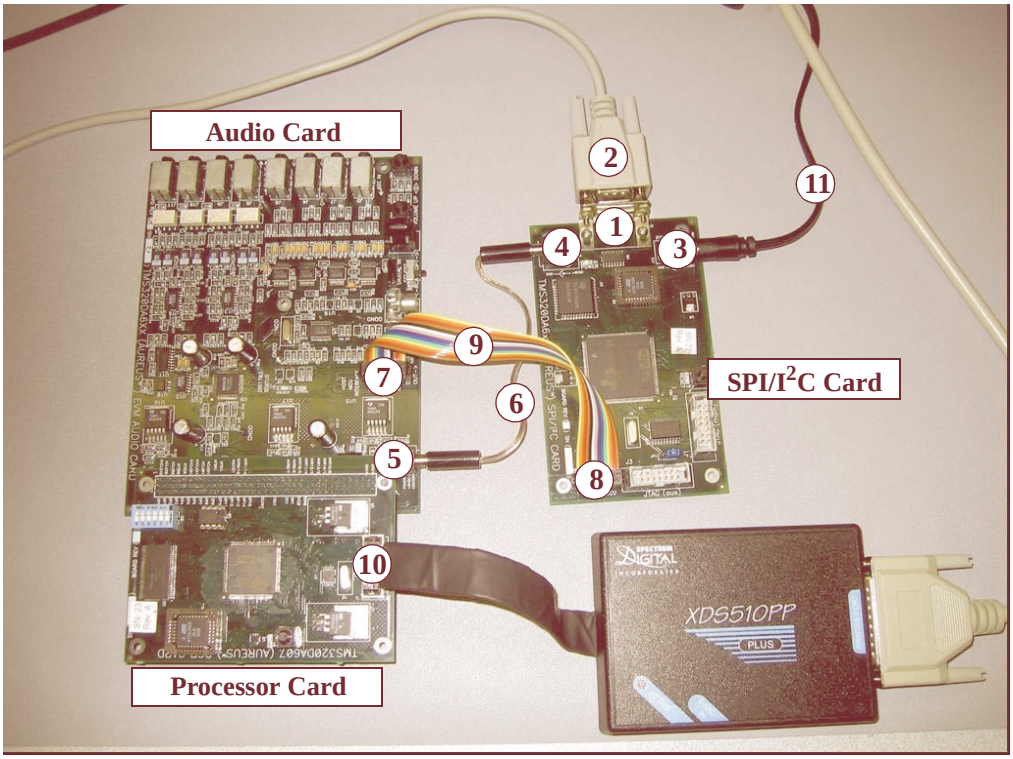


TABLE 0-1 **Key for Figure 0-1**

Number	Description
1	RS-232 connector
2	RS-232 cable
3	Power jack (J12)
4	Power jack (J4)
5	Audio Card power jack
6	Power cable (jumper cable)
7	Audio Card Host MCU Interface header (J14)
8	SPI/I^{2}C Card Target MCU Interface header (J14)
9	SPI cable (keyed)
10	Processor Card JTAG header (J1)
11	Power Adapter cable

Setup Hardware: SPI/I^{2}C Card: Parallel Debug Mode

Note: *As described in Appendix B, Section B.2.4.3 (XDS510PP Plus Texas Instruments and Momentum Data Systems Software), it is necessary to add a second CPU to Code Composer Studio setup for Parallel Debug Mode.*

Refer to Figure 0-2 and Table 0-2 for performing the following setup procedure.

1. On the SPI/I^{2}C Card, set jumper JP1 to connect pins 2 and 3.
   Note: Pin 1 is indicated by the silkscreen on the board.
2. Remove the serial EEPROM (U8) from the Processor Card.
3. Connect your JTAG emulator to the JTAG Main header (J1) on the SPI/I^{2}C card.
4. Using the SPI/I^{2}C power cable (the jumper cable, not the DA6XXEVM Power Adapter), connect power jack (J4) on the SPI/I^{2}C Card to the power jack on the Audio Card.
5. Connect the 14-pin ribbon cable (keyed) from Target MCU Interface header (J14) on the SPI/I^{2}C Card to the Host MCU Interface header (J14) on the Audio Card.
6. Connect the RS-232 pass-through cable from your PC's serial port to the RS-232 connector (J2) on the SPI/I^{2}C Card.
7. Connect the JTAG cable from JTAG Aux header (J3) on SPI/I^{2}C Card to JTAG header (J1) on the Processor Card.
8. Connect power cable to power jack J12 on SPI/I2C Card.

FIGURE 0-2 **Setup SPI/I2C Card: Parallel Debug Mode**

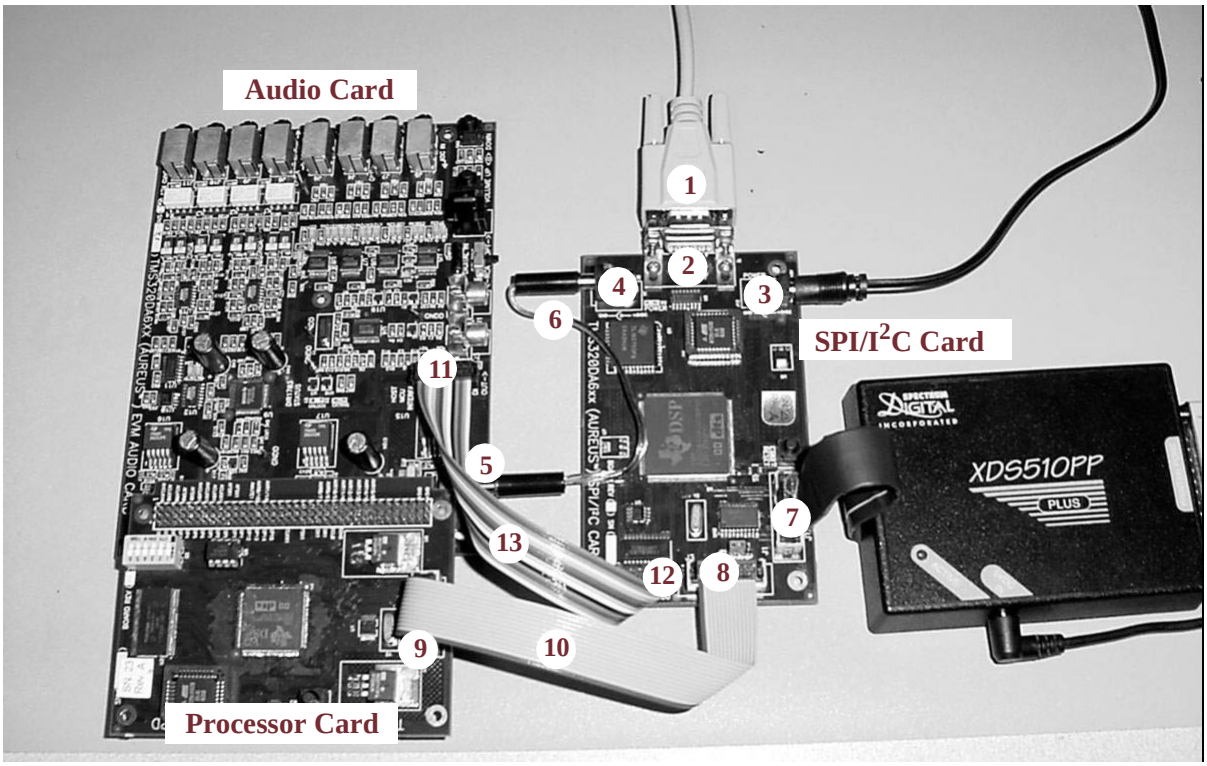


TABLE 0-2 **Key for Figure 0-2**

Number	Description
1	RS-232 cable
2	RS-232 connector
3	Power jack (J12)
4	Power jack (J4)
5	Audio Card power jack
6	Power cable (jumper cable)
7	SPI/I^{2}C JTAG Main header (J1)
8	SPI/I^{2}C JTAG Auxiliary header (J3)
9	Processor card JTAG header (J1)
10	JTAG cable (keyed)
11	Audio Card Host MCU Interface header (J14)
12	SPI/I^{2}C Card Target MCU Interface header (J14)
13	SPI cable (keyed)

Setup Software: Run From Flash[7]

1. Check that digital input(s) and analog output(s) are connected and the DA6XXEVM is powered on.
2. Source material may be played at this point (digital output from source to digital input on DA6XXEVM), and should produce output on the analog outputs.[8]

Setup Software: Run Test Programs From Emulator

Memory diagnostic and audio "loopback" test programs are provided on "DA6XXEVM Hardware Distribution CD" in file "`EVM_Hardware_Dist_CD_YYMMDD.zip`," where "`YYMMDD`" is the date of the release. After unzipping this file on your hard drive, the memory and loopback test programs are located in directories "`\tests\memory\`" and "`\tests\loopback\`":

Memory diagnostic program:

- SDRAM test (`sdram_wbyte_test.out`)

Audio loopback programs:

- 8-Channel Analog Input with Analog Output (`Ain4_Aout4.out`)
- Left/Right Front Analog Input with Digital Output (`Ain1_Dout.out`)
- Left/Right Surround Analog Input with Digital Output (`Ain2_Dout.out`)
- Center/Subwoofer Analog Input with Digital Output (`Ain3_Dout.out`)
- Left/Right Back Analog Input with Digital Output (`Ain4_Dout.out`)
- Digital Input with Analog Output (`Din_Aout4.out`)
- Digital Input with Digital Output (`Din_Dout.out`)

1. Copy file "`EVM_Hardware_Dist_CD_YYMMDD.zip`," from the CD to an arbitrary directory, then unzip the file allowing directory creation.
   Note: The remainder of this procedure uses "`c:\temp\`" as the arbitrary directory.
2. With the DA6XXEVM powered-on, start Code Composer Studio (CCS).
3. In the CCS pull-down menu, select **Debug / Reset CPU**.[9]
4. Select **File / Load GEL** then navigate to: "`c:\temp\gel\board_evmda607_d607.gel`."
5. Select **File / Load Program** then navigate to "`c:\temp\`*testfolder*`\`*filename*`.out`," where *testfolder* is either "`memory`" or "`loopback`," and *filename* is any of the "`.out`" files within those folders.
6. Select **Debug / Run**.
   *Note: The above can also be performed pressing the **F5** key, or click the **Run** icon on the left side of the window.*

7. Note that if an emulator is connected to the DA6XXEVM, starting Code Composer Studio while the DA6XXEVM is running from flash will halt the DSP and stop output audio.
8. See Appendix D (Errata) for limitiations of flash boot.
9. It is critical to always select **Debug / Reset CPU** then load/reload the GEL file before running a program.

7. When finished, select **Debug / Halt** to stop the program.
   *Note: The above can also be performed pressing **Shift-F5**, or click the **Halt** icon.*
8. To run the same program again, repeat steps 3 and 4, then select **File / Reload Program**, then step 6. To run a different program, repeat steps 3 through 6.

Building and Running Performance Audio System Code

Note: The "Listening Test" factory-programmed into flash memory is a Performance Audio PA17 Release. Performing Setup Software: Run From Flash on page QSG-6 runs PA17 via Flash Boot. The procedure below describes how to build and run Performance Audio System Code via JTAG Boot.

Performance Audio code is provided on "DA6XXEVM Hardware Distribution CD":[10]

- Performance Audio PA17 Release `(pa17i_YYMMDD.zip)`
  `Note: YYMMDD` is a date code.
- Performance Audio PA17 Software Development Kit Release `(pa17isdk_YYMMDD.zip, pa-pa17isdk_YYMMDD.zip`

The following procedure describes building and running PA17 Software Development Kit firmware:

1. Perform procedure(s):
   - Setup Hardware: DA6XXEVM
   - (Optional:) Setup Hardware: SPI/I^{2}C Card: Debug Mode
2. Copy file "`pa-pa17isdk_YYMMDD.zip`" from the CD to an arbitrary directory on your hard drive, then unzip it allowing directory creation.
   Note: The remainder of this procedure uses "`c:\temp`" as the arbitrary directory.
3. With the DA6XXEVM powered-on, start Code Composer Studio (CCS).
4. In the CCS pull-down menu, select **Project / Open**, then navigate to select:
   "`c:\temp\pa\f\s3\pa17isdk.pjt`"
5. In a separate DOS window, type "`subst T: c:\temp`"
   Note: If `T:` drive is already substituted, type "`subst T: /D`" to un-substitute it first.
6. In the CCS pull-down menu, select Build Configuration:[11]
   EVMDA607_D607A003_Debug
7. Select **Project / Build**. The resulting "`pa17isdk.out`" file will appear in directory:
   `c:\temp\pa\f\s3\EVMDA607_D607A003_Debug`

10. For more detailed instructions and additional information on building and running Performance Audio code, please refer to Chapter 7, Performance Audio Software and Firmware.

11. In step 5, note that one may also select Build Configuration **EVMDA607_D607A003_Release**. The Debug configuration is intended for debugging purposes; the Release configuration will have better performance, and should be used in the case of performance testing.

8. Perform steps [3] through [7] from Setup Software: Run Test Programs From Emulator above for running "`pa17isdk.out`"file.

Note: ***To build a new "`.out`" file with a different Input/Output configuration:***

1. Make changes to file "`c:\temp\pa\f\s3\pa17isdk-cus_atboot_s.c`." See comments within the file for instructions.

2. (Optional) Navigate to **Project / Build Options**, *click on the* **Linker** *tab, then enter a new "filename`.out`" name in the* **Destination Output File** *field. This step is not mandatory, but if not done, the existing "`pa17isdk.out`" file will be overwritten.*

3. Perform steps 5 and 6 above.

Communicating With Performance Audio System Code

The DA6XX SPI/I^{2}C Card (SPI Dongle) can be used to communicate with Performance Audio System Code by performing the following procedure. This procedure describes communicating with the DSP after Flash Boot of the DA6XXEVM.

1. Perform procedure(s):
   - Setup Hardware: DA6XXEVM
   - Setup Hardware: SPI/I^{2}C Card: Debug Mode
2. Unzip "`pa17isdk_YYMMDD.zip`" into an arbitrary directory.
   Note: The remainder of this procedure uses "`c:\temp`" as the arbitrary directory.
3. Open a shell for entering commands, such as a DOS shell.
4. Establish the required path by executing:
   ```
   C:\ti\DosRun.bat
   ```
5. Substitute the `P:` drive to the location of the unzipped PA17 release, as defined in section "Software Deliverable Installation" in the *PAn User's Guide.*
   ```
   C:\> subst P: c:\temp\pa17i
   ```
6. Change to the `P:\` drive:
   ```
   C:\> P:
   ```
7. Add the `.\bin` directory to your path:
   ```
   P:\> SET PATH=%PATH%;P:\bin;
   ```
8. Enter a test alpha code command:
   ```
   P:\ > calfa -hpa17i-evmda607 -IP:\alpha readSTDReady
   ```

   If the response is anything other than "`alpha readSTDReady`", please consult Chapters 2 and 4 of the *PAn User's Guide* for configuring your system to communicate with the PA System.

Table of Contents

Table of Contents *(continued)*

Table of Contents *(continued)*

Table of Contents *(continued)*

Table of Contents *(continued)*

List of Figures

List of Figures *(continued)*

List of Tables

CHAPTER 1 *Introduction*

This chapter presents:

- An Overview of the DA6XX Evaluation Module (DA6XXEVM)
- DA6XXEVM Hardware Functional Overview
- DA6XXEVM Features
- DA6XXEVM Software Overview

1.1 Overview

The DA6XXEVM is a low-cost, general-purpose platform for the development, analysis, and testing of DA601/607 digital signal processor (DSP) algorithms and applications. The DA6XXEVM allows you to evaluate the DA601/607 DSP to determine how it may meet your application requirements. The DA6XXEVM hardware design information provides a reference design that facilitates your own DA601/607-based hardware development.

The DA6XXEVM is bundled with a "DA6XXEVM Hardware Distribution CD" which contains schematic diagrams, data sheets, Test/Example Software with source code and Performance Audio Software and Firmware releases. This integrated package is provided to assist you in hardware debugging and getting started with your custom hardware and software development. See the "`readme.txt`" file in the root directory of the CD for information about the CD's contents.

The DA6XXEVM operates "stand-alone" on a desktop or lab bench. Applications are typically loaded by Texas Instruments Code Composer Studio (CCS), using the board's embedded JTAG interface by a Spectrum Digital XDS510PP Plus or equivalent JTAG emulator connected to the DSP card's JTAG connector. The DA6XXEVM has "Flash Boot" capability that enables it to boot an application itself upon reset for a daughterboard designed with nonvolatile memory, as provided on the DA601 Processor Card (PC601) and DA607 Processor Card (PC607).

The DA6XXEVM is provided with a factory-programmed “Listening Test” program in the processor card’s flash memory, which is useful as a first step in getting the DA6XXEVM up and running in your development environment. See the Quick Start Guide for details on setting up and running this test program.

The processor card of the DA6XXEVM provides an interface which can be used to bread-board/prototype your custom system. Additional cards can be ordered from Momentum Data Systems, Inc.

Also provided in the DA6XXEVM package, a DA6XX SPI/I^{2}C Card (SPI Dongle) is useful for performing host communications until your host microcontroller code is working. This allows you to send “alpha code” to the DSP until you are ready to use the DA6XXEVM audio card’s Host MCU interface.

For I/O, performance, and power specifications, refer to the DA6XXEVM Data Sheet.

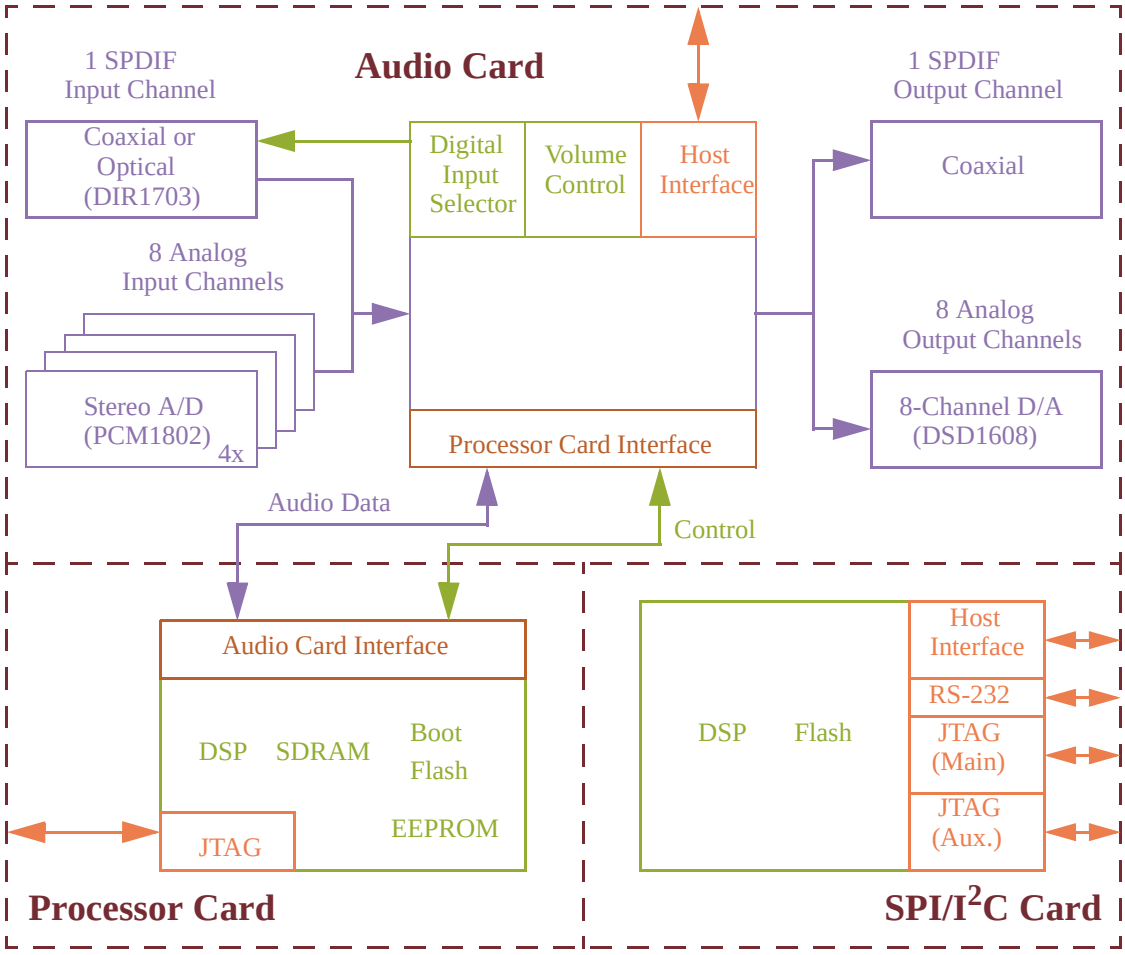
1.2 DA6XXEVM Hardware Functional Overview

The DA6XXEVM is composed of the following circuit boards:

- DA6XX EVM Audio Card (AC6XX)
- DA601 Processor Card (PC601)
- DA607 Processor Card (PC607)
- DA6XX SPI/I^{2}C Card (SPI Dongle)

Figure 1-1 shows a functional block diagram of the DA6XXEVM. For more detailed information, see the DA6XXEVM Data Sheet available at:
http://www.mds.com/Products/product.asp?prod=DA%2D60x%2DKIT

FIGURE 1-1 **DA6XXEVM Block Diagram---Functional Overview**



1.3 DA6XXEVM Features

The DA6XXEVM is operated stand-alone on a desktop with the use of an external power adapter; a Spectrum Digital XDS510PP Plus or equivalent JTAG emulator is used for downloading programs and software development. The DA6XXEVM provides embedded JTAG emulation support. The DA6XXEVM also supports “Flash Boot.”

The DA6XXEVM provides a DA601/607 hardware reference design that can assist you in the development of your own DA601/607-based products. In addition to providing a reference for interfacing the DSP to various types of memories and peripherals, the design also addresses power, clock and JTAG interfaces.

The following subsections summarize the features of the boards of the DA6XXEVM. More details of each card is provided in separate chapters. The DA6XXEVM consists of the following boards:

- DA6XX EVM Audio Card (AC6XX)
- DA601 Processor Card (PC601)
- DA607 Processor Card (PC607)
- DA6XX SPI/I^{2}C Card (SPI Dongle)

and:

- Power Adapter
- JTAG, MCU Host Interface and power cables for SPI Dongle

The Audio Card and DSP Card are delivered attached, but they may be separated.

The following items are not included in the DA6XXEVM but are required:

- Texas Instruments C6000 PC Code Composer Studio IDE, Version 2.2 (DSP4387U). Texas Instruments Code Composer Studio (CCS) is a fully integrated development environment supporting the Texas Instruments TMS320DA6XX platform.
- Spectrum Digital XDS510PP Plus or equivalent JTAG emulator.

Requirements and installation for use of these items are described in detail in Appendix B.

1.3.1 DA6XX EVM Audio Card (AC6XX)

The DA6XX EVM Audio Card (AC6XX) is a motherboard which supports the DA601/607 Processor Cards . The AC6XX provides external input/output connectivity and a site for attaching the Processor Card.

Features of the AC6XX include:

- 4 miniature stereo phone input jacks (8 analog input channels)
- 4 miniature stereo phone output jacks (8 analog output channels)
- 1 digital input (S/PDIF) with selector to use RCA connector (coaxial) or optical connector (TOSLINK)
- 1 digital output (S/PDIF) RCA connector (coaxial)
- Host Interface via Host MCU Interface connector (or DA6XX SPI/I^{2}C Card (SPI Dongle) may be used)
- Volume Up/Down Buttons
- Connector for attaching to the Processor Card
- Connector for attaching to the Power Adapter

For more information, see Chapter 2, DA6XX EVM Audio Card (AC6XX) Details.

1.3.2 DA601 Processor Card (PC601)

The DA601 Processor Card (PC601) is a daughterboard to the DA6XX Audio Card. The PC601 contains the Digital Signal Processor (DSP), memory, and other required support circuitry for the DSP. The PC601 allows full-speed verification of DA601 code using the Texas Instruments Code Composer Studio (CCS) source debugger.

Features of the PC601 include:

- JTAG interface
- User DIP switches and LED
- DSP reset button
- Connector for attaching to the DA6XX EVM Audio Card (AC6XX)

The AC6XX connector can be used to breadboard/prototype your custom system.

For more information, see Chapter 3, DA601 Processor Card (PC601) Details.

1.3.3 DA607 Processor Card (PC607)

The DA607 Processor Card (PC607) is a daughterboard to the DA6XX Audio Card. The PC607 contains the Digital Signal Processor (DSP), memory, and other required support circuitry for the DSP. The PC607 allows full-speed verification of DA607 code using the Texas Instruments Code Composer Studio (CCS) source debugger.

Features of the PC607 include:

- JTAG interface
- User DIP switches and LED
- DSP reset button
- Connector for attaching to the DA6XX EVM Audio Card (AC6XX)

The AC6XX connector can be used to breadboard/prototype your custom system.

For more information, see Chapter 4, DA607 Processor Card (PC607) Details.

1.3.4 DA6XX SPI/I^2C Card (SPI Dongle)

The DA6XX SPI/I^2C Card (SPI Dongle) is a daughterboard to the DA6XX Audio Card. It provides serial communications which may be used as an alternate to using the Host MCU Interface on the DA6XX EVM Audio Card (AC6XX).

Features of the SPI Dongle include:

- JTAG interfaces (Main and Auxiliary) with JTAG Configuration Jumper
- Target Host MCU Interface
- RS-232 Interface via DB-9 Connector
- Power LED
- Reset Button and LED
- Connectors for attaching to the Power Adapter (source) and to power the AC6XX

The SPI Dongle is useful for performing host communications until your host microcontroller code is working. This allows you to send “alpha code” to the DSP until you are ready to use the Host MCU interface on the AC6XX.

For more information, see Chapter 5, DA6XX SPI/I^{2}C Card (SPI Dongle) Details.

1.4 DA6XXEVM Software Overview

1.4.1 Test/Example Software

Test and example software are provided on the “DA6XXEVM Hardware Distribution CD” in file “`EVM_Hardware_Dist_CD_YYMMDD.zip`,” where “`YYMMDD`” is the date of the release.

- Memory Diagnostics Tests
- Audio Loop-Back Tests
- DA6XXEVM Audio Loopback Project

After unzipping this file on your hard drive, memory and audio loopback test programs are located in directories “`\tests\memory\`” and “`\tests\loopback\`.” An audio loopback project for DA601 Processor Card (PC601) and DA607 Processor Card (PC607) is located in directory “`\pc607\loopback\`.”

This software is useful for confidence testing and troubleshooting of hardware problems, as well as and learning how to build/run programs using the DA6XXEVM. This software is described in Chapter 6, “Test/Example Software,” on page 59.

1.4.2 Performance Audio Software and Firmware

Performance Audio software and firmware release packages are provided on the “DA6XXEVM Hardware Distribution CD”:

- Performance Audio PA17 Release
- Performance Audio PA17 Software Development Kit Release

These releases are described in Chapter 7, “Performance Audio Software and Firmware,” on page 63.

1.4.3 Texas Instruments Code Composer Studio (CCS)

Software development performed using the DA6XXEVM requires the Texas Instruments C6000 PC Code Composer Studio IDE, Version 2.2 (DSP4387U). Code Composer Studio (CCS) is a fully integrated development environment supporting the Texas Instruments TMS320DA6XX platform.

It is assumed that the user has CCS Version 2.2 or later.

The main features of CCS include:

- Integrated development environment with editor, debugger, project manager, profiler and Probe Points™
- C/C++ Compiler, Assembler, and Linker (Code Generation Tools)
- Instruction set simulator
- Real-time software kernel (DSP/BIOS™)
- Real-Time Data Exchange between host and target (RTDX™)
- Real-time analysis and data visualization tools

Requirements for, installation of, and use of these items are described in detail in Appendix B (Code Composer Studio for DA6XXEVM).

For more information on CCS, see the Development Tools section of the TI website: http://dspvillage.ti.com/

CHAPTER 2 *DA6XX EVM Audio Card (AC6XX) Details*

This chapter shows the details of the DA6XX EVM Audio Card (AC6XX) motherboard.

See Appendix A for links to web pages containing data sheets for the devices discussed in this section.

2.1 DA6XX EVM Audio Card (AC6XX) Major Components

Figure 1-1, “DA6XXEVM Block Diagram---Functional Overview,” on page 3, gives an overview of the circuitry on the AC6XX. The location, name and brief description of the major parts on the AC6XX are shown below in Figure 2-1 and Table 2-1. Schematic diagrams are provided in “AC607 Schematic Diagrams” on page 11.

FIGURE 2-1 **DA6XX EVM Audio Card (AC6XX) Photograph**

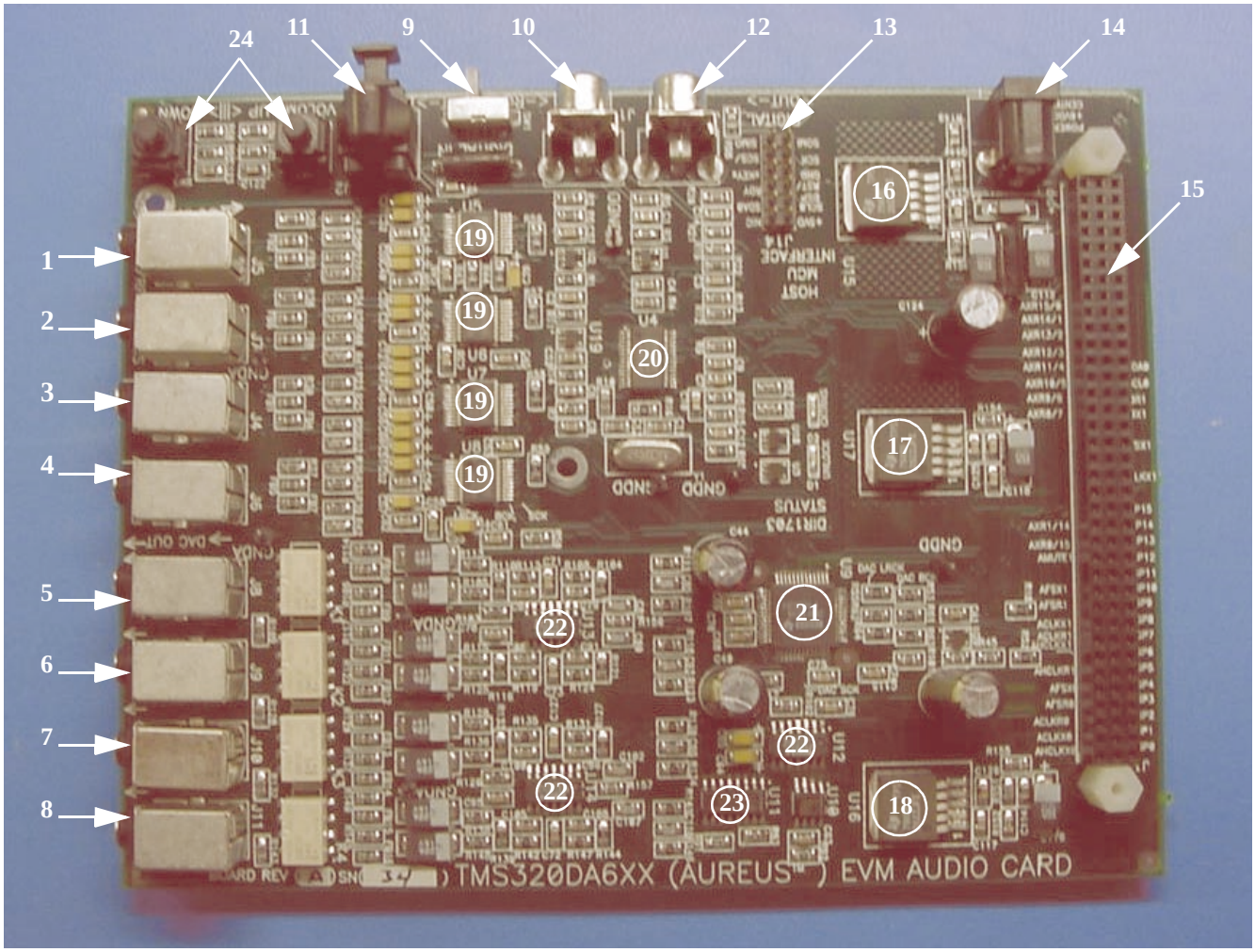


TABLE 2-1 **DA6XX Audio Card (AC6XX) Major Parts Description**

Number	Part Name	Description
1	LF/RF Analog Input	Input for Left/Right Front
2	LS/RS Analog Input	Input for Left/Right Surround
3	C/SW Analog Input	Input for Center/Subwoofer
4	LB/RB Analog Input	Input for Left/Right Back
5	LF/RF Analog Output	Output for Left/Right Front
6	LS/RS Analog Output	Output for Left/Right Surround
7	C/SW Analog Output	Output for Center/Subwoofer

Number	Part Name	Description
8	LB/RB Analog Output	Output for Left/Right Back.
9	Digital Input Selector Switch	Coaxial/Optical Input Selector.
10	Coaxial Digital Input	Input for SPDIF.
11	Optical Digital Input	Input for TOS.
12	Coaxial Digital Output	Output for SPDIF.
13	Host MCU Interface	Interface for host microcontroller.
14	Power Connector	Connects to Power Adapter or SPI Dongle.
15	Processor Card Connector	Connects to PC701/PC607 board.
16	TPS79601	Voltage regulator for Processor Card's 5 volt power supply. See "TPS79601/633 Data Sheet" on page 65.
17	TPS79633	Voltage regulator for 3.3 volt power supply. See "TPS79601/633 Data Sheet" on page 65.
18	TPS79601	Voltage regulator for 5 volt power supply. See "TPS79601/633 Data Sheet" on page 65.
19 (4x)	PCM1802	Stereo Analog-to-Digital Converter. See "PCM1802 Data Sheet" on page 65.
20	DIR1703	Digital Input Receiver. See "DIR1703 Data Sheet" on page 66.
21	DSD1608	8-Channel Digital-to-Analog Converter. See "DSD1608 Data Sheet" on page 66.
22 (3x)	OPA4344 	Quad Operational Amplifier. See "OPA4344 Data Sheet" on page 66.
23	ULN2003	Darlington transistor array. See "ULN2003 Data Sheet" on page 66.
24	Volume Up/Down Buttons	Controls for volume level increase/decrease.

2.2 AC607 Schematic Diagrams

The following schematics are provided herein for convenience. It is advised that you view the schematic diagrams provided on the "DA6XXEVM Hardware Distribution CD," as it may contain updated versions. See directory `/doc/Schematics/AC6XX`.

- AC607 Schematic Diagram: Title
- AC607 Schematic Diagram: Digital Input/Output
- AC607 Schematic Diagram: Analog Inputs 0-7
- AC607 Schematic Diagram: ADC, Inputs 0-3
- AC607 Schematic Diagram: ADC, Inputs 4-7
- AC607 Schematic Diagram: DAC, Outputs 0-7
- AC607 Schematic Diagram: Analog Outputs 0-3
- AC607 Schematic Diagram: Analog Outputs 4-7

- AC607 Schematic Diagram: Power
- AC607 Schematic Diagram: Connectors
- AC607 Schematic Diagram: User Interface

FIGURE 2-2 **AC607 Schematic Diagram: Title**

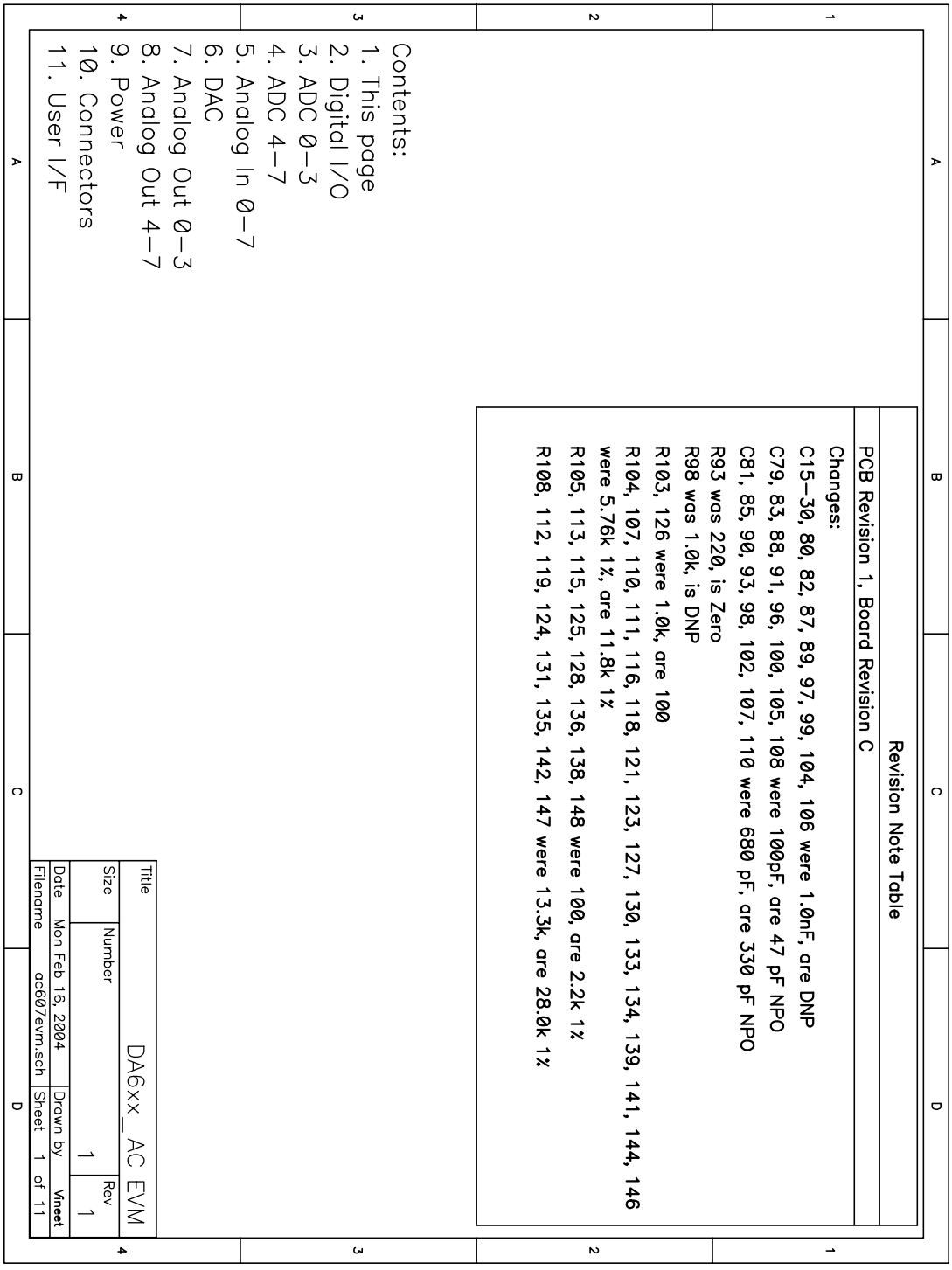


FIGURE 2-3 **AC607 Schematic Diagram: Digital Input/Output**

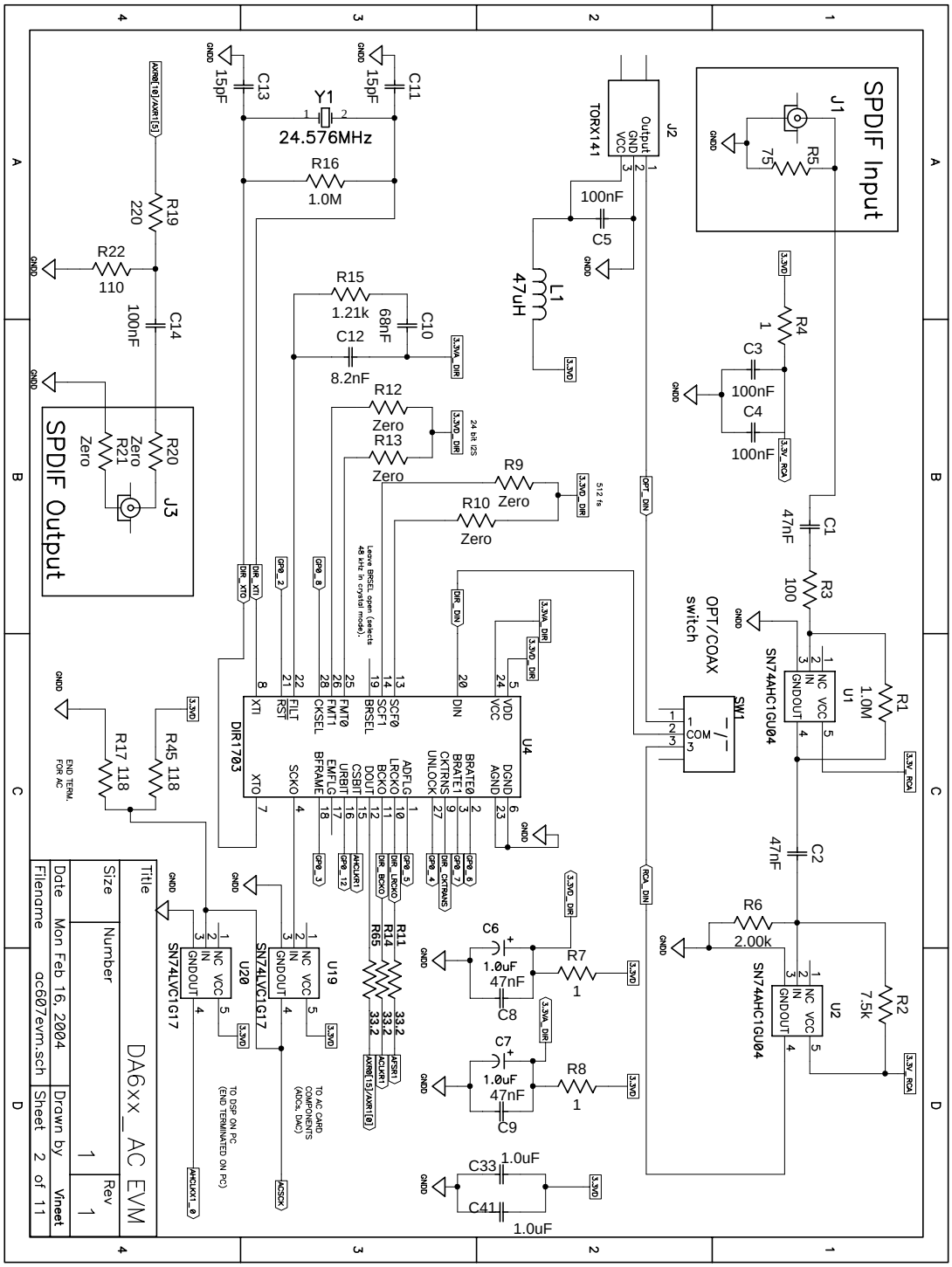


FIGURE 2-4 **AC607 Schematic Diagram: Analog Inputs 0-7**

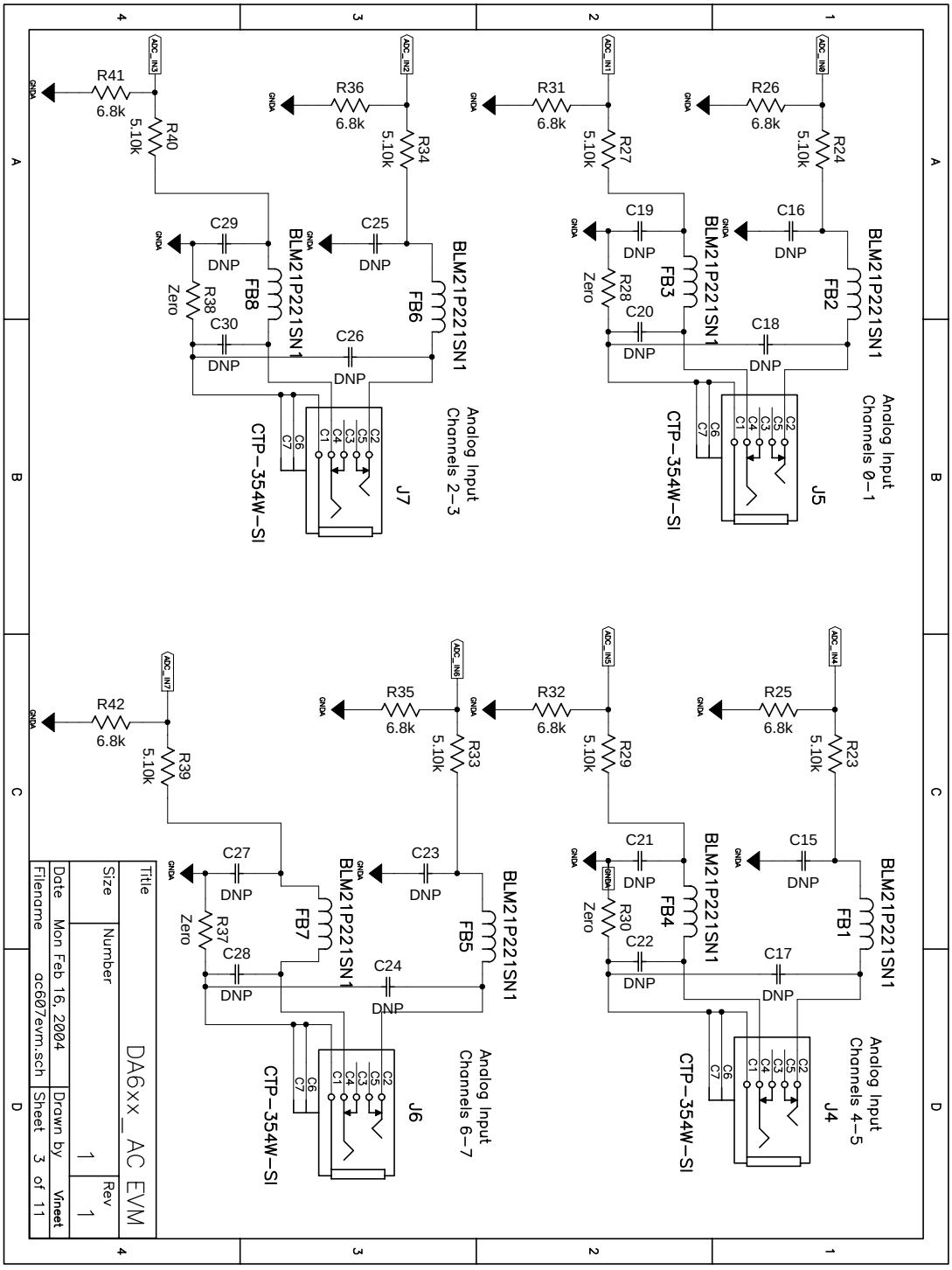


FIGURE 2-5 AC607 Schematic Diagram: ADC, Inputs 0-3

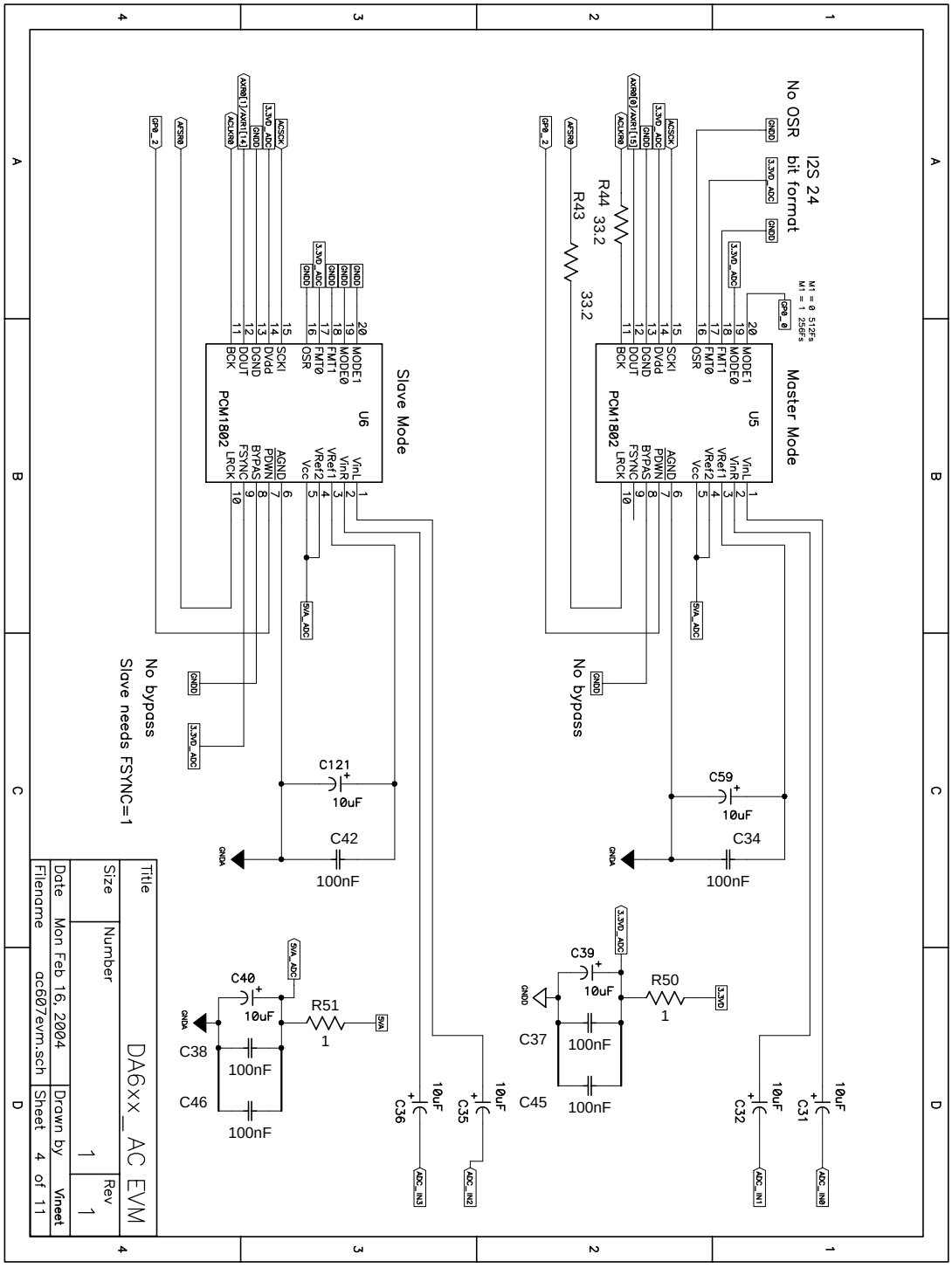


FIGURE 2-6 **AC607 Schematic Diagram: ADC, Inputs 4-7**

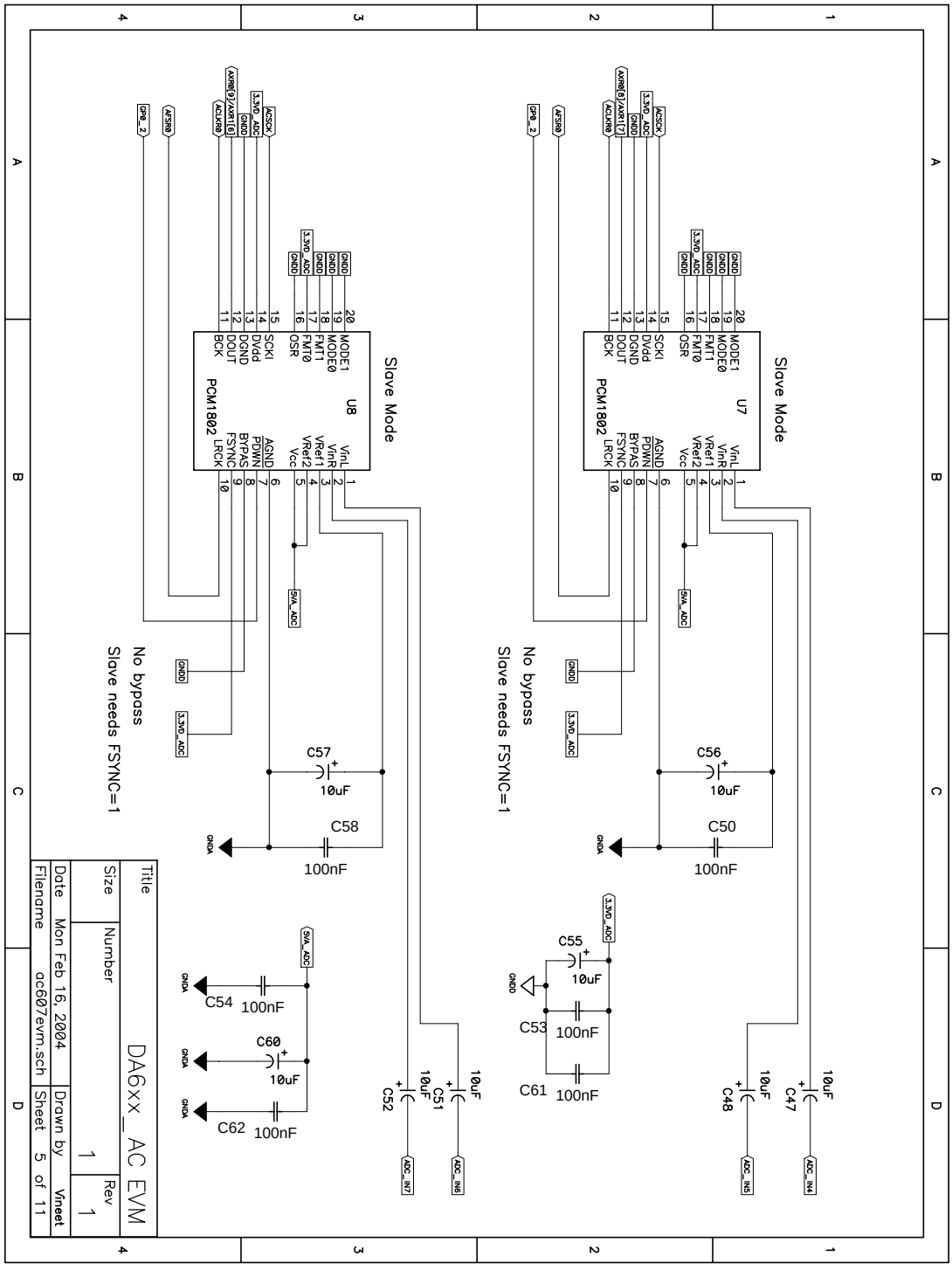


FIGURE 2-7 **AC607 Schematic Diagram: DAC, Outputs 0-7**

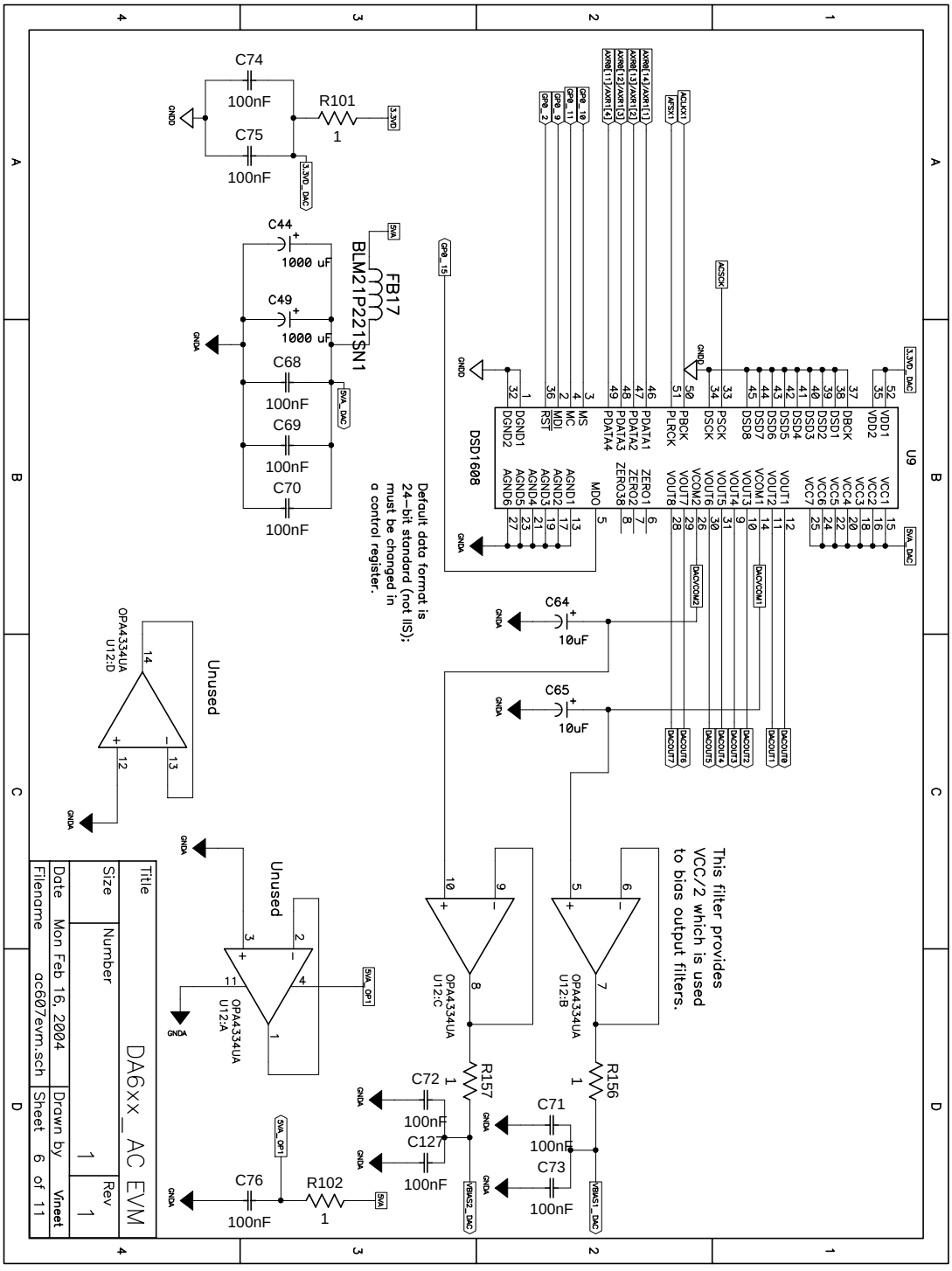


FIGURE 2-8 **AC607 Schematic Diagram: Analog Outputs 0-3**

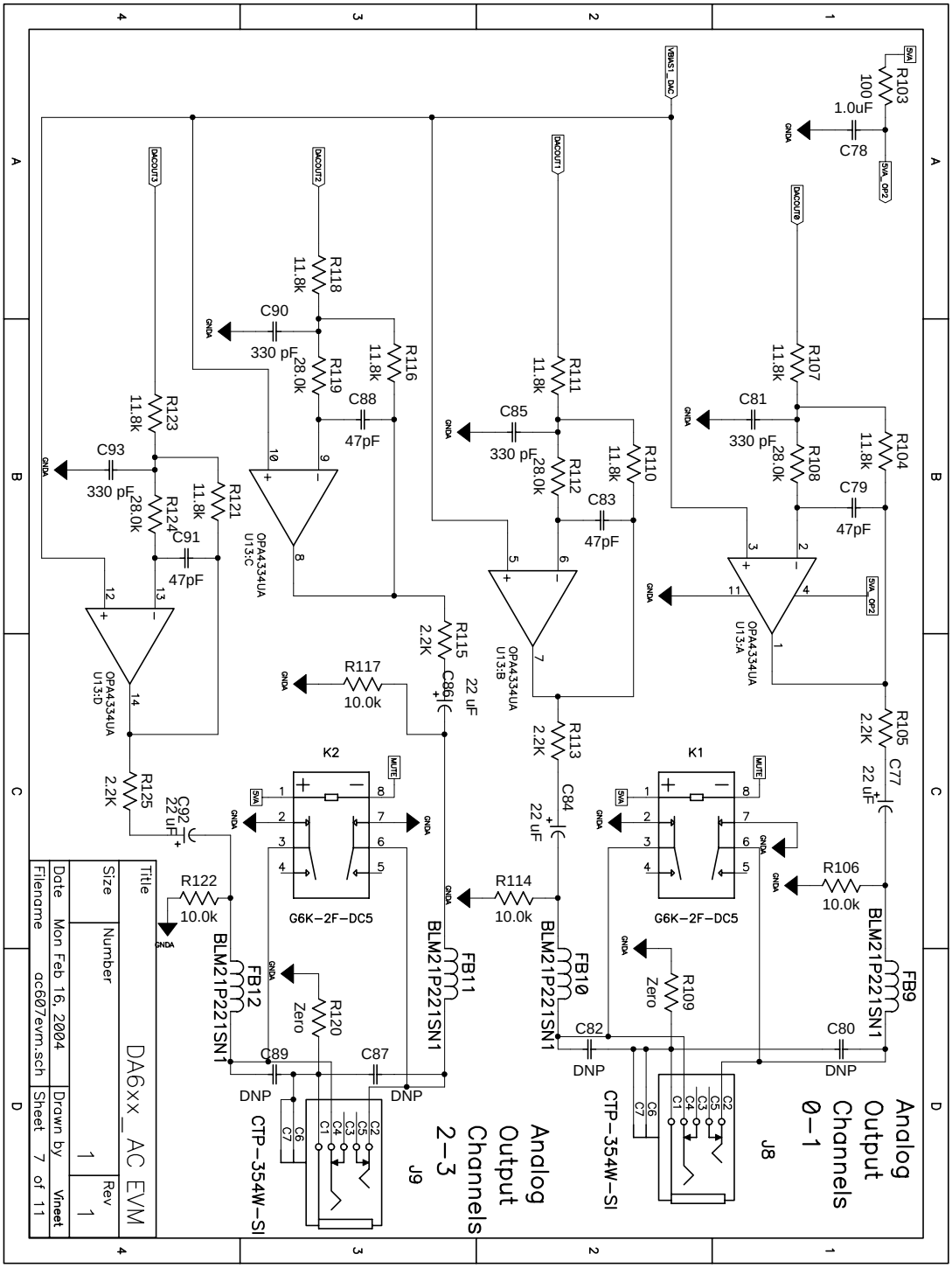


FIGURE 2-9 **AC607 Schematic Diagram: Analog Outputs 4-7**

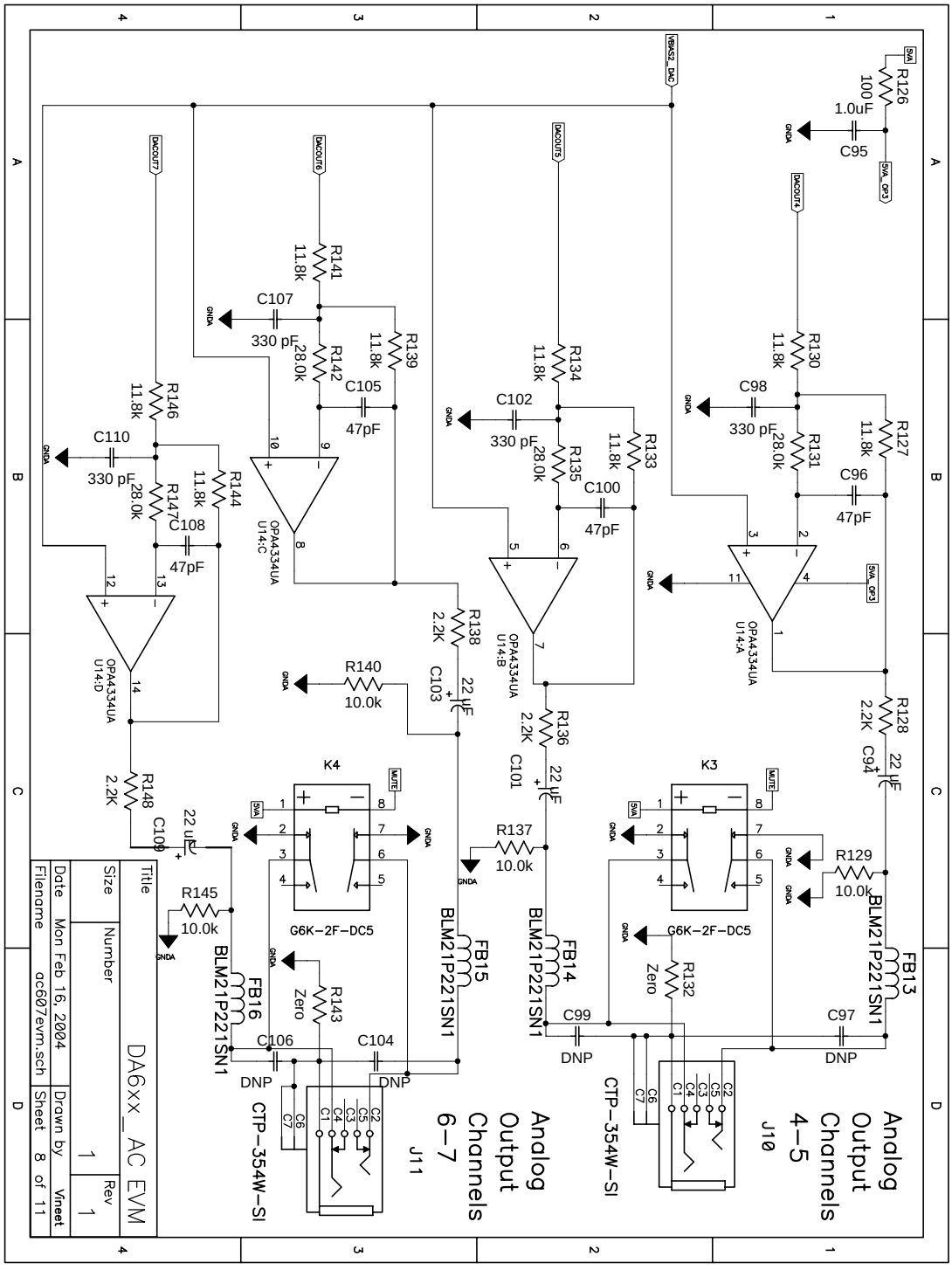


FIGURE 2-10 **AC607 Schematic Diagram: Power**

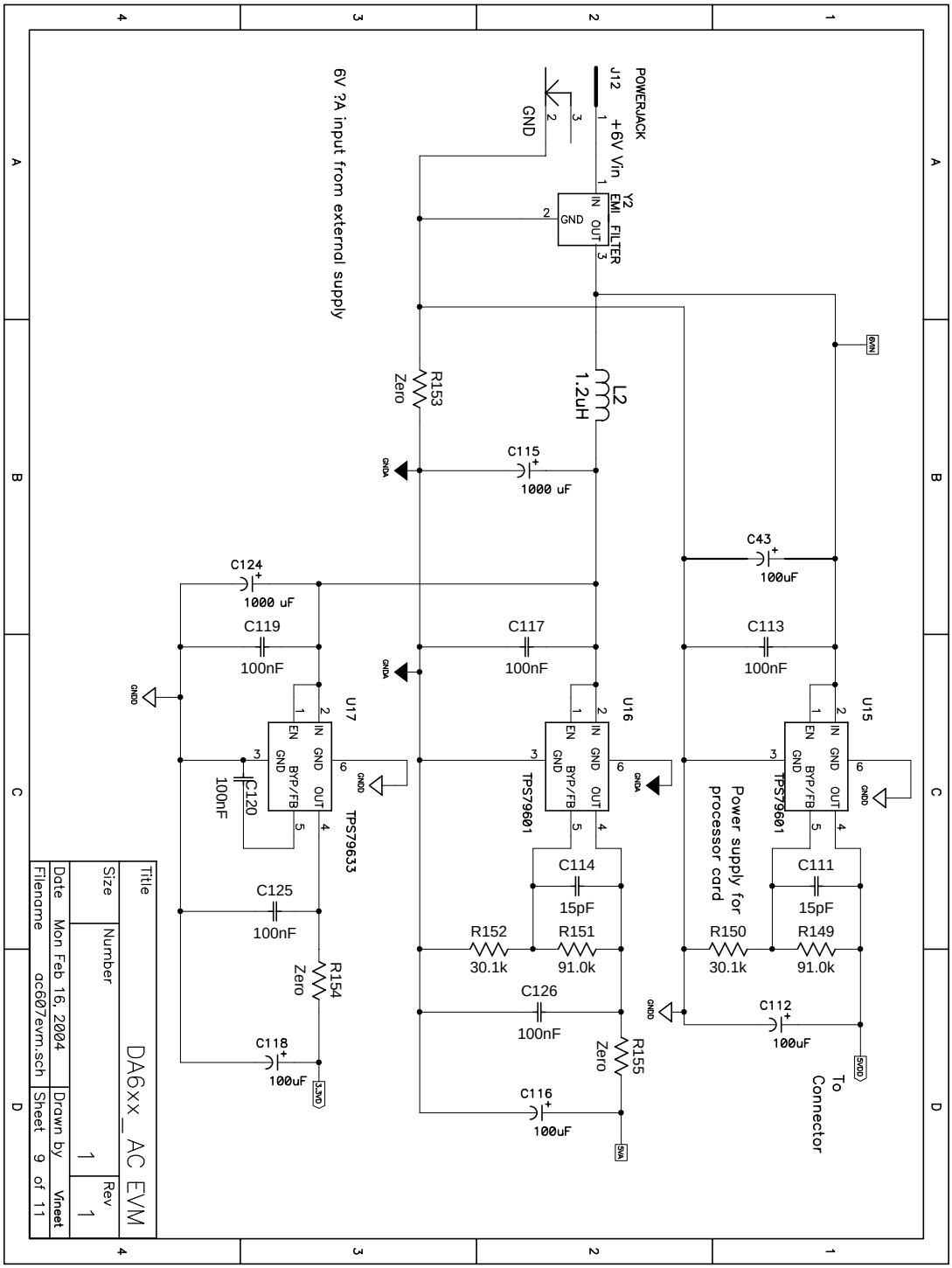


FIGURE 2-11 **AC607 Schematic Diagram: Connectors**

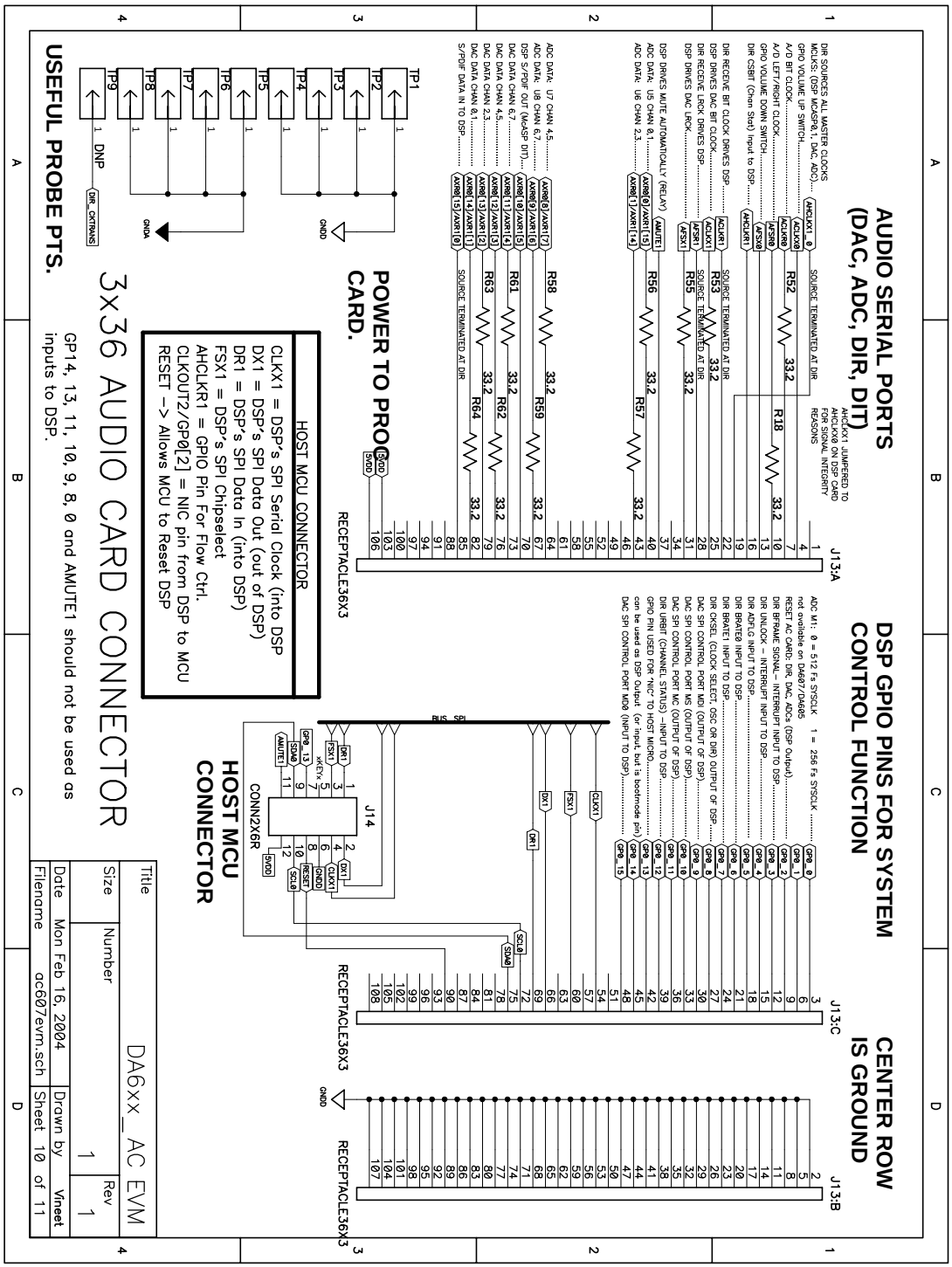
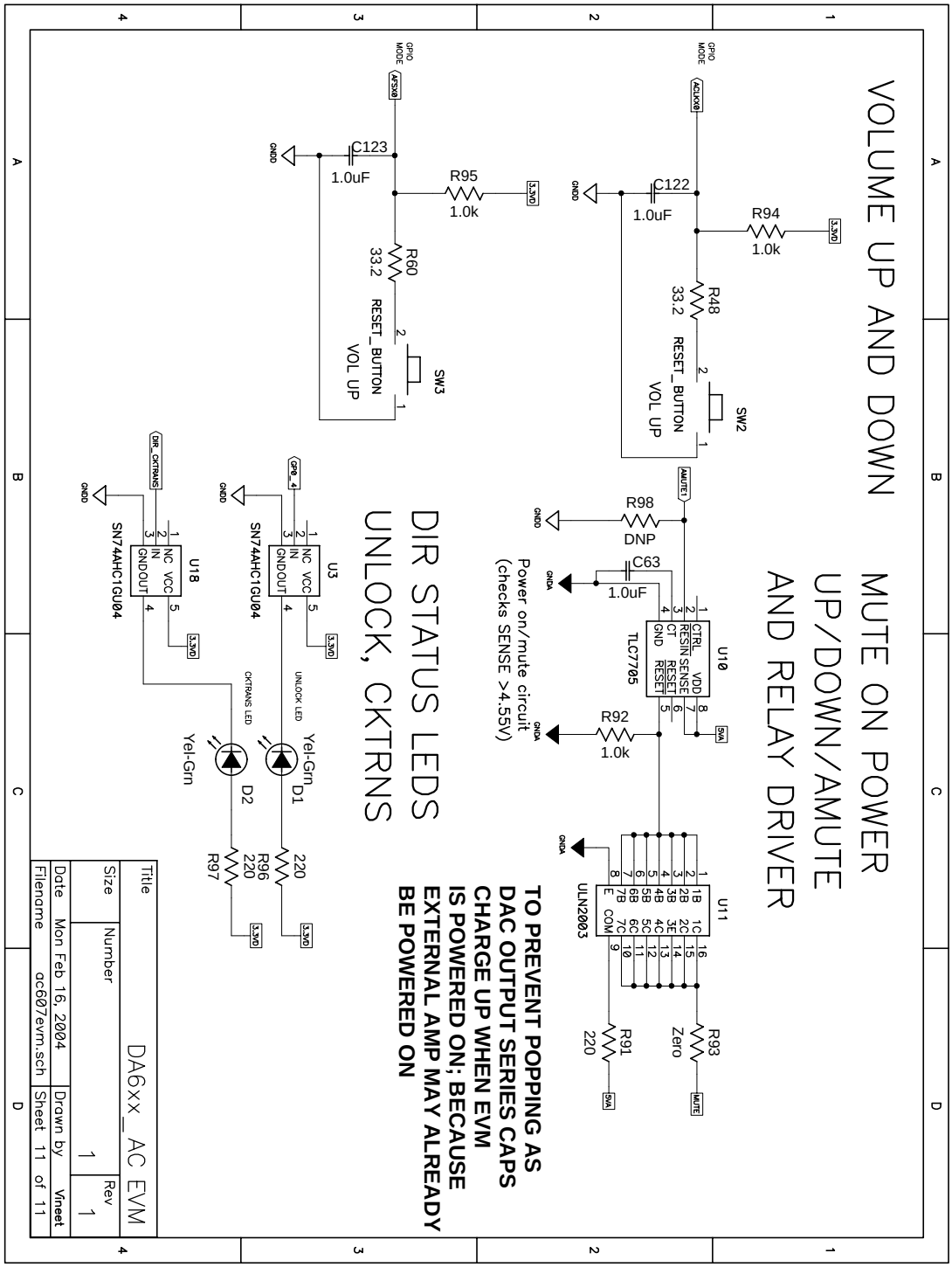


FIGURE 2-12 **AC607 Schematic Diagram: User Interface**



CHAPTER 3 *DA601 Processor Card (PC601) Details*

This chapter shows the details of the DA601 Processor Card (PC601) daughterboard.

The DA6XXEVM Data Sheet is available at:
http://www.mds.com/Products/product.asp?prod=DA%2D60x%2DKIT

See Appendix A for links to web pages containing data sheets for the devices discussed in this section.

3.1 DA601 Processor Card (PC601) Major Components

Figure 1-1, “DA6XXEVM Block Diagram---Functional Overview,” on page 3, gives an overview of the circuitry on the PC601. The location, name and brief description of the major parts on the PC601 are shown below in Figure 3-1 and Table 3-1. Schematic diagrams are provided in “PC601 Schematic Diagrams” on page 30.

FIGURE 3-1 **DA601 Processor Card (PC601) Photograph**

Photo not available.
See Figure 4-1.

TABLE 3-1 **DA601 Processor Card (PC601) Major Parts Description**

Number	Part Name	Description
1	JTAG Interface	Connects to JTAG emulator.
2	Audio Card Interface	Connector for attaching to the AC6XX board.
3	DSP Reset Button and LED	Button for reset of DSP; Reset LED Indicator.
4	Switches	Boot Mode configuration DIP switches. See Section 3.2 for more information.
5 (2x)	REG1117F4 / REG1117FA	Voltage Regulator. See "REG1117A Data Sheet" on page 66.

Number	Part Name	Description
6	HCU04	Hex Inverter. See “HCU04 Data Sheet” on page 66.
7	DA601 DSP	The main event. See “TMS320DA607 Data Sheet” on page 66.
8	AT29LV020	256 Kbytes Parallel Flash memory. See “AT29LV020 Data Sheet” on page 66.
9	K4S641632H	1 Mwords x 4 banks x 16-bit synchronous dynamic RAM. See “K4S641632H Data Sheet” on page 67.
10	AT25HP512	64 Kbytes SPI Bus Serial EEPROM. See “AT25HP512 Data Sheet” on page 67.

3.2 DIP Switches

Figure 3-1 shows the location of the DIP switches. Switches 1 through 6 are used to configure the boot modes and configurations used by the bootloader ROM program. See Section 3.3 for more information on Boot configuration.

3.3 DSP Boot Modes

Before getting into the details, Table 3-2 is a simple table that shows the factory default setting for Flash Boot and JTAG Boot. Most of the time, you can simply use this setting and not worry about the details in the following subsection.

Power-on without an emulator connected to the JTAG Interface shown in Figure 3-1 will result in Flash Boot of the DA6XXEVM from the Flash Memory on the PC601. This runs a factory-installed “Listening Test” program as described in the Quick Start Guide on page QSG-1.

Note: *Power-on with an XDS JTAG emulator connected, but Texas Instruments Code Composer Studio (CCS) not running, will produce the same result. Starting CCS thereafter will halt the CPU, stopping the Listening Test program.*

For software development work, you will want to connect an XDS JTAG Emulator to the JTAG Interface, and thereby use JTAG Boot under the control of Texas Instruments Code Composer Studio (CCS).

TABLE 3-2 DIP Switches: Default Setting for Flash and JTAG Boot

DIP Switch Setting Switches 1,5,6	Description
off, off, off[a]	Flash or JTAG Boot

a. “Off” represents a closed switch. “On” represents an open switch.

3.3.1 DA601 DSP Device Boot Modes

Standard ROM versions of the DA601 contain a software-based bootloader. This bootloader supports various boot configurations as described in Table 39 of the TMS320DA601 Data Sheet. The specific boot mode that the software-based bootloader uses is determined by `GP0[0]`, `AMUTE1`, and `GP0[11:9]`. The signal assignment of the DIP switches on the PC601 card is described in Table 3-3. Refer to “TMS320DA601 Data Sheet” on page 67 to find detailed information regarding the default boot mode selected (DIP Switches: Default Setting for Flash and JTAG Boot) and alternate boot modes.

The “On” position indicated on the DIP switches represents a closed switch. A closed switch presents a logic low (0) to the corresponding DA601 input pin---all except for switch 5, which has the reverse polarity: closed is a logic high (1). See “Boot Mode Switch” circuitry illustrated in Figure 3-6, “PC601 Schematic Diagram: Connectors,” on page 35 for more detail.

Note: The polarity reversal for switch 5 is due to the corresponding DA601 input pin having an internal pulldown resisitor, and is not an inconsistency in the design of the PC601 card.

TABLE 3-3 DIP Switches: Signal Assignment

DIP Switch Number	Signal
1	GP0[13]
2	GP0[11]
3	GP0[10]
4	GP0[9]
5	GP0[0][a]
6	AMUTE1

a. The polarity of Switch 5 is reversed relative to the remainder of switches.

Given the above, the DIP swiches default setting establishes the configuration listed in Table 3-4.

TABLE 3-4 **DIP Swiches: Default DA601 Bootloader Configuration**

PC601 DIP Switches 1,2,3,4,5,6	Input Level DA601: GP0[13]	Input Level DA601: GP0[0], AMUTE1	Input Level DA601: GP0[11:9]	DA601 Bootloader Configuration	External Device
off,off,off,off,off,off	1	0,1	1,1,1[a]	8-bit Asynch. ROM	Flash

a. GP0[11:9] are “Don’t care” for this bootloader configuration.

3.4 JTAG Interface (Emulation Connector)

To access the JTAG port on the DSP:

- Connect a JTAG emulator (XDS) directly to the JTAG connector on the PC601. This port is known as the high-speed JTAG port. The JTAG connector is shown in Figure 3-1 on page 26.

For detailed instructions on connecting a Texas Instruments XDS to the PC601 and configuring Code Composer Studio to use the JTAG emulator, refer to Appendix B.

3.5 Serial EEPROM Memory

64 Kbytes of SPI Serial EEPROM are intended for configuration information. For device details, see “AT25HP512 Data Sheet” on page 67.

3.6 Parallel “Boot” Flash Memory

The parallel flash is used for booting in standalone mode. This is often referred to as “Boot Flash” or “Flash Boot Memory.” See Table 3-2 for necessary switch settings to boot from Flash.

Note: *See Appendix C (Flash Boot) for more information on the Flash Boot process. For details of how to program flash memory, see “Field Upgrade” on page 81 in Appendix B (Code Composer Studio for DA6XXEVM).*

Parallel flash memory is implemented using a AT29LV020 device (U4 in PC601 Schematic Diagram: Memory). The base address for Flash memory is `0x90000000`.

3.7 PC601 Schematic Diagrams

The following schematics are provided herein for convenience. It is advised that you view the schematic diagrams provided on the "DA6XXEVM Hardware Distribution CD," as it may contain updated versions. See directory `/doc/Schematics/PC601`.

- PC601 Schematic Diagram: Title
- PC601 Schematic Diagram: DSP
- PC601 Schematic Diagram: Memory
- PC601 Schematic Diagram: Power
- PC601 Schematic Diagram: Connectors

FIGURE 3-2 PC601 Schematic Diagram: Title

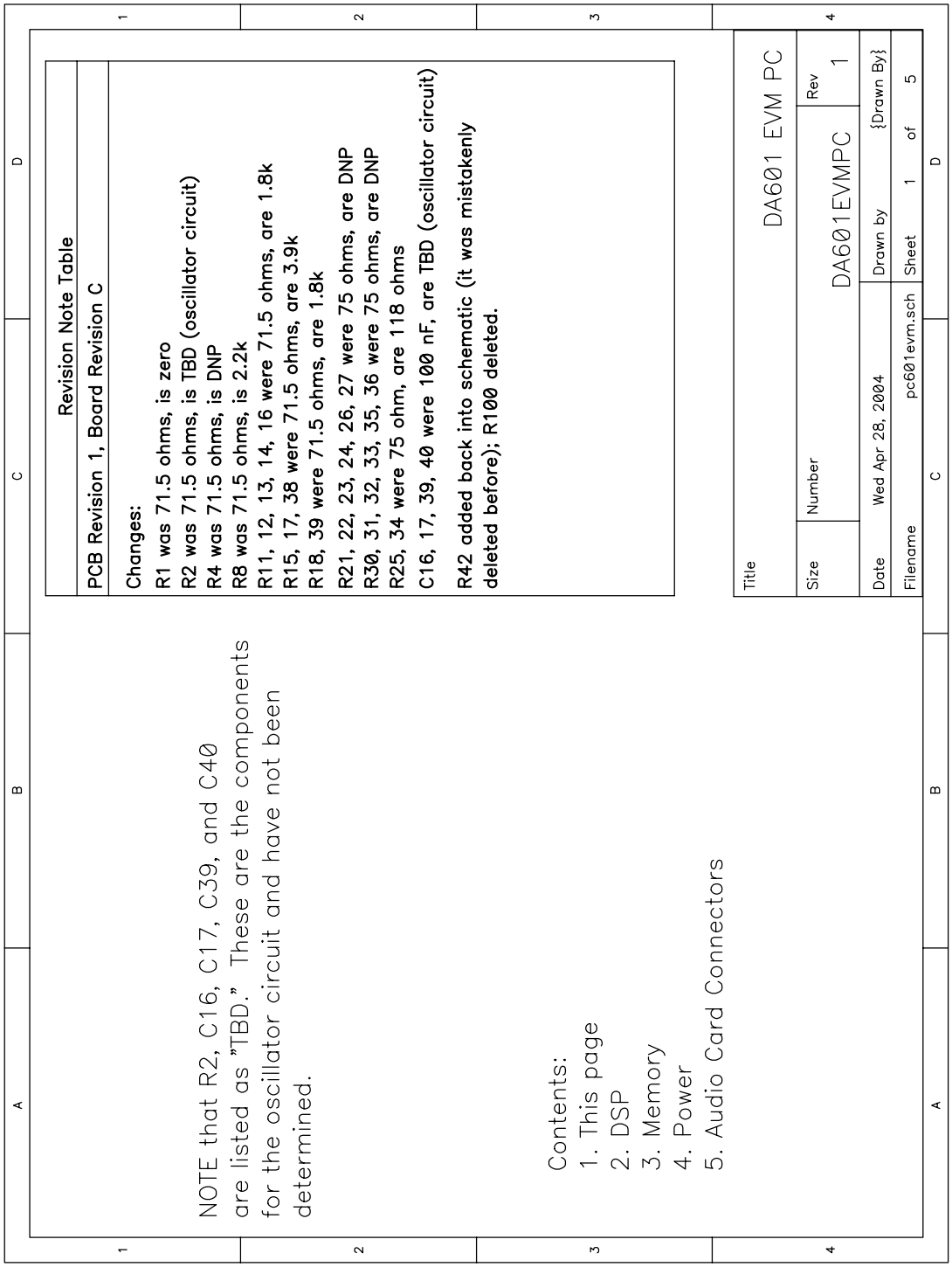


FIGURE 3-3 PC601 Schematic Diagram: DSP

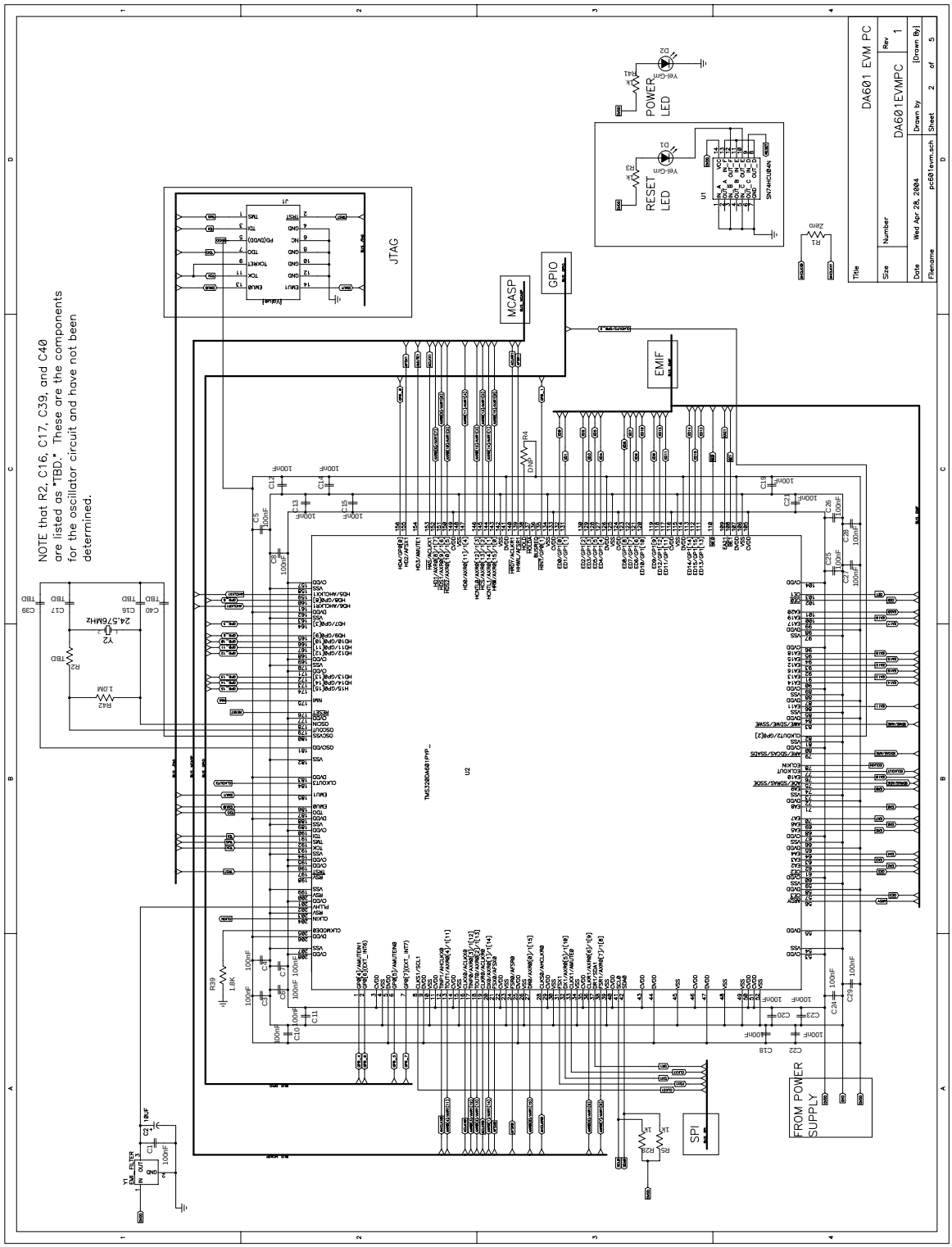


FIGURE 3-4 **PC601 Schematic Diagram: Memory**

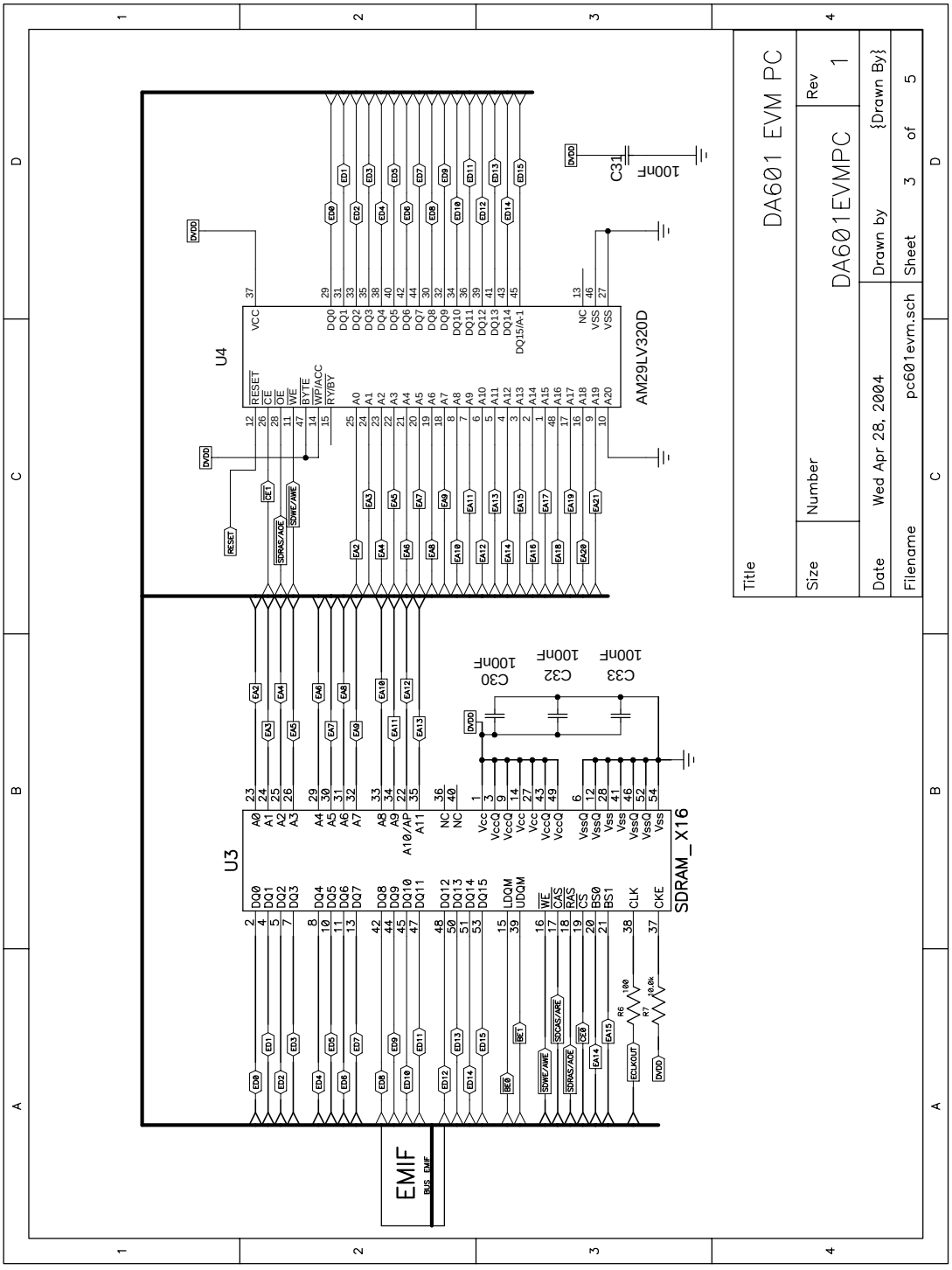


FIGURE 3-5 PC601 Schematic Diagram: Power

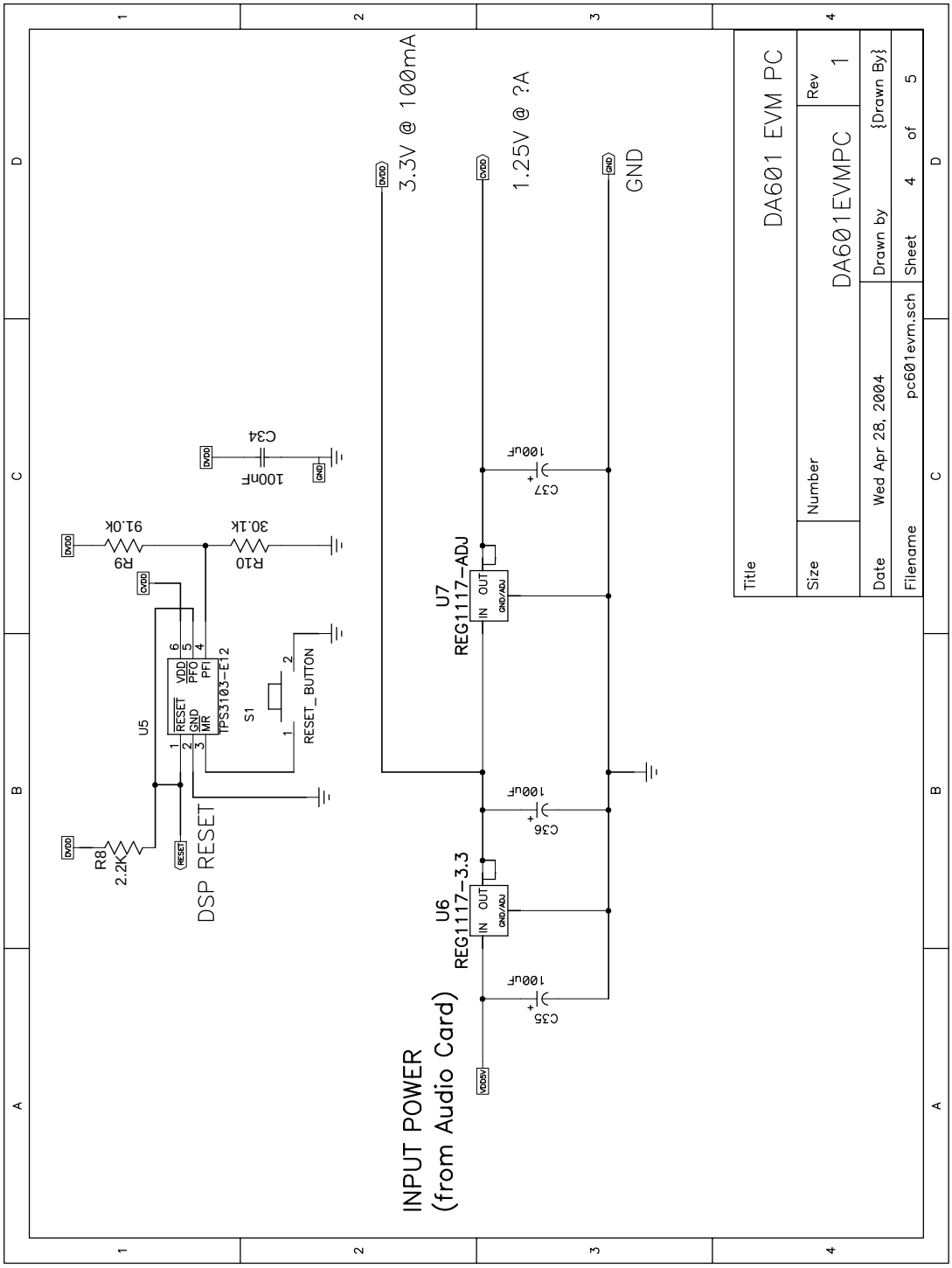
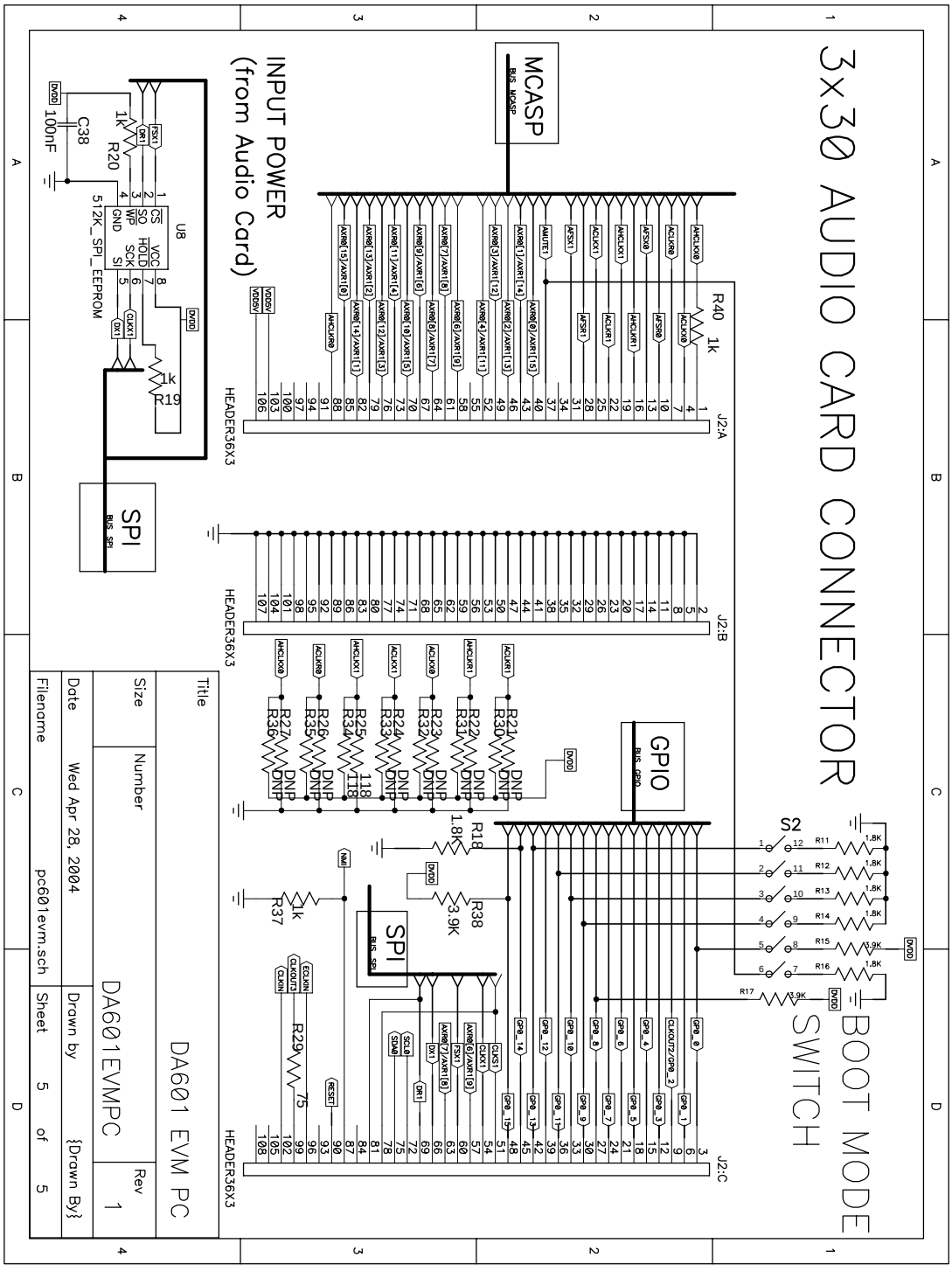


FIGURE 3-6 **PC601 Schematic Diagram: Connectors**



CHAPTER 4 *DA607 Processor Card (PC607) Details*

This chapter shows the details of the DA607 Processor Card (PC607) daughterboard.

The DA6XXEVM Data Sheet is available at:
http://www.mds.com/Products/product.asp?prod=DA%2D60x%2DKIT

See Appendix A for links to web pages containing data sheets for the devices discussed in this section.

4.1 DA607 Processor Card (PC607) Major Components

Figure 1-1, “DA6XXEVM Block Diagram---Functional Overview,” on page 3, gives an overview of the circuitry on the PC607. The location, name and brief description of the major parts on the PC607 are shown below in Figure 4-1 and Table 4-1. Schematic diagrams are provided in “PC607 Schematic Diagrams” on page 42.

FIGURE 4-1 **DA607 Processor Card (PC607) Photograph**

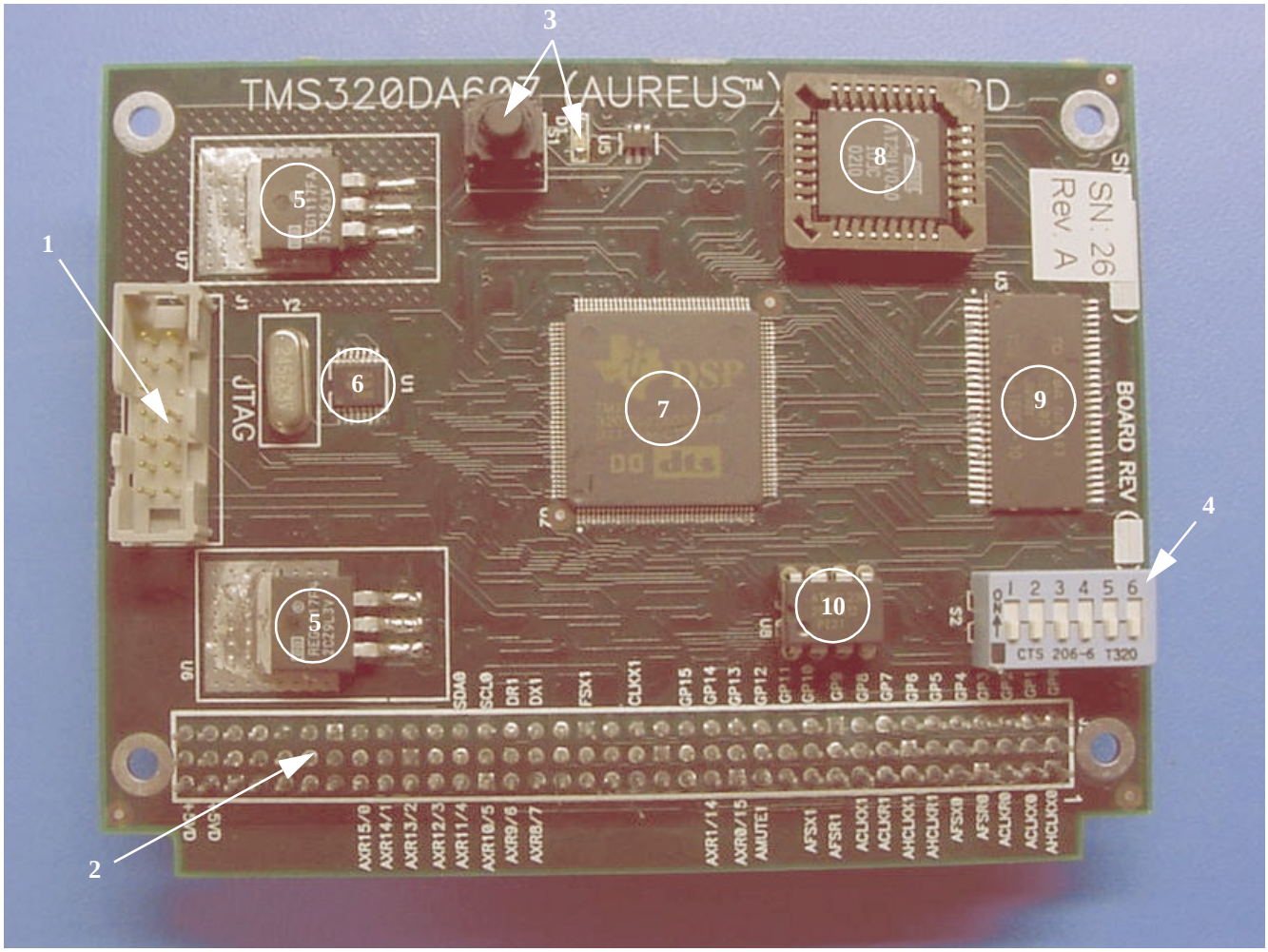


TABLE 4-1 **DA607 Processor Card (PC607) Major Parts Description**

Number	Part Name	Description
1	JTAG Interface	Connects to JTAG emulator.
2	Audio Card Interface	Connector for attaching to the AC6XX board.
3	DSP Reset Button and LED	Button for reset of DSP; Reset LED Indicator.
4	Switches	Boot Mode configuration DIP switches. See Section 4.2 for more information.
5 (2x)	REG1117F4 / REG1117FA	Voltage Regulator. See “REG1117A Data Sheet” on page 66.

Number	Part Name	Description
6	HCU04	Hex Inverter. See “HCU04 Data Sheet” on page 66.
7	DA607 DSP	The main event. See “TMS320DA607 Data Sheet” on page 66.
8	AT29LV020	256 Kbytes Parallel Flash memory. See “AT29LV020 Data Sheet” on page 66.
9	K4S641632H	4 Mbytes (accessible) Synchronous Dynamic RAM.[a] See “K4S641632H Data Sheet” on page 67.
10	AT25HP512	64 Kbytes SPI Bus Serial EEPROM. See “AT25HP512 Data Sheet” on page 67.

a. During initialization using byte-wide SDRAM, the DSP programs the SDRAM for a burst size of 4 (initializing only one access per burst in most cases would be very inefficient). This means that every read/write access is always 4 cycles. This is performed due to overhead associated with the static part of SDRAM; due to latency, it may take 3 to 4 cycles before the first data is available. For example, first data may require 40 ns, two data within 50 ns, three data within 60 ns, etc.; burst frequency needs to approach 100MHz. In the case of writing only one byte, the BEO signal is used to mask off unwanted writes during the burst of 4 writes.

4.2 DIP Switches

Figure 4-1 shows the location of the DIP switches. Switches 1 through 6 are used to configure the boot modes and configurations used by the bootloader ROM program. See Section 4.3 for more information on Boot configuration.

4.3 DSP Boot Modes

Before getting into the details, Table 4-2 is a simple table that shows the factory default setting for Flash Boot and JTAG Boot. Most of the time, you can simply use this setting and not worry about the details in the following subsection.

Power-on without an emulator connected to the JTAG Interface shown in Figure 4-1 will result in Flash Boot of the DA6XXEVM from the Flash Memory on the PC607. This runs a factory-installed “Listening Test” program as described in the Quick Start Guide near the beginning of this manual.

Note: *Power-on with an XDS JTAG emulator connected, but Texas Instruments Code Composer Studio (CCS) not running, will produce the same result. Starting CCS thereafter will halt the CPU, stopping the Listening Test program.*

For software development work, you will want to connect an XDS JTAG Emulator to the JTAG Interface, and thereby use JTAG Boot under the control of Texas Instruments Code Composer Studio (CCS).

TABLE 4-2 **DIP Switches: Default Setting for Flash and JTAG Boot**

DIP Switch Setting Switches 1,5,6	Description
off, off, off[a]	Flash or JTAG Boot

a. “Off” represents a closed switch. “On” represents an open switch.

4.3.1 DA607 DSP Device Boot Modes

Standard ROM versions of the DA607 contain a software-based bootloader. This bootloader supports various boot configurations as described in Table 39 of the TMS320DA607 Data Sheet. The specific boot mode that the software-based bootloader uses is determined by `GP0[0]`, `AMUTE1`, and `GP0[11:9]`. The signal assignment of the DIP switches on the PC607 card is described in Table 4-3. Refer to “TMS320DA607 Data Sheet” on page 66 to find detailed information regarding the default boot mode selected (DIP Switches: Default Setting for Flash and JTAG Boot) and alternate boot modes.

The “On” position indicated on the DIP switches represents a closed switch. A closed switch presents a logic low (0) to the corresponding DA607 input pin---all except for switch 5, which has the reverse polarity: closed is a logic high (1). See “Boot Mode Switch” circuitry illustrated in Figure 4-6, “PC607 Schematic Diagram: Connectors,” on page 47 for more detail.

Note: The polarity reversal for switch 5 is due to the corresponding DA607 input pin having an internal pulldown resisitor, and is not an inconsistency in the design of the PC607 card.

TABLE 4-3 **DIP Switches: Signal Assignment**

DIP Switch Number	Signal
1	GP0[13]
2	GP0[11]
3	GP0[10]
4	GP0[9]
5	GP0[0][a]
6	AMUTE1

a. The polarity of Switch 5 is reversed relative to the remainder of switches.

Given the above, the DIP swiches default setting establishes the configuration listed in Table 4-4.

TABLE 4-4 **DIP Swiches: Default DA607 Bootloader Configuration**

PC607 DIP Switches 1,2,3,4,5,6	Input Level DA607: GP0[13]	Input Level DA607: GP0[0], AMUTE1	Input Level DA607: GP0[11:9]	DA607 Bootloader Configuration	External Device
off,off,off,off,off,off	1	0,1	1,1,1[a]	8-bit Asynch. ROM	Flash

a. GP0[11:9] are “Don’t care” for this bootloader configuration.

4.4 JTAG Interface (Emulation Connector)

To access the JTAG port on the DSP:

- Connect a JTAG emulator (XDS) directly to the JTAG connector on the PC607. This port is known as the high-speed JTAG port. The JTAG connector is shown in Figure 4-1 on page 38.

For detailed instructions on connecting a Texas Instruments XDS to the PC607 and configuring Code Composer Studio to use the JTAG emulator, refer to Appendix B.

4.5 Serial EEPROM Memory

64 KBytes of SPI Serial EEPROM are intended for configuration information. For device details, see “AT25HP512 Data Sheet” on page 67.

4.6 Parallel “Boot” Flash Memory

The parallel flash is used for booting in standalone mode. This is often referred to as “Boot Flash” or “Flash Boot Memory.” See Table 4-2 for necessary switch settings to boot from Flash.

Note: *See Appendix C (Flash Boot) for more information on the Flash Boot process. For details of how to program flash memory, see “Field Upgrade” on page 81 in Appendix B (Code Composer Studio for DA6XXEVM).*

Parallel flash memory is implemented using a AT29LV020 device (U4 in PC607 Schematic Diagram: Memory). As described in the AT29LV020 Data Sheet, this 2 Mbit device is organized as 1024 sectors of 256 bytes/sector. The base address for Flash memory is `0x90000000`. Address lines `A8` to `A17` specify the sector address; `A0` to `A7` specify the byte address within the sector.

4.7 PC607 Schematic Diagrams

The following schematics are provided herein for convenience. It is advised that you view the schematic diagrams provided on the "DA6XXEVM Hardware Distribution CD," as it may contain updated versions. See directory `/doc/Schematics/PC607`.

- PC607 Schematic Diagram: Title
- PC607 Schematic Diagram: DSP
- PC607 Schematic Diagram: Memory
- PC607 Schematic Diagram: Power
- PC607 Schematic Diagram: Connectors

FIGURE 4-2 PC607 Schematic Diagram: Title

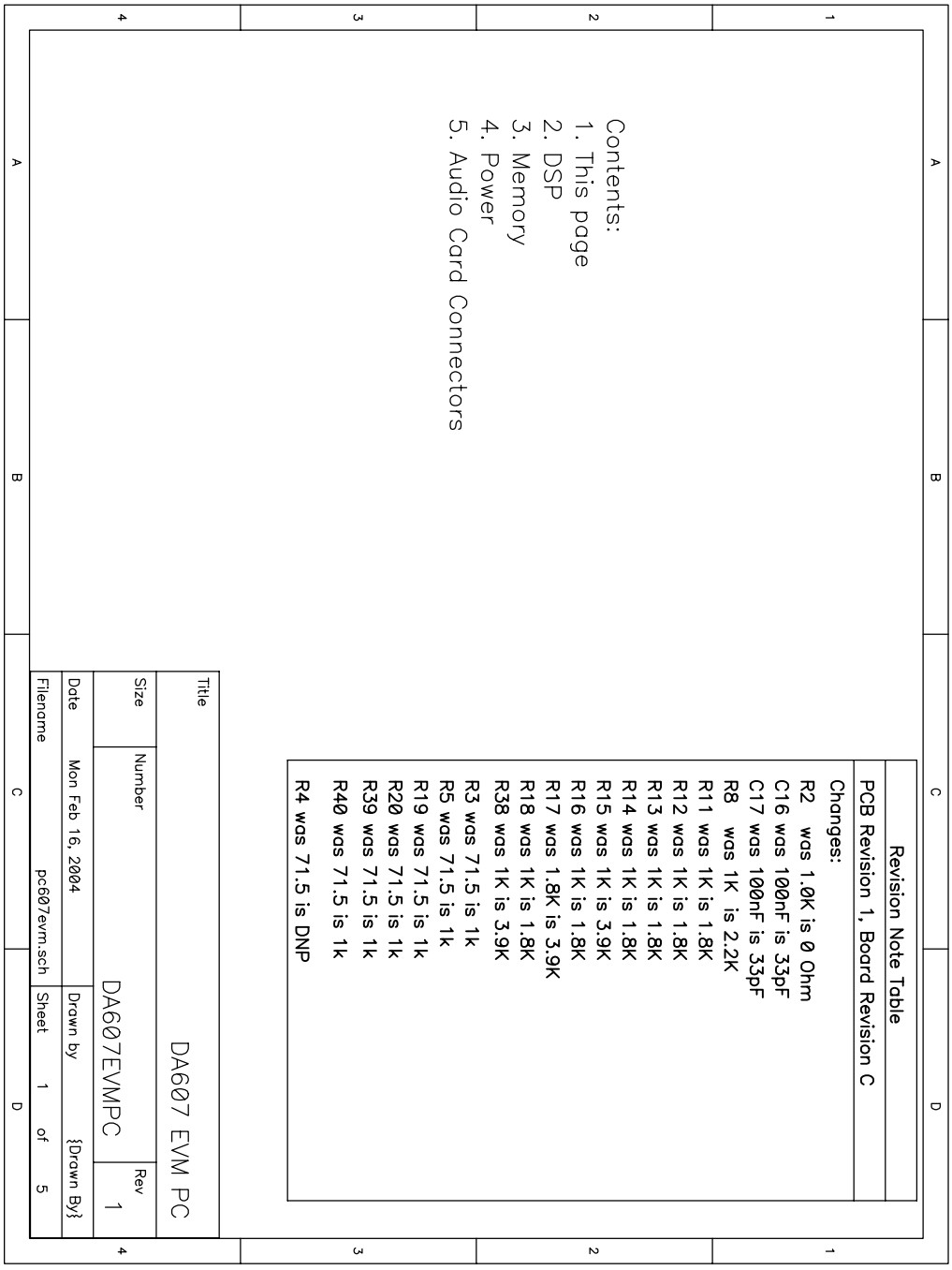


FIGURE 4-3 PC607 Schematic Diagram: DSP

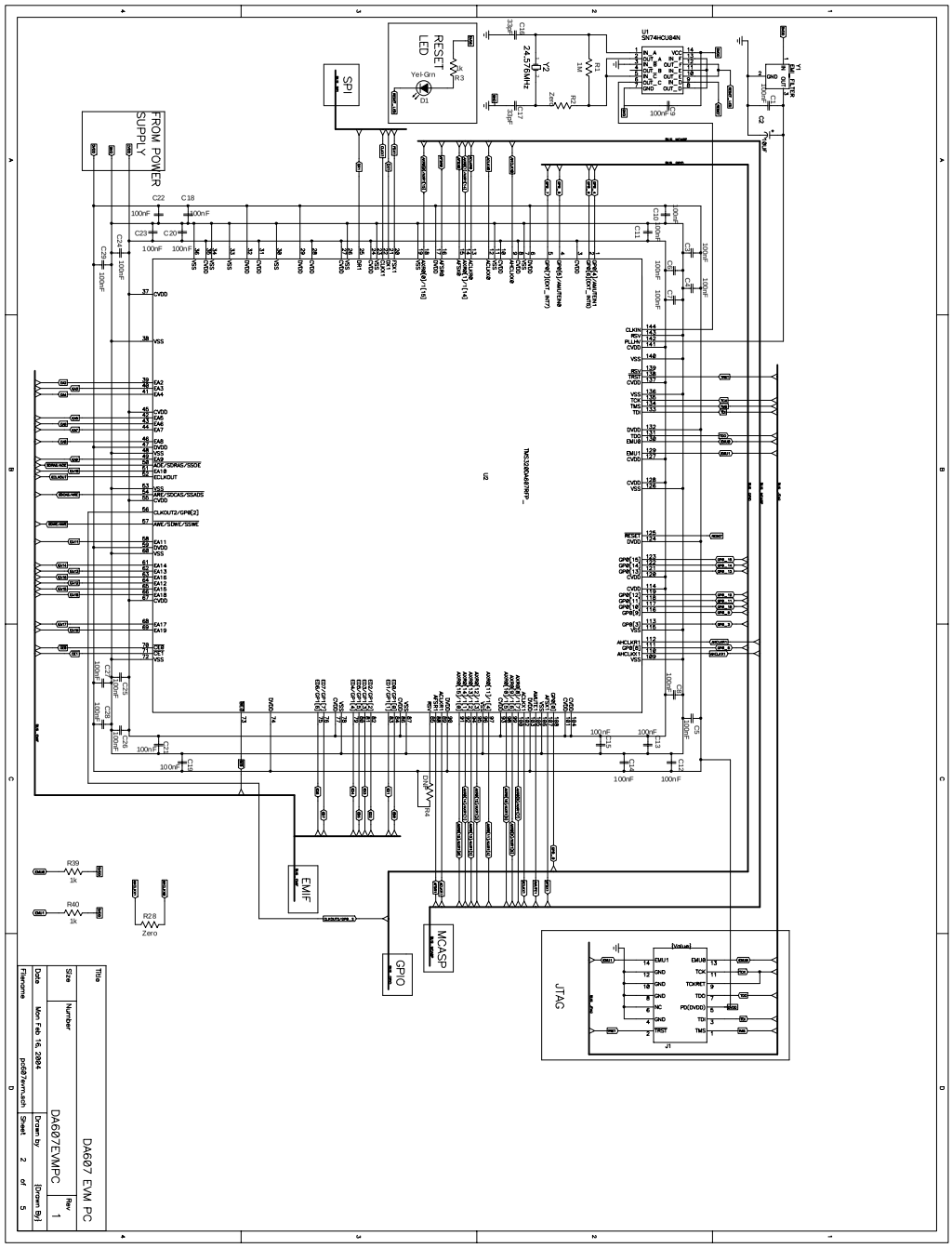


FIGURE 4-4 **PC607 Schematic Diagram: Memory**

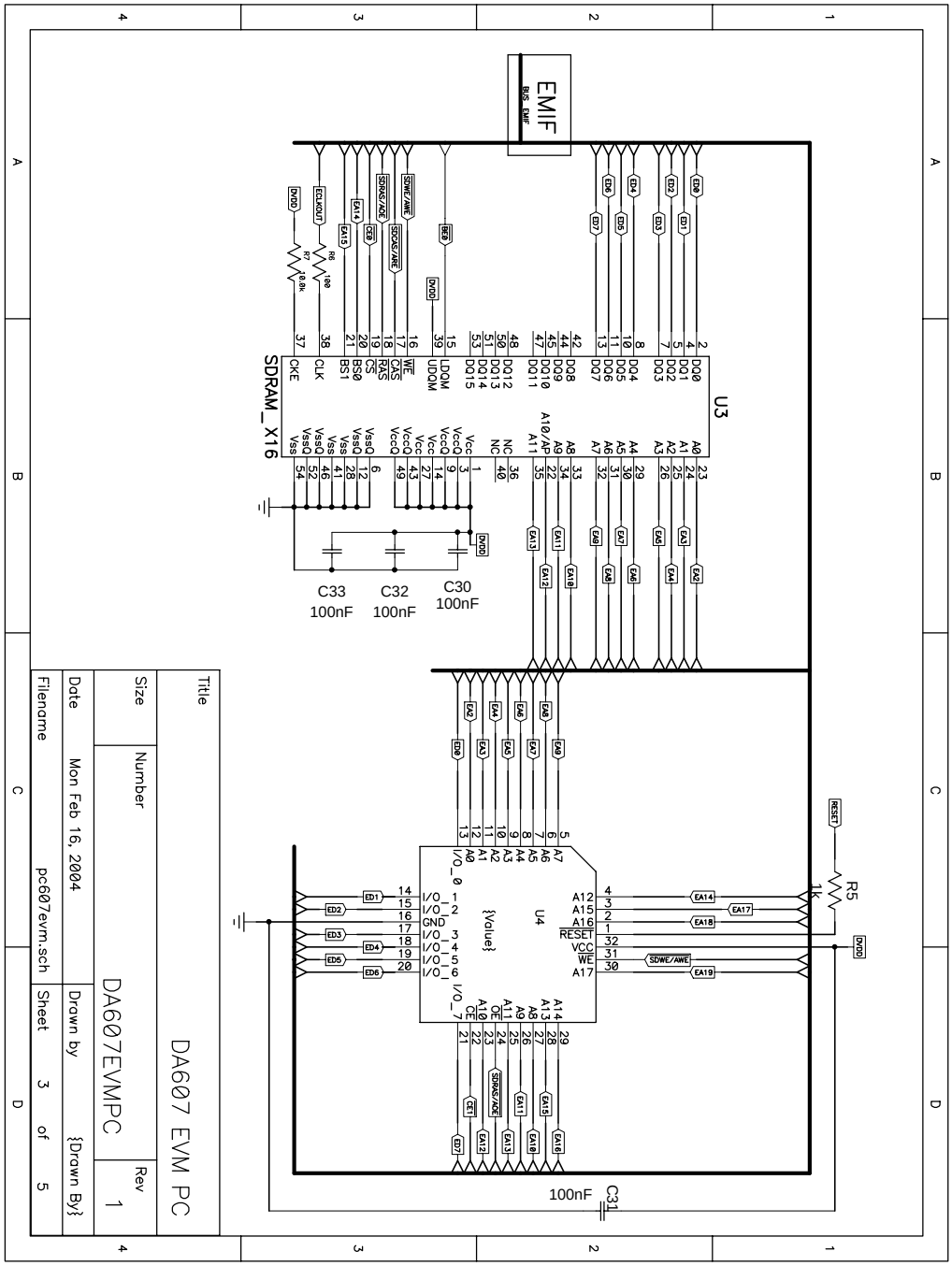


FIGURE 4-5 PC607 Schematic Diagram: Power

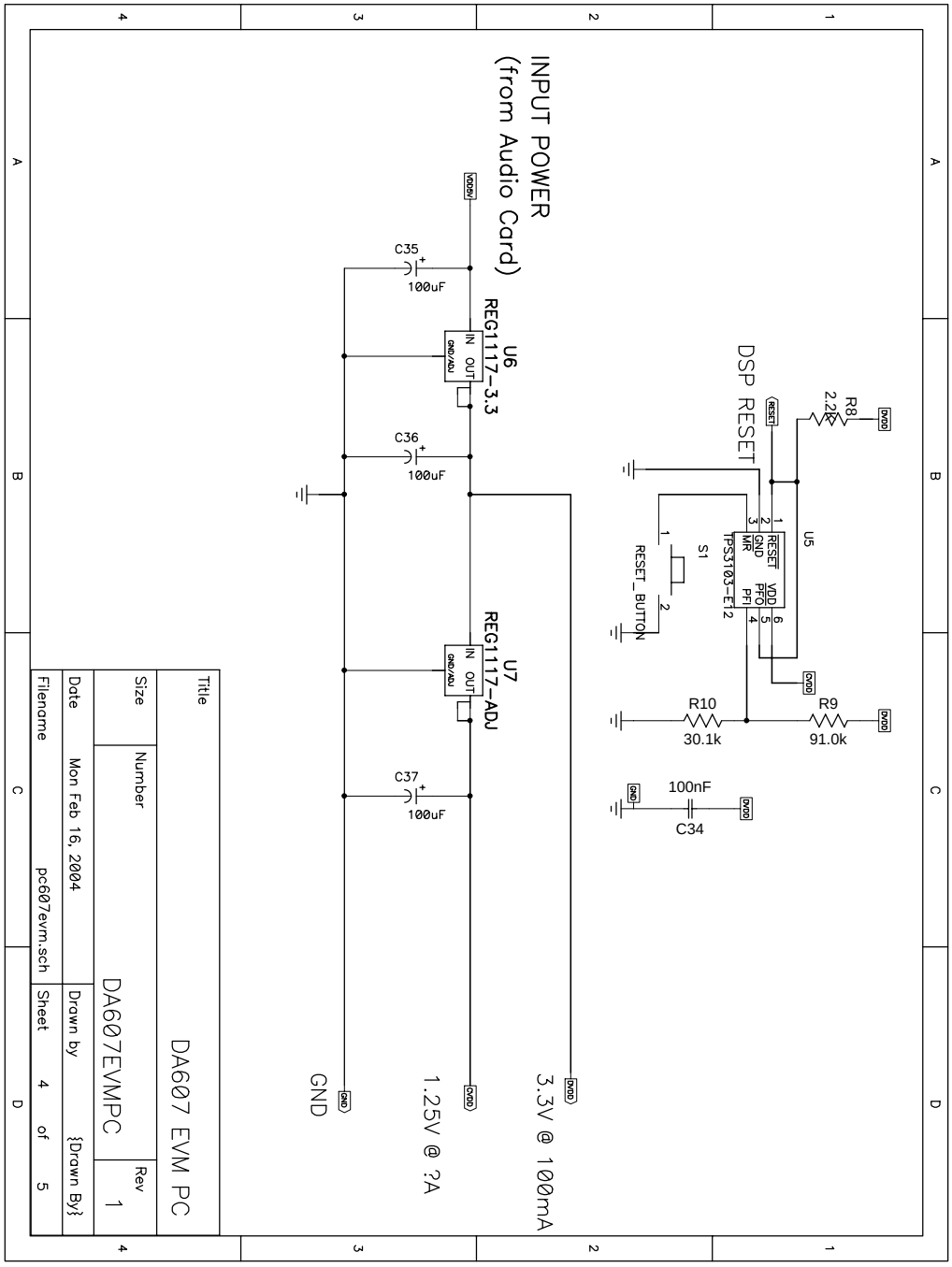
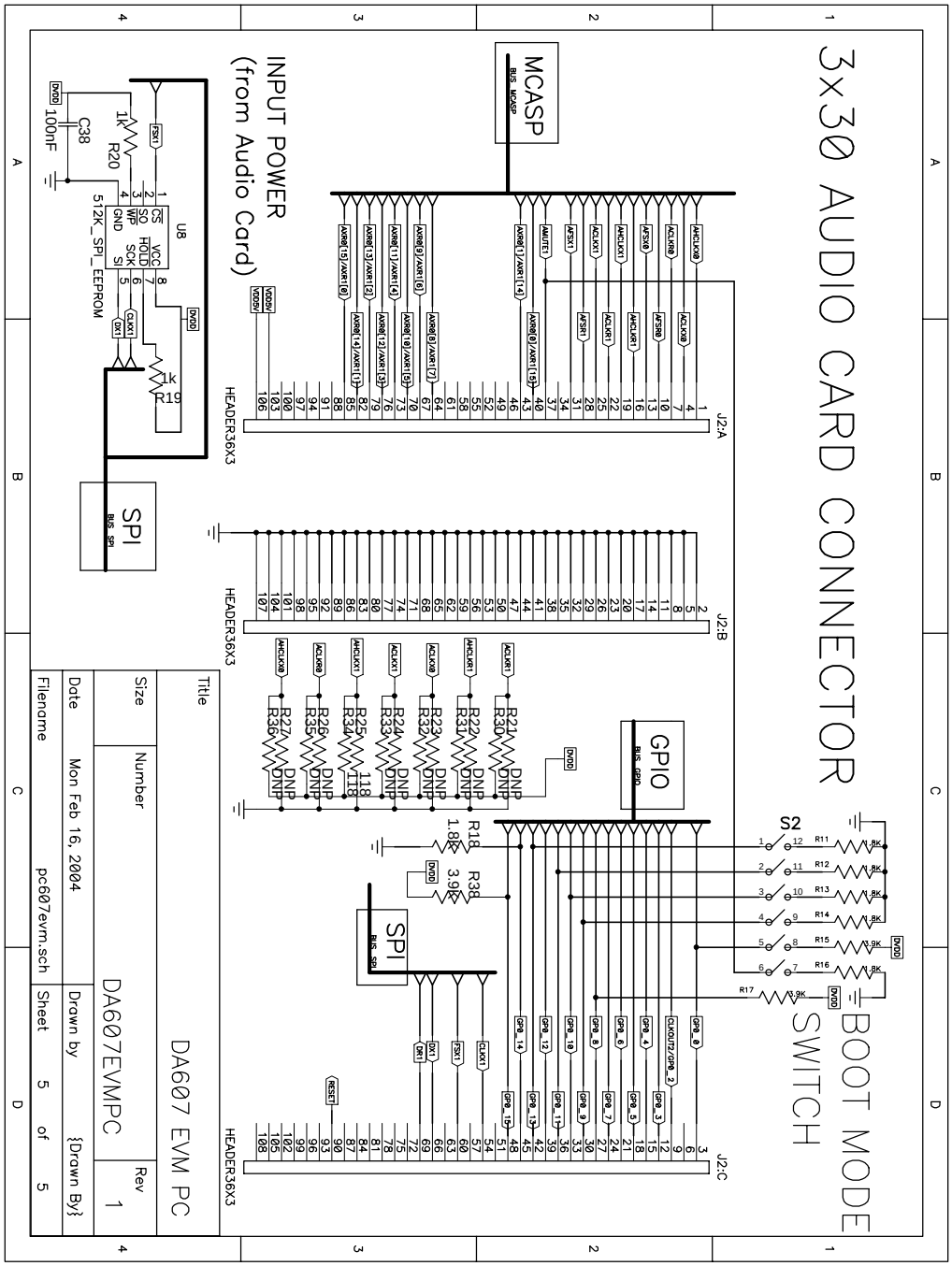


FIGURE 4-6 **PC607 Schematic Diagram: Connectors**



CHAPTER 5 DA6XX SPI/I^{2}C Card (SPI Dongle) Details

This chapter shows the details of the DA6XX SPI/I^{2}C Card (SPI Dongle) daughterboard.

See Appendix A for links to web pages containing data sheets for the devices discussed in this section.

Note: When using the SPI Dongle, the serial EEPROM device must be removed from the Processor Card as they share the same serial peripheral interface.

Note: I2C bus pullup resistors `R5` *and* `R28` *(1 kohm) are not populated on the SPI Dongle, as there are already associated pullup resistors on the processor card for these signals (*`SDA0` *and* `SCL0` *respectively; see Figure 5-3, "SPI Dongle Schematic Diagram: DSP," on page 55).*

If a custom processor card is to be used with the SPI Dongle for I^{2}C communications, and the custom processor card has the pullup resistors, `R5` *and* `R28` *should not be populated. If the custom processor card does not have the pullup resistors,* `R5` *and* `R28` *can be populated if needed.*

5.1 DA6XX SPI/I^{2}C Card (SPI Dongle) Major Components

Figure 1-1, "DA6XXEVM Block Diagram---Functional Overview," on page 3, gives an overview of the circuitry on the SPI Dongle. The location, name and brief description of the major parts on the SPI Dongle are shown below in Figure 5-1 and Table 5-1. Schematic diagrams are provided in "SPI Dongle Schematic Diagrams" on page 53.

FIGURE 5-1 DA6XX SPI/I²C Card (SPI Dongle) Photograph

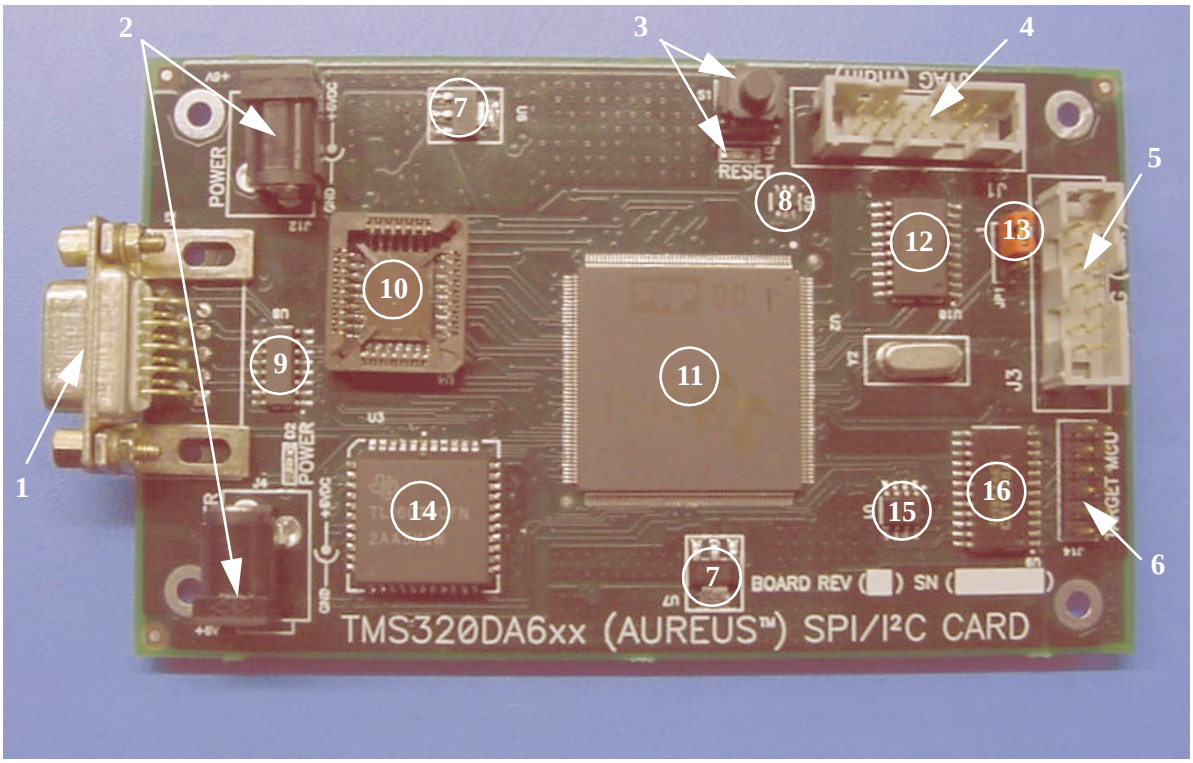


TABLE 5-1 DA6XX SPI/I²C Card (SPI Dongle) Major Components

Number	Part Name	Description
1	DB-9 Connector	Connector for RS-232 interface.
2	Power Connectors	Power connectors for attaching to Power Adapter and to power the DA6XX EVM Audio Card (AC6XX) or target board.
3	DSP Reset Button and LED	Button for resetting the DSP; LED Reset Indicator.
4	JTAG Connector	Connector for main JTAG interface.
5	JTAG Connector	Connector for auxiliary JTAG interface.
6	Host MCU Connector	Connector for target MCU interface.
7 (2x)	REG1117	Voltage Regulator. See “REG1117A Data Sheet” on page 67.
8	TPS3103E12	Supply-Current/Supply-Voltage Supervisory Circuit. See “TPS3103E12 Data Sheet” on page 67.
9	MAX3232ESE	RS-232 Transceiver. See “MAX3232ESE Data Sheet” on page 67.

Number	Part Name	Description
10	AT29LV020	256 Kbytes Parallel Flash memory. See "AT29LV020 Data Sheet" on page 66.
11	DA601 DSP	See "TMS320DA601 Data Sheet" on page 67.
12	SN74LVTH244	Octal Buffer. See "SN74LVTH244A Data Sheet" on page 67.
13	Jumper	JTAG / Debug Mode Configuration: 1. Debug Mode Configuration: Connect JP1-1 to JP1-2 (default setting): Connects Main JTAG data out (J1-7) to DSP JTAG data out (TDO). 2. Parallel Debug Mode Configuration: Connect JP1-2 to JP1-3: Connects Main JTAG data out (J1-7) to Auxiliary JTAG data out (J3-7).
14	TL16C750	Asynchronous Communications Element. See "TL16C750 Data Sheet" on page 67.
15	TLC7705	MicroPower Supply Voltage Supervisor. See "TLC7705 Data Sheet" on page 67.
16	SN74CBTD3861	10-bit FET Bus Switch, See "SN74CBTD3861 Data Sheet" on page 67.

5.2 Setup SPI Dongle Hardware

Perform these procedures provided in the Quick Start Guide:

- Setup Hardware: DA6XXEVM on page QSG-2.
- Setup Hardware: SPI/I^{2}C Card: Debug Mode on page QSG-2, or Setup Hardware: SPI/I^{2}C Card: Parallel Debug Mode on page QSG-4.

5.3 SPI Dongle Operation

Flash memory of the SPI Dongle is factory-programmed with a SPI interface processor program. When performing setup of SPI Dongle hardware as described above, power-on results in the SPI Dongle booting and running this program. Alternately, an I^{2}C interface processor program is also provided which can be programmed into the SPI Dongle's flash memory. SPI/I^{2}C operation and flash programming (Firmware Download) is described in the following sections.

5.3.1 SPI Operation

The SPI interface processor program converts S-Records sent via the serial port of a PC to SP-B16 SPI format. For details about S-Records and SP-B16 protocol, refer to *"Performance Audio Messaging Transport Protocol"* (file `\doc\paf-scp.pdf`, provided on the "DA6XXEVM Hardware Distribution CD").

To perform SPI communications, following this procedure provided in the Quick Start Guide :

- Communicating With Performance Audio System Code on page QSG-8.

5.3.2 I²C Operation

This section is under construction.

5.3.3 Firmware Download

The following sections describe procedures for programming of SPI Interface Processor and I²C Interface Processor[12] programs into the SPI Dongle's flash memory.

5.3.3.1 SPI Interface Processor

Flash memory of the SPI Dongle is factory-programmed with a SPI interface processor program. This same program is provided on "DA6XXEVM Hardware Distribution CD" in file "`xk-ip4_YYMMDD.zip`."
`Note: YYMMDD` is a date code.

To program the SPI Dongle flash memory with the SPI interface processor program, perform the following procedure:

1. On the SPI/I²C Card, set jumper JP1 to connect pins 1 and 2.
   Note: Pin 1 is indicated by the silkscreen on the board.
2. Using the SPI/I²C power cable (the jumper cable, not the DA6XXEVM Power Adapter), connect power jack (J4) on the SPI/I²C Card to the power jack on the Audio Card.
3. Connect your JTAG emulator to the JTAG Main header (J1) on the SPI/I²C card.
4. If connected, disconnect the 14-pin ribbon cable from Target MCU Interface header (J14) on the SPI/I²C Card.
5. If connected, disconnect the JTAG cable from JTAG Aux header (J3) on SPI/I²C Card.
6. Connect the Power Adapter to power jack (J12) on the SPI/I²C Card.
7. Unzip "`xk-ip4_YYMMDD.zip`" into a arbitrary directory.
   Note: The remainder of this procedure uses "`c:\temp`" as the arbitrary directory.
8. Start Code Composer Studio (CCS).

12. The I²C interface processor program is not yet available.

9. In the CCS pull-down menu, select **File / Load GEL...**, then navigate to select:
   "`c:\temp\xk-ip4_YYMMDD\board_e3.gel`"
10. In the CCS pull-down menu, select **Tools / FlashBurn**.
11. In the FlashBurn window, select **File / Open...** then navigate to select:
    "`c:\temp\xk-ip4_YYMMDD\fb_boot.cdd`"
    Note: This results in the loading of the associated "`fbtc.out`" file, with success indicated by a "Connected" icon in the "Target System" sub-window.
12. In the FlashBurn window, select **Program / Erase Flash**.
13. For "File To Burn," select **Browse...** then navigate to select:
    "`c:\temp\xk-ip4_YYMMDD\fl_boot.hex`"
14. Select **Program / Program Flash**.
15. In the FlashBurn window, select **File / Close** to close "`fb_boot.cdd`."
16. In the FlashBurn window, select **File / Open...** then navigate to select:
    "`c:\temp\xk-ip4_YYMMDD\fb_program.cdd`"
17. For "File To Burn," select **Browse...** then navigate to select:
    "`c:\temp\xk-ip4_YYMMDD\euart_spb16.hex`"
18. Select **Program / Program Flash**.

5.3.3.2 I^2C Interface Processor

This section is under construction.

5.4 SPI Dongle Schematic Diagrams

The following schematics are provided herein for convenience. It is advised that you view the schematic diagrams provided on the "DA6XXEVM Hardware Distribution CD," as it may contain updated versions. See directory `/doc/Schematics/SPIDongle`.

- SPI Dongle Schematic Diagram: Title
- SPI Dongle Schematic Diagram: DSP
- SPI Dongle Schematic Diagram: Memory
- SPI Dongle Schematic Diagram: Power
- SPI Dongle Schematic Diagram: Connectors

FIGURE 5-2 SPI Dongle Schematic Diagram: Title

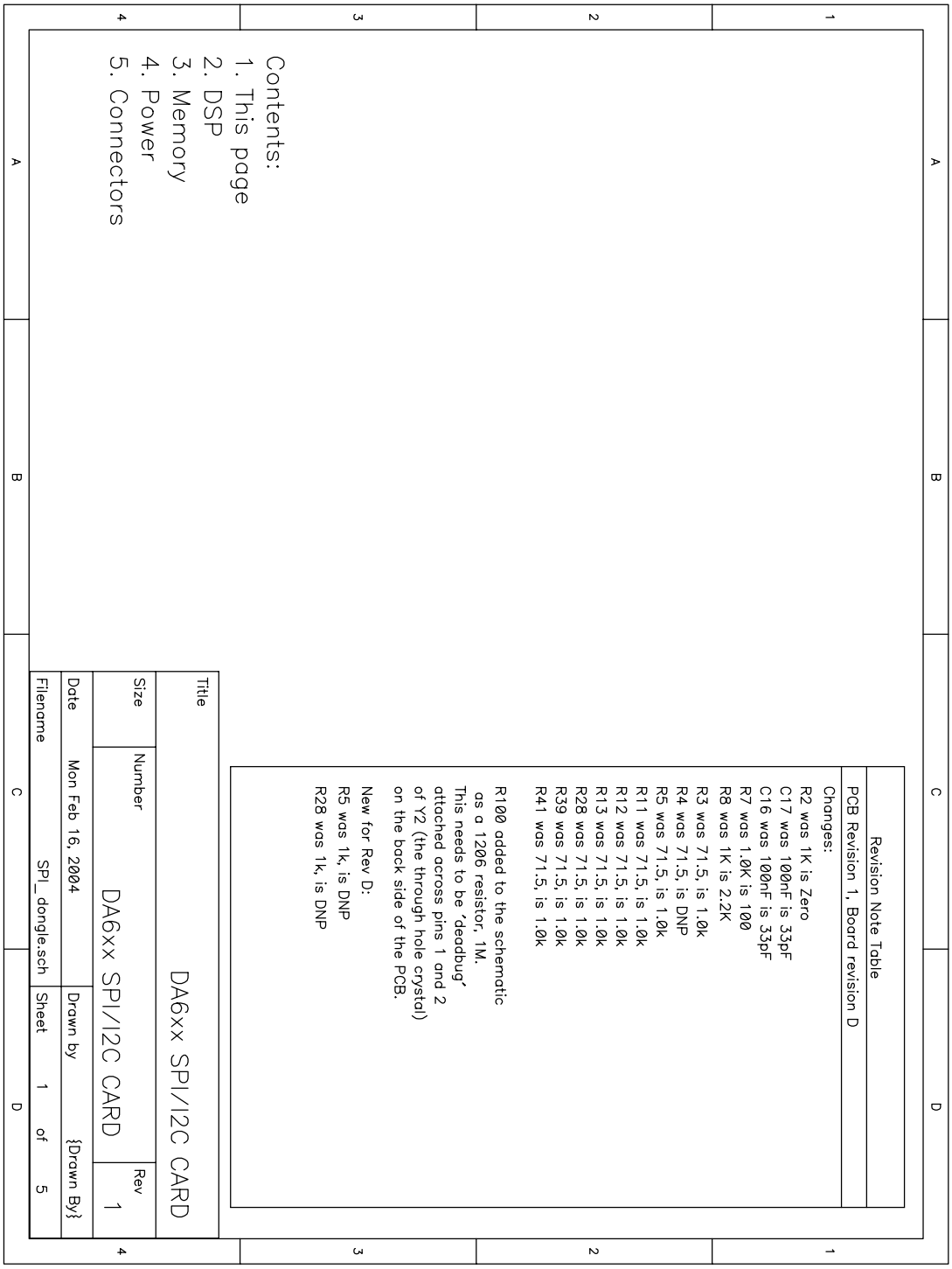


FIGURE 5-3 SPI Dongle Schematic Diagram: DSP

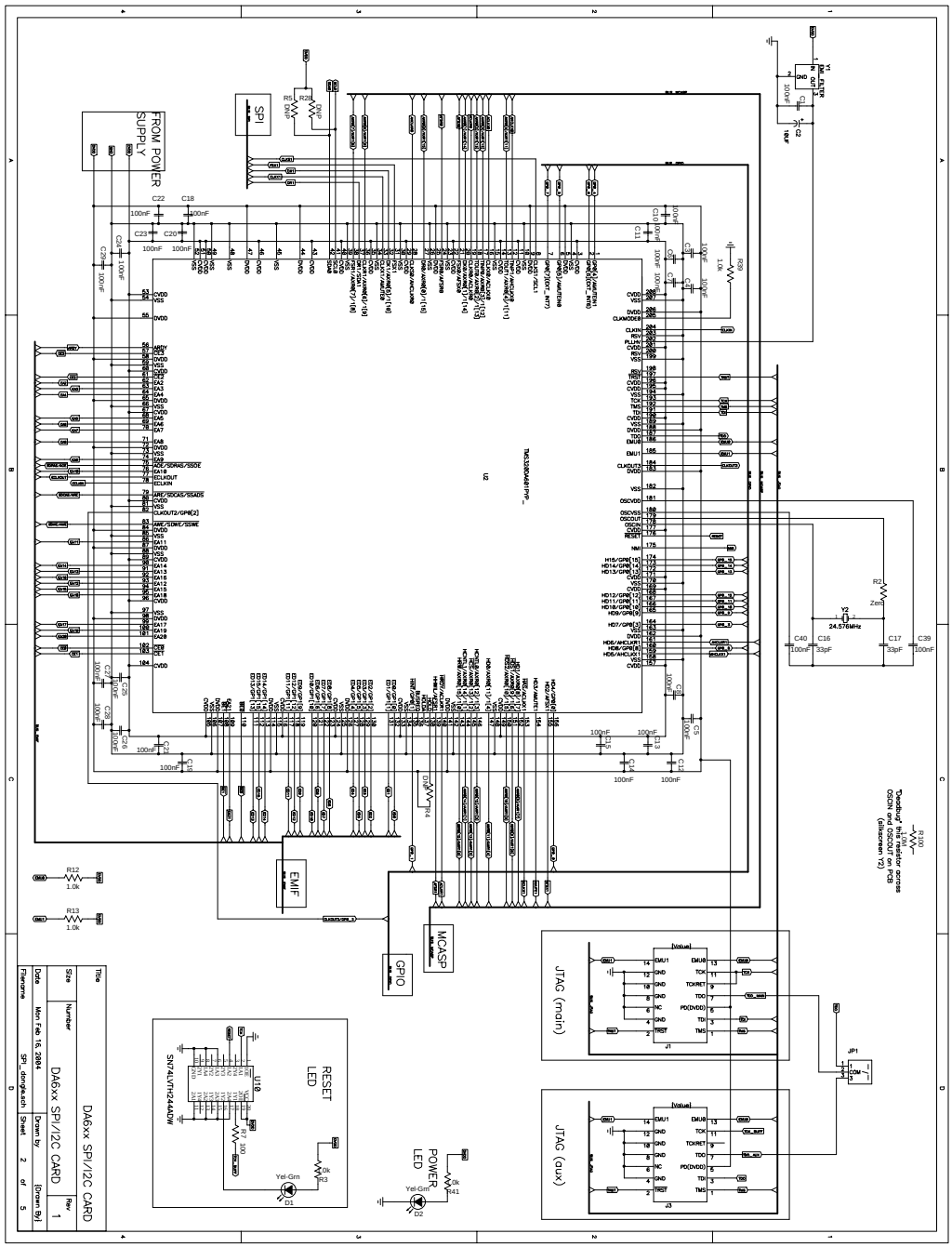


FIGURE 5-4 SPI Dongle Schematic Diagram: Memory

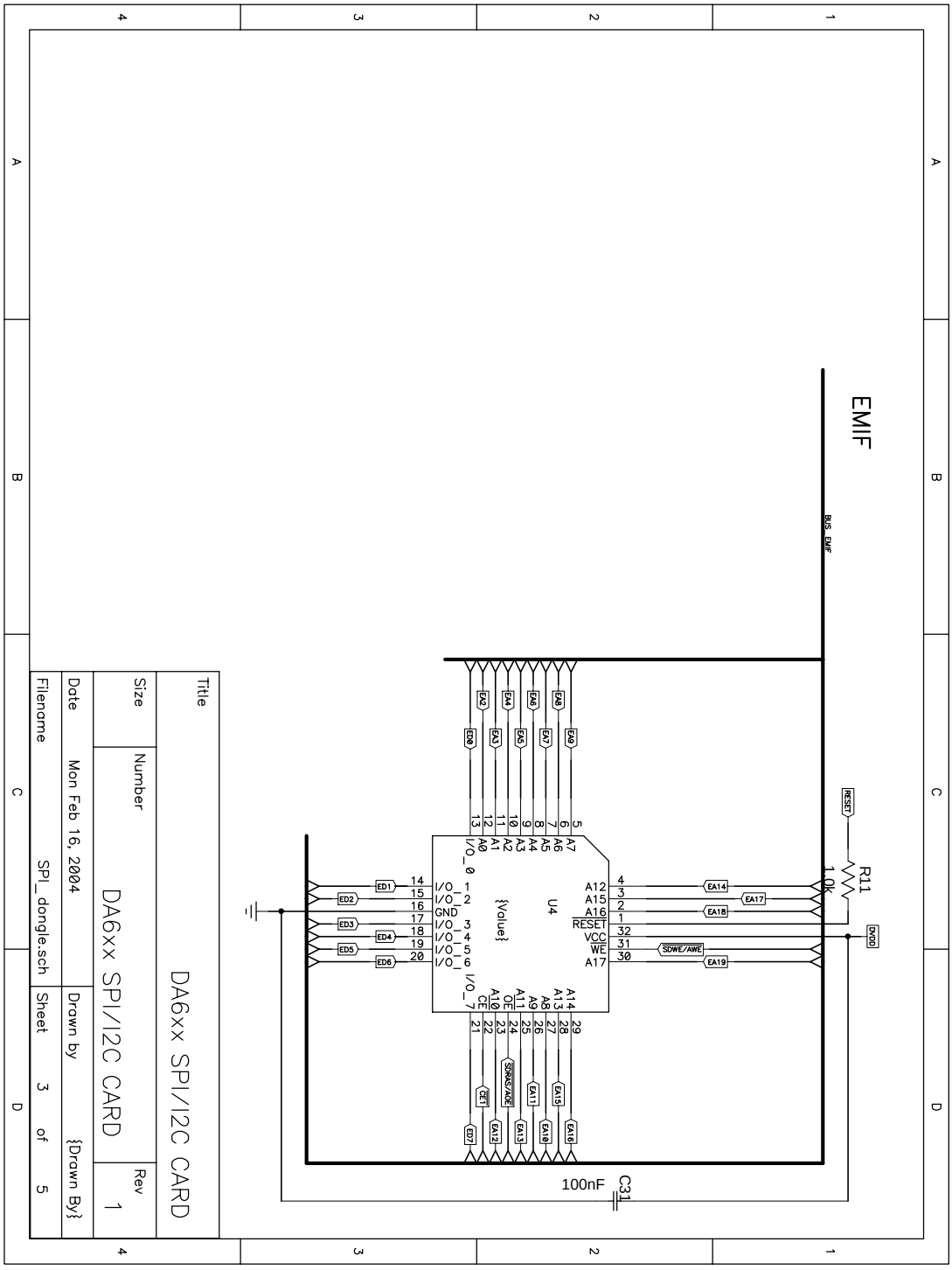


FIGURE 5-5 SPI Dongle Schematic Diagram: Power

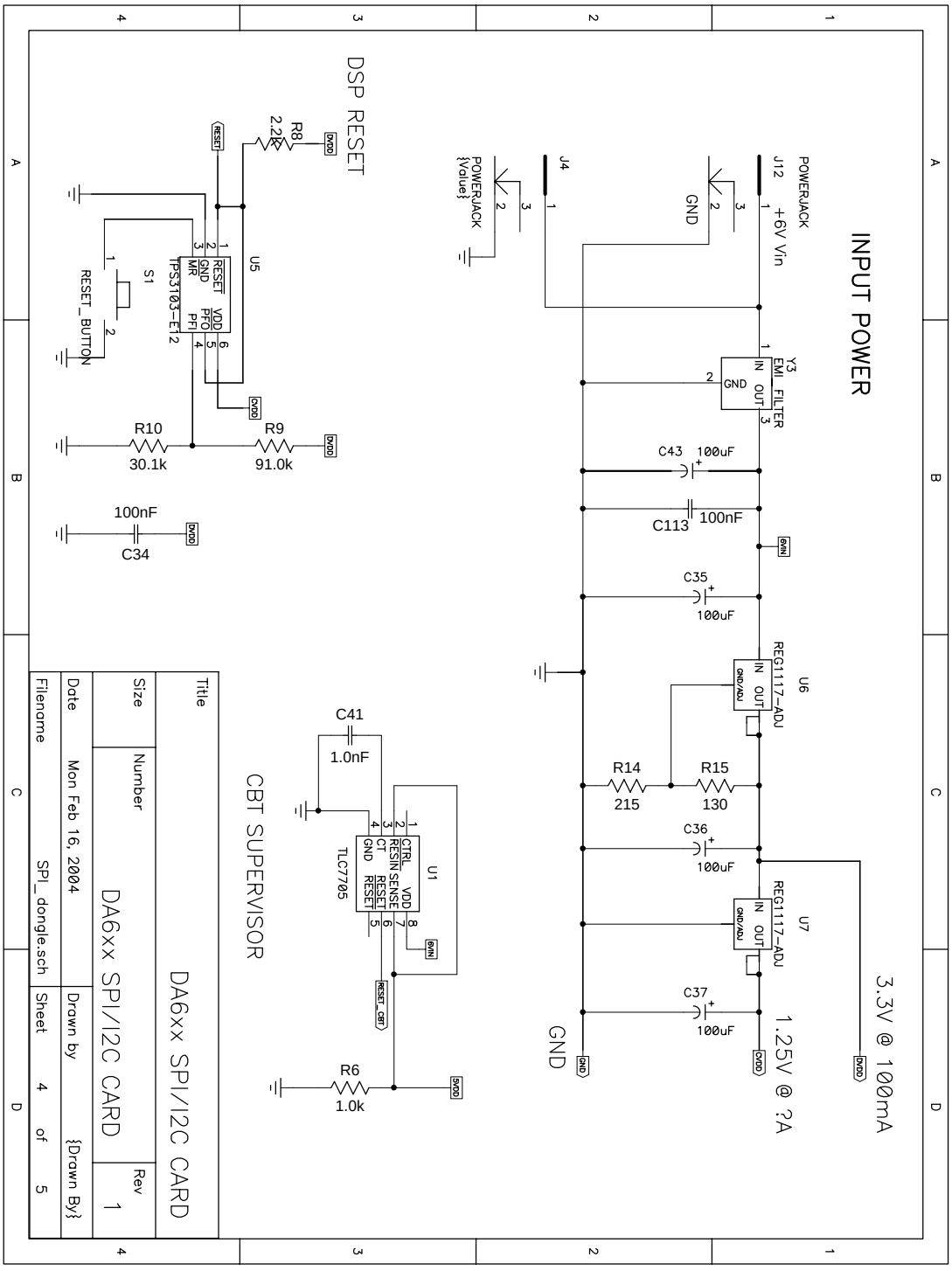
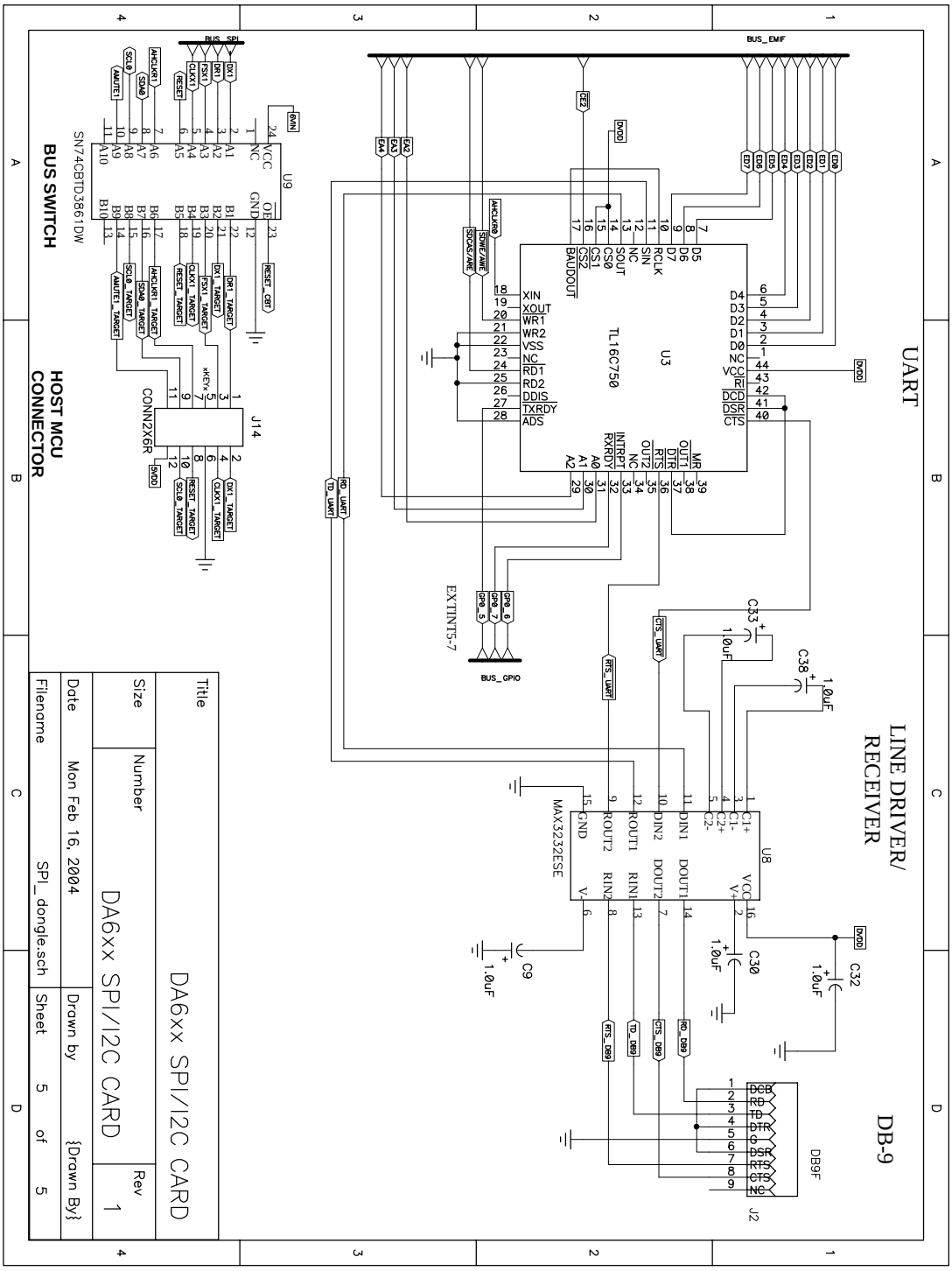


FIGURE 5-6 **SPI Dongle Schematic Diagram: Connectors**



CHAPTER 6 *Test/Example Software*

This chapter describes test and example software for the DA6XX Evaluation Module (DA6XXEVM) hardware. DA6XXEVM hardware uses the DA601/607 Processor Card (PC601/PC607), which contains the Texas Instruments DA601/607 DSP.

The Test Software is useful for "confidence testing" and for trouble-shooting problems.

The Example Software consists of source code for building and running an audio loopback program.

6.1 Installation

The test and example software are provided on the "DA6XXEVM Hardware Distribution CD" in file "`EVM_Hardware_Dist_CD_YYMMDD.zip`," where "`YYMMDD`" is the date of the release. After unzipping this file on your hard drive, the Test Software is located in directories "`\tests\memory\`" and "`\tests\loopback\`." The Example Software is located in directory "`\pc607\loopback\`."

The tests described below are contained under the "`tests`" directory in the distribution software. Refer to the Quick Start Guide near the beginning of this manual for instructions on how to load and run these tests.

6.2 Test Software

Test software includes the following function(s):

- SDRAM Test
- 8-Channel Analog Input with Analog Output
- Left/Right Front Analog Input with Digital Output
- Left/Right Surround Analog Input with Digital Output

- Center/Subwoofer Analog Input with Digital Output
- Left/Right Back Analog Input with Digital Output
- Digital Input with Analog Output

6.2.1 Memory Diagnostics Tests

6.2.1.1 SDRAM Test

The SDRAM Test is provided in file `sdram_wbyte_test.out`.

6.2.2 Audio Loop-Back Tests

6.2.2.1 Starting and Stopping the Audio Loop-Back Examples

Because of the real-time nature of the loop-back example code, and due to the way the DSP's McASP hardware is designed, it is not possible to stop the code and continue from within Code Composer Studio (CCS).

If you stop the code manually, or hit a breakpoint, you cannot just hit the **Run** button to continue. You must re-start the program using one of the following methods:

Use the CCS **Debug / Restart** menu, then hit the **Run** button (or use the **Debug / Run** menu).

Use the **File / Reload Program** menu to re-load the program, then **Run**.

More Detail:

Once the processor stops, if audio is still being input to the DSP via the McASP inputs, the McASP will detect an overrun error if the DSP CPU does not retrieve the data before the next sample comes in. If an overrun error occurs, the McASP must be re-initialized. Practically speaking, this means that the entire program must be re-started.

6.2.2.2 8-Channel Analog Input with Analog Output

8-Channel analog input is echoed to 8-Channel analog output; Left/Right Front input loops back to Left/Right Front output, Left/Right Surround input loops back to Left/Right Surround output, etc.

The 8-Channel Analog Input with Analog Output audio loop-back is provided in file `Ain4_Aout4.out`.

6.2.2.3 Left/Right Front Analog Input with Digital Output

Left/Right Front analog input is echoed to the digital output.

The Left/Right Front Analog Input with Digital Output audio loop-back is provided in file `Ain1_Dout.out`.

6.2.2.4 Left/Right Surround Analog Input with Digital Output

Left/Right Surround analog input is echoed to the digital output.

The Left/Right Surround Analog Input with Digital Output audio loop-back is provided in file `Ain2_Dout.out`.

6.2.2.5 Center/Subwoofer Analog Input with Digital Output

Center/Subwoofer analog input is echoed to the digital output.

The Center/Subwoofer Analog Input with Digital Output audio loop-back is provided in file `Ain3_Dout.out`.

6.2.2.6 Left/Right Back Analog Input with Digital Output

Left/Right Back analog input is echoed to the digital output.

The Left/Right Back Analog Input with Digital Output audio loop-back is provided in file `Ain4_Dout.out`.

6.2.2.7 Digital Input with Analog Output

Digital input is echoed to analog output.

The Digital Input with Analog Output audio loop-back is provided in file `Din_Aout4.out`.

6.2.2.8 Digital Input with Digital Output

Digital input is echoed to digital output.

The Digital Input with Digital Output audio loop-back is provided in file `Din_Dout.out`.

6.3 Example Software

6.3.1 DA6XXEVM Audio Loopback Project

A project with source code is provided for building and running a "Digital Input to Analog Output Loopback" example on the DA6XXEVM. The project, which can be used for the DA601 Processor Card (PC601) and DA607 Processor Card (PC607), is located in directory "`/pc607/loopback`."

6.3.1.1 Building and Running the DA6XXEVM Audio Loopback Project

If you haven't done so already, copy the contents of the "DA6XXEVM Hardware Distribution CD" to an arbitrary directory on your hard disk. Be sure to copy all subdirectories. *Note: The procedure below uses "`c:\temp\`" as the arbitrary directory.*

The following procedure for building and running the loopback project is described in context of the PC607 loopback project, but applies for other loopback projects that may be provided.

1. Perform Setup Hardware: DA6XXEVM on page QSG-2 of the Quick Start Guide.
2. In a separate DOS window, type "`subst T: c:\temp`"
   Note: If `T:` drive is already substituted, type "`subst T: /D`" to un-substitute it first.
3. With the DA6XXEVM powered-on, start Code Composer Studio (CCS).
4. In the CCS pull-down menu, select **Debug / Reset CPU**.
5. Select **Project / Open** then navigate to select "`T:\pc607\loopback.pjt`."
6. In the CCS pull-down menu, select Build Configuration:[13] **Debug**.
7. Select **Project / Build**. The resulting "`loopback_da607.out`" file will appear in directory: `T:\pc607\loopback\Debug`.
8. In the CCS pull-down menu, select **Debug / Reset CPU**.[14]
9. Select **File / Load GEL** then navigate to select:
   "`T:\gel\board_evmda607_d607.gel`."
10. Select **File / Load Program** then navigate to select:
    "`T:\pc607\loopback\Debug\loopback_da607.out`."
11. Select **Debug / Run**.
    *Note: The above can also be performed pressing the **F5** key, or click the **Run** icon on the left side of the window.*
12. When finished, select **Debug / Halt** to stop the program.
    *Note: The above can also be performed pressing **Shift-F5**, or click the **Halt** icon.*
13. To run the same program again, repeat steps 8 and 9, then select **File / Reload Program**, then continue with step 11.

13. Note that one may also select Build Configuration **Release**. The Debug configuration is intended for debugging purposes; the Release configuration will have better performance, and should be used in the case of performance testing.

14. It is critical to always select **Debug / Reset CPU** then load/reload the GEL file before running a program.

CHAPTER 7 *Performance Audio Software and Firmware*

This chapter describes Performance Audio (PA) software and firmware releases provided on the "DA6XXEVM Hardware Distribution CD."

7.1 Installation

If you haven't done so yet, install the contents of the "DA6XXEVM Hardware Distribution CD" to a directory on your hard disk as described in Section 6.1, "Installation," on page 59. The releases described below are contained therein.

7.2 Performance Audio PA17 Release

PA17 software and firmware are provided for use with DA6XXEVM hardware. PA17 is an instance of the Open Audio System, which includes the Performance Audio Framework (PA/F). For more information, refer to the *"PA3C User's Guide"* provided on the "DA6XXEVM Hardware Distribution CD" at "`/doc/pa3c-ug.pdf`." *(PA17 User's Guide is not yet available.)*

After installing the contents of the CD, and performing Installation of Code Composer Studio 2.2 for use with the XDS510PP Plus or other JTAG emulator (see Appendix B, "Code Composer Studio for DA6XXEVM," on page 69), follow the procedures in the Quick Start Guide to setup your hardware. The Quick Start Guide's Building and Running Performance Audio System Code on page QSG-7 describes how to build and run PA17 software/firmware (`pa17i_YYMMDD.zip`). For more information, see the *"PA3C User's Guide."*

7.2.1 Firmware Download To Flash Memory

See "Firmware Download" on page 83 of *Appendix B (Code Composer Studio for DA6XXEVM)*

7.3 Performance Audio PA17 Software Development Kit Release

PA17 Software Development Kit (SDK) software and firmware are provided for use with DA6XXEVM hardware. PA17 SDK is an instance of the Open Audio System, which includes the Performance Audio Framework (PA/F). For more information on the SDK, refer to the *"PA3C SDK User's Guide"* provided on the "DA6XXEVM Hardware Distribution CD" at "`/doc/pa3csdk-ug.pdf`."[15]

After installing the contents of the CD, and performing Installation of Code Composer Studio 2.2 for use with the XDS510PP Plus or other JTAG emulator (see Appendix B, "Code Composer Studio for DA6XXEVM," on page 69), follow the procedures in the Quick Start Guide to setup your hardware. The Quick Start Guide's Building and Running Performance Audio System Code on page QSG-7 describes how to build and run PA17 SDK software/firmware (`pa17isdk_YYMMDD.zip/pa-pa17isdk_YYMMDD.zip`). For more information, see the *"PA3C SDK User's Guide."*

7.3.1 Firmware Download To Flash Memory

See "Firmware Download" on page 83 of *Appendix B (Code Composer Studio for DA6XXEVM)*

15. PA17/SDK User Guides are not yet available.

APPENDIX A *Data Sheets and References*

This appendix gives a list of web sites where the data sheets and related references for DA6XXEVM can be found. Data sheets for Texas Instruments devices are also provided in the `/doc/DataSheets` directory on the "DA6XXEVM Hardware Distribution CD."

Note: Documents available via manufacturers web sites may contain updated versions relative to the documents provided on the "DA6XXEVM Hardware Distribution CD."

A.1 DA6XXEVM Technical Documentation

A.1.1 DA6XXEVM Data Sheet

http://www.mds.com/Products/product.asp?prod=DA%2D60x%2DKIT

A.1.2 DA6XX EVM Audio Card (AC6XX)

Reference: "AC607 Schematic Diagrams" on page 11.

A.1.2.1 TPS79601/633 Data Sheet

http://focus.ti.com/lit/ds/symlink/tps79601.pdf

A.1.2.2 PCM1802 Data Sheet

http://focus.ti.com/lit/ds/symlink/pcm1802.pdf

A.1.2.3 DIR1703 Data Sheet

http://focus.ti.com/lit/ds/symlink/dir1703.pdf

A.1.2.4 DSD1608 Data Sheet

http://focus.ti.com/lit/ds/symlink/dsd1608.pdf

A.1.2.5 OPA4344 Data Sheet

http://focus.ti.com/lit/ds/symlink/opa4344.pdf

A.1.2.6 ULN2003 Data Sheet

http://focus.ti.com/lit/ds/symlink/uln2003ai.pdf

A.1.3 DA60X Processor Card (PC601, PC607)

References:

- "PC607 Schematic Diagrams" on page 42.
- "PC601 Schematic Diagrams" on page 30.

A.1.3.1 REG1117A Data Sheet

http://focus.ti.com/lit/ds/symlink/reg1117a.pdf

A.1.3.2 HCU04 Data Sheet

http://focus.ti.com/lit/ds/symlink/cd74hcu04.pdf

A.1.3.3 TMS320DA601 Data Sheet

This data sheet is provided on the "DA6XXEVM Hardware Distribution CD" at:
`/doc/DataSheets/sprs002g.pdf`

A.1.3.4 TMS320DA607 Data Sheet

This data sheet is provided on the "DA6XXEVM Hardware Distribution CD" at:
`/doc/DataSheets/sprs212b.pdf`

A.1.3.5 AM29LV320 Data Sheet

Data sheet is available from: http://www.amd.com

A.1.3.6 AT29LV020 Data Sheet

Data sheet is available from: http://www.atmel.com

A.1.3.7 K4S641632H Data Sheet

Data sheet is available from: http://www.usa.samsungsemi.com

A.1.3.8 AT25HP512 Data Sheet

Data sheet is available from: http://www.atmel.com

A.1.4 DA6XX SPI/I²C Card (SPI Dongle)

Reference: "SPI Dongle Schematic Diagrams" on page 53.

A.1.4.1 REG1117A Data Sheet

http://focus.ti.com/lit/ds/symlink/reg1117a.pdf

A.1.4.2 TPS3103E12 Data Sheet

http://focus.ti.com/lit/ds/symlink/tps3103e12.pdf

A.1.4.3 MAX3232ESE Data Sheet

Data sheet is available from: http://www.maxim-ic.com

A.1.4.4 TMS320DA601 Data Sheet

This data sheet is provided on the "DA6XXEVM Hardware Distribution CD" at:
`/doc/DataSheets/sprs002g.pdf`

A.1.4.5 SN74LVTH244A Data Sheet

http://focus.ti.com/lit/ds/symlink/sn74lvth244a.pdf

A.1.4.6 TL16C750 Data Sheet

http://focus.ti.com/lit/ds/symlink/tl16c750.pdf

A.1.4.7 TLC7705 Data Sheet

http://focus.ti.com/lit/ds/symlink/tlc7705.pdf

A.1.4.8 SN74CBTD3861 Data Sheet

http://focus.ti.com/lit/ds/symlink/sn74cbtd3861.pdf

A.2 Development Technical Documentation

A.2.1 TMS320C67x Fast RTS Library (SPRC060)

Texas Instruments TMS320C67x Fast RTS Library is available as part of the standard version of CCS 2.2. It is also available at:

http://www-s.ti.com/sc/psheets/sprc060/sprc060.zip

A.2.2 TMS320C67x Fast RTS Library User's Guide (SPRU100)

Texas Instruments TMS320C67x Fast RTS Library User's Guide is available at:

http://focus.ti.com/lit/ug/spru100a/spru100a.pdf

A.2.3 Spectrum Digital XDS510PP Plus Emulator

http://www.spectrumdigital.com/cgi/catalog.cgi?show_product=701014

APPENDIX B *Code Composer Studio for DA6XXEVM*

Requirements and installation of the following items for DA6XXEVM is documented in this appendix:

- Texas Instruments C6000 PC Code Composer Studio IDE (DSP4387U) Version 2.2 and supplementary materials.
- Texas Instruments FlashBurn Flash Memory Programmer Utility, version 2.71 or later.
- Spectrum Digital XDS510PP Plus Emulator or equivalent JTAG emulator.
- The Hex Conversion Utility (`coff2hex`) contained in the Performance Audio PA17 Software Development Kit Release.

B.1 Requirements

Texas Instruments Code Composer Studio (CCS) Version 2.2 is used with DA6XXEVM. Use of other versions of CCS is not supported.

Texas Instruments Fast RTS Library is used with Performance Audio Software and Firmware. The current version of the Fast RTS Library is supported.

Texas Instruments FlashBurn Flash Memory Programmer Utility (FlashBurn) is used with Performance Audio firmware. FlashBurn, version 2.71 or later, with FlashBurn Target Component (FBTC) Communications Protocol Version 3, is recommended.

The Hex Conversion Utility (`coff2hex`) is provided in and used with the Performance Audio PA17 Software Development Kit Release. This utility is required in order to convert developed firmware to a format suitable for FlashBurn.

A JTAG emulator is used with DA6XXEVM. Use of the Spectrum Digital XDS510PP Plus JTAG emulator is documented here. Other, equivalent emulators may be substituted for this particular unit. Both parallel port and ISA-based JTAG emulators may be utilized. Documentation for other such emulators is not, however, specifically provided herein.

B.2 Installation

This section provides complete information on how to install the tools required to download DA6XXEVM software.

B.2.1 Code Composer Studio 2.2

Texas Instruments Code Composer Studio (CCS) 2.2 must be installed on your system to download DA6XXEVM software:

- Texas Instruments C6000 PC Code Composer Studio IDE (DSP4387U) Version 2.2 can be obtained via http://www.ti.com.

Install CCS according to the directions provided in the distribution.[16] Installation in the default location C:\ti is recommended.

Code Composer Studio (CCS) version 2.20.14 or later must be installed in order to properly build the projects. Check the version of CCS by selecting **Help / About** in the CCS pull-down menu.

If command-line tools are to be used from a shell, run `c:\ti\DosRun.bat` to add CCS tools to your path:

```
C:\> C:\ti\DosRun.bat
```

Note: *Install CCS in the directory C:\ti.*

Although installation in this directory is not required, it is highly recommended.

B.2.2 TMS320C67x Fast RTS Library

Performance Audio PA17 Release and Performance Audio PA17 Software Development Kit Release use the TMS320C67x Fast RTS Library which accompanies CCS:

- Texas Instruments TMS320C67x Fast RTS Library (`SPRC060`) is available as part of the standard version of CCS 2.2. It is also available directly from Texas Instruments; see "TMS320C67x Fast RTS Library User's Guide (SPRU100)" on page 68.
- Texas Instruments TMS320C67x Fast RTS Library User's Guide (`SPRU100`) is available directly from Texas Instruments; see "TMS320C67x Fast RTS Library User's Guide (SPRU100)" on page 68.
- Additional information, as well as the two links above, is available directly at: http://focus.ti.com/docs/tool/toolfolder.jhtml;?PartNumber=SPRC060.

16. When installing CCS under Windows XP, before you run setup, click on properties, select the compatibility tab (only shows up on XP) and select compatible for Windows NT4.

The *TMS320C67x Fast RTS Library User's Guide* describes how to use the TMS320C67x Fast RTS Library. Its use is automatic with Performance Audio PA17 Release and Performance Audio PA17 Software Development Kit Release, and no action other than installation is required.

B.2.3 FlashBurn Flash Memory Programmer Utility

Texas Instruments FlashBurn Flash Memory Programmer Utility (FlashBurn) must be installed on your system to program Performance Audio and custom firmware into the flash device:

- Texas Instruments FlashBurn Flash Memory Programmer Utility can be obtained via http://www.ti.com. **Update Advisor** is a live update feature in the Code Composer Studio version described above that will take you to a web page to download FlashBurn.

Install FlashBurn according to the directions provided in the distribution. The installer should place the FlashBurn components into the same top-level directory as Code Composer Studio (usually `C:\ti`), under `C:\ti\bin\utilities\flashburn`.

FlashBurn is required to be run on a command line from a shell for programming Performance Audio or custom firmware into the flash device. The associated tools require that FlashBurn be included in your path. The following command will set your path to include FlashBurn:

```
C:\> SET PATH=%PATH%;c:\ti\bin\utilities\flashburn;
```

Note: *Install FlashBurn in the directory C:\ti\bin\utilities\flashburn.*

Although installation in this directory is not required, it is highly recommended. If FlashBurn is not installed in this directory, tools which refer to files resident in `C:\ti\bin\utilities\flashburn` *will have to be modified to refer to the files in the directory in which FlashBurn has been alternatively installed.*

B.2.4 XDS510PP Plus

A JTAG emulator must be installed on your system to download software and firmware to the DA6XXEVM:

- For example, see “Spectrum Digital XDS510PP Plus Emulator” on page 68.
- Other, equivalent emulators may be substituted for this particular unit.[17]

Install hardware and software for support of XDS510PP Plus from Spectrum Digital, Texas Instruments, and Momentum Data Systems according to the instructions below.

17. Documentation of installation instructions for such emulators is beyond the scope of this document.

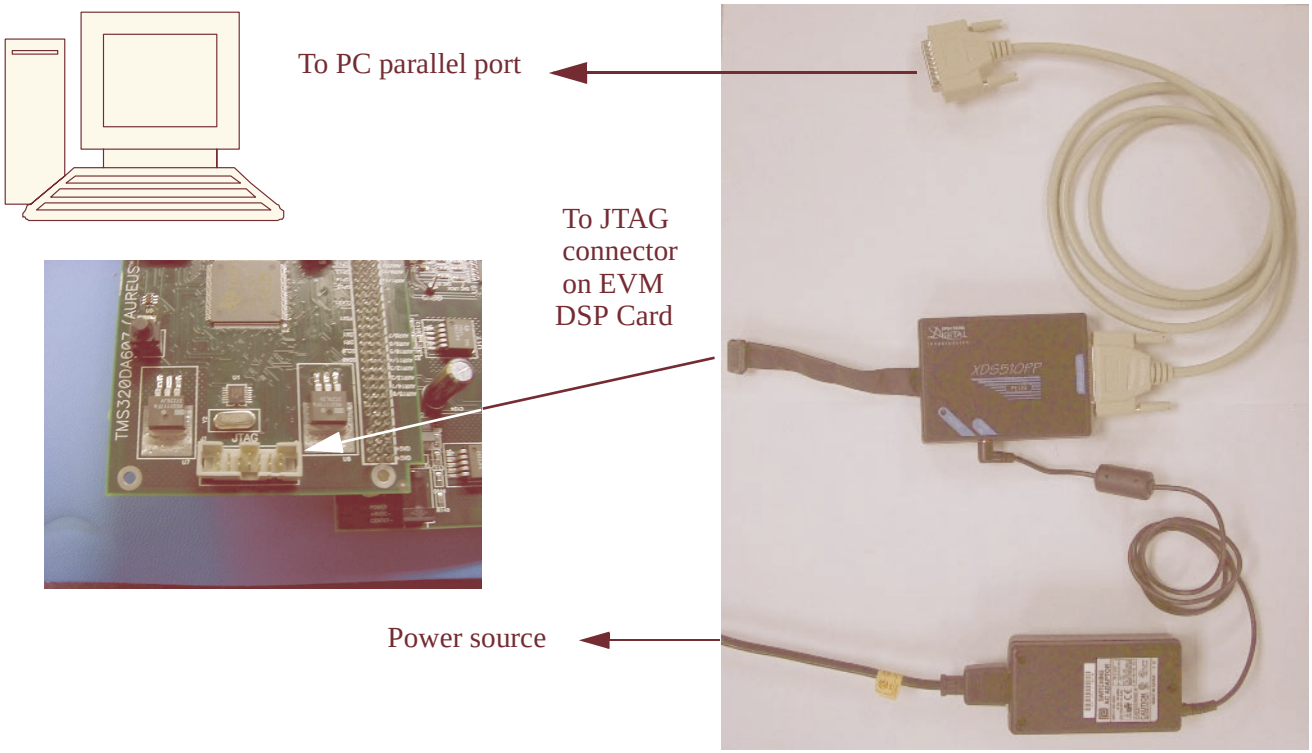
B.2.4.1 XDS510PP Plus Hardware

The XDS510PP Plus JTAG emulator, along with the required cables and connections, is shown in Figure B-1 (XDS510PP Plus JTAG Emulator with Cables).

Connect XDS510PP Plus to PC and DA6XXEVM as described below:

- Using the parallel cable, connect the emulator to the parallel port of the PC.
- Connect the ribbon cable on the emulator to the JTAG connector on the DA607 Processor Card (PC607).
- Connect the power supply to the emulator.

FIGURE B-1 XDS510PP Plus JTAG Emulator with Cables



B.2.4.2 XDS510PP Plus Spectrum Digital Software

Install software for support of XDS510PP Plus from Spectrum Digital as follows:

A. Download software from the Spectrum Digital FTP site:[18]
http://www.spectrumdigital.com/drivers/docstore/CodeComposerDrivers/C6000/Emulator/CodeComposerStudio-2.x/Release-2.20/

18. This is the latest software at the time of this writing. Be sure to check the site for any newer versions. One way is to use the highest-numbered directory you find under this directory: http://www.spectrumdigital.com/drivers/docstore/CodeComposerDrivers/C6000/Emulator/CodeComposerStudio-2.x/

The file downloaded should be unzipped into a temporary directory to extract its contents.

If requested, a UserID and Password can be obtained from: http://www.spectrumdigital.com/cgi/sdlogin.cgi.

B. Run the downloaded executable. This is a Windows installation program that will install the Spectrum Digital Drivers. Install the drivers into the default directory suggested by the install program `C:\ti`.

C. The `sdgo6x.dvr` driver is used by Code Composer Studio to communicate through the Spectrum Digital XDS510PP JTAG with the processor. Check that the correct version of the driver was installed. Right click on `C:\ti\drivers\sdgo6x.dvr`, select **Properties**, and then the **Version** tab. The **File version** should be 5.22.0.0.

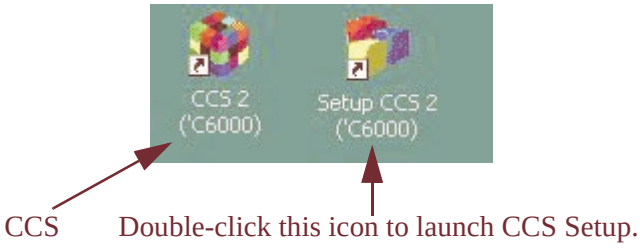
D. Take all the defaults, and finish installing the driver that is appropriate for your system: Windows 98, Windows NT, etc.

B.2.4.3 XDS510PP Plus Texas Instruments and Momentum Data Systems Software

Install software for support of XDS510PP Plus from Texas Instruments and Momentum Data Systems as follows:

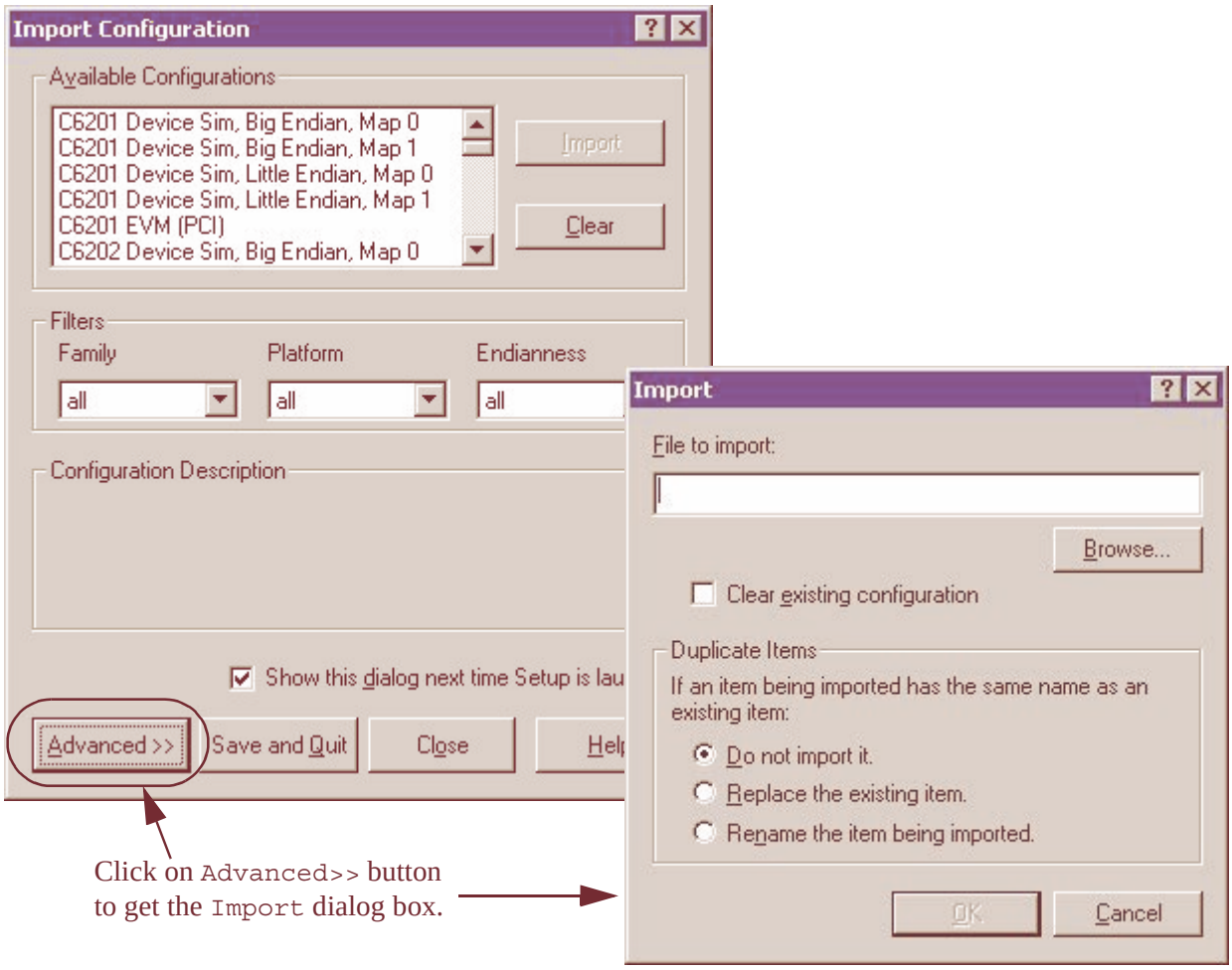
1. Run the CCS Setup Program. There should be an entry in your Windows **Start / Programs / Texas Instruments / Code Composer Studio 2 ('C6000) / Setup Code Composer Studio** menu for this.
   As shown in Figure B-2, there also may be an icon on your desktop that can launch the CCS Setup Program directly (probably named "Setup CCS 2 ('C6000)").

FIGURE B-2 CCS Desktop Icons



2. After you start the program, you will be in the **Import Configuration** dialog box shown in Figure B-3. Use the **Advanced >>** button. This opens the `Import dialog` box.

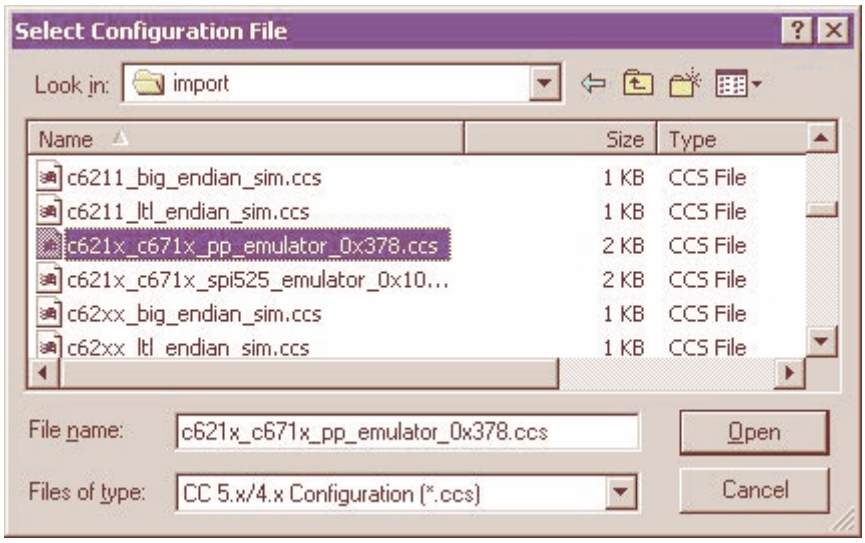
FIGURE B-3 **Import Configuration Dialog**



3. Browse to **c:\ti\drivers\import** and select the file c621x_c671x_pp_emulator_0x378.ccs (if you are using the LPT1 parallel port).[19] Either double-click the file, or select it and hit the **Open** button**.** This is shown in Figure B-4.

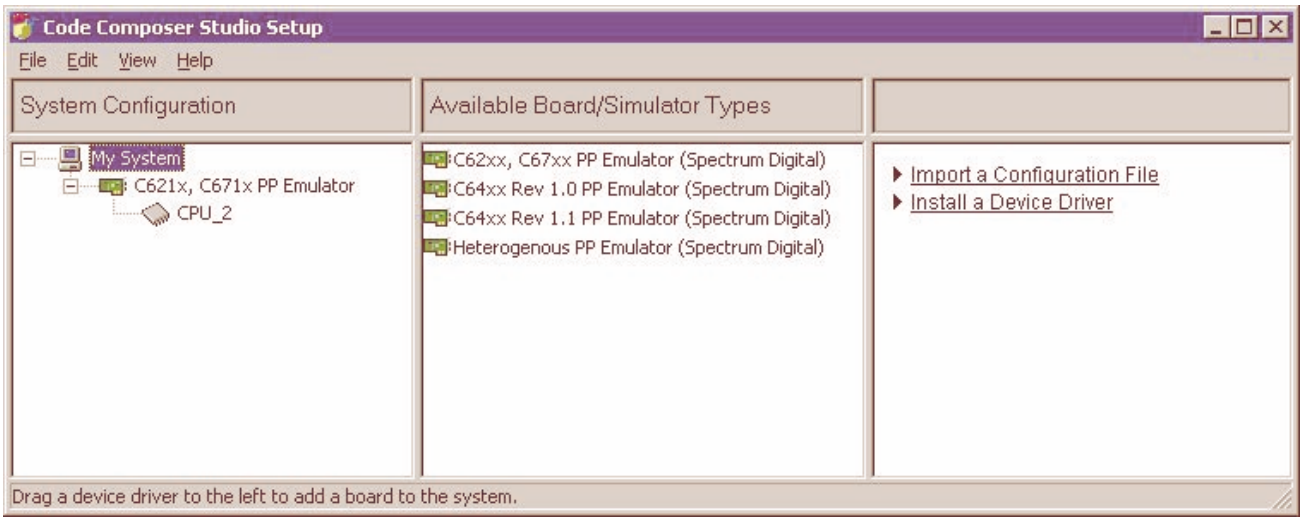
19. Select the file that is appropriate for your system. "_0x378" selects the driver that uses the I/O port that is located at hex address 378, which is usually the address that corresponds to LPT1. If you need to use a different port, consult the documentation for Code Composer Studio, for the Spectrum Digital XDS, and for Windows.

FIGURE B-4 **Select the File c621x_c671x_pp_emulator_0x378.ccs**



4. Select **OK** to get out of the Import dialog, then select **Close** to get out of the **Import Configuration** box.
   This should make a “C621x, C671x PP Emulator” entry show up in the “System Configuration” pane of the Code Composer Studio Setup window as shown in Figure B-5.

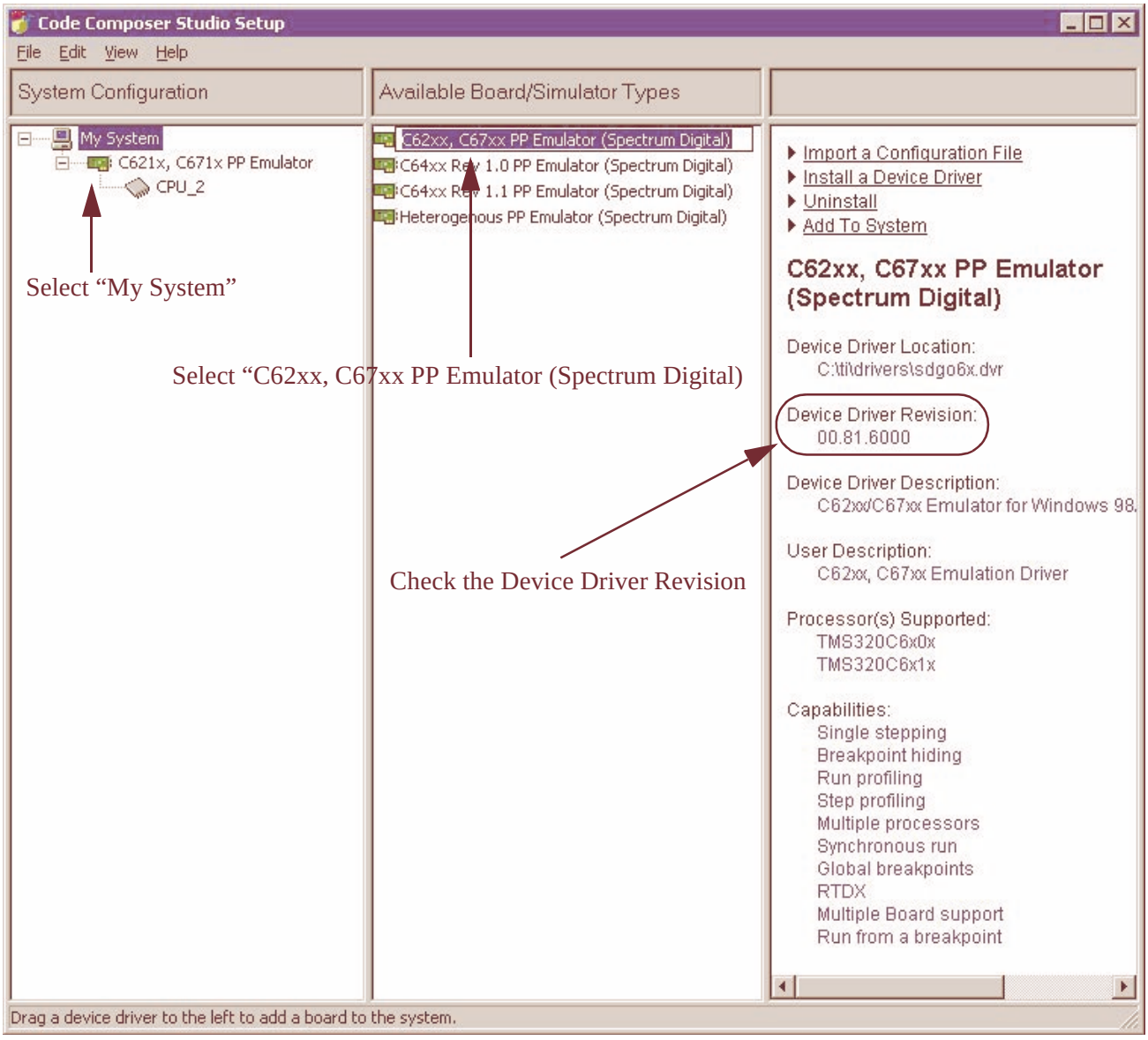
FIGURE B-5 **System Configuration Pane**



5. As shown in Figure B-6, left-click once on “My System” in the “System Configuration” pane to select it, and then on “C62xx, C67xx PP Emulator (Spectrum Digital)” in the center pane. As of this writing, the Device Driver Revision number in the right-

most pane of CCS Setup should read 00.81.6000. If it shows an earlier version, be sure to follow the instructions in the previous subsection **XDS510PP Plus Spectrum Digital Software**.

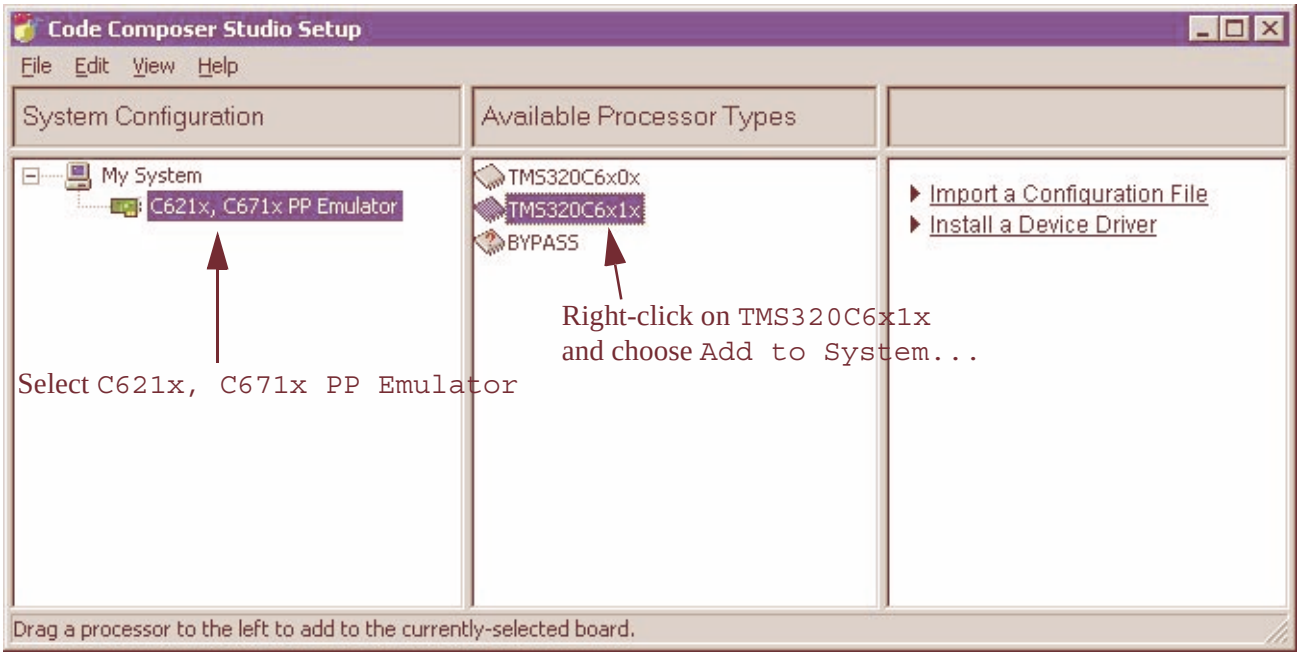
FIGURE B-6 Device Driver Revision



6. In the "System Configuration" pane, if there are any other entries under the "My System" tree besides the "C621x, C671x PP Emulator" one, right-click it and choose **remove**. Answer "Yes" to remove the board. (You only want to have one "board" entry in the System Configuration pane, for now.)

7. Now, right-click the “CPU_2” entry underneath the “C621x, C671x PP Emulator” entry in the “My System” tree. Choose **remove**. Answer “Yes” to remove the processor.
8. Next, single-click the “C621x, C671x PP Emulator” in the “System Configuration” pane to select it. As seen in Figure B-7, you should see some entries appear in the “Available Processor Types” pane (the middle one).

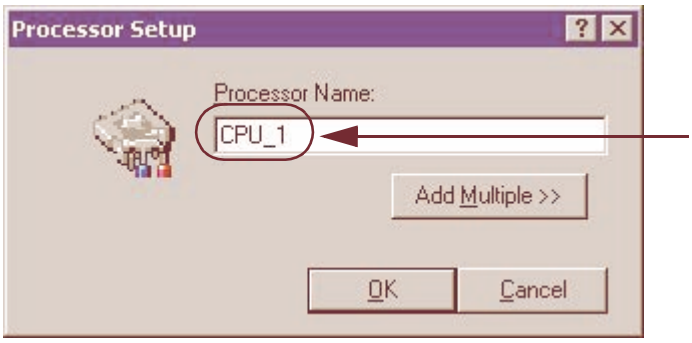
FIGURE B-7 System Configuration Pane and Available Processor Types Pane



9. In the “Available Processor Types” pane, right-click the “TMS320C6x1x” entry, and choose **Add to System...**.
10. In the “Processor Setup” dialog box, change the name “CPU” to “CPU_1”. This is shown in Figure B-8. Hit OK.

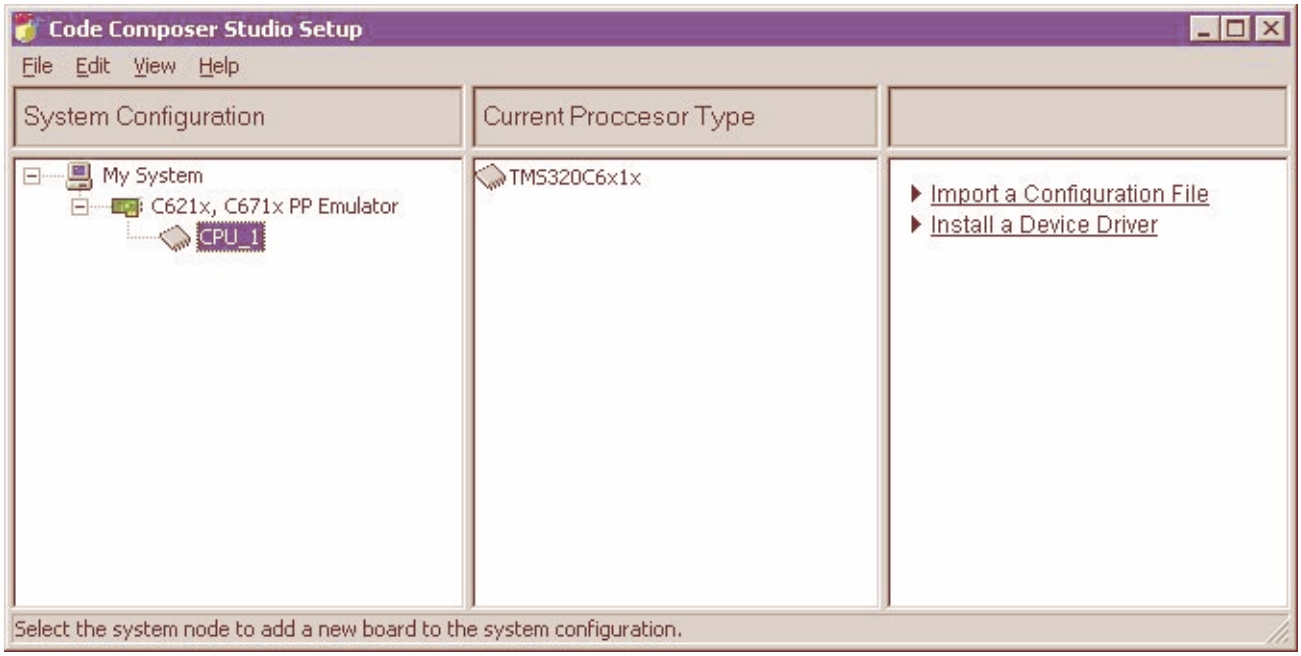
Note: *For Setup Hardware: SPI/I²C Card: Parallel Debug Mode, it is necessary to add a second CPU. This can be performed by repeating steps [9] and [10] to add “CPU_2”.*

FIGURE B-8 **Processor Setup Dialog**



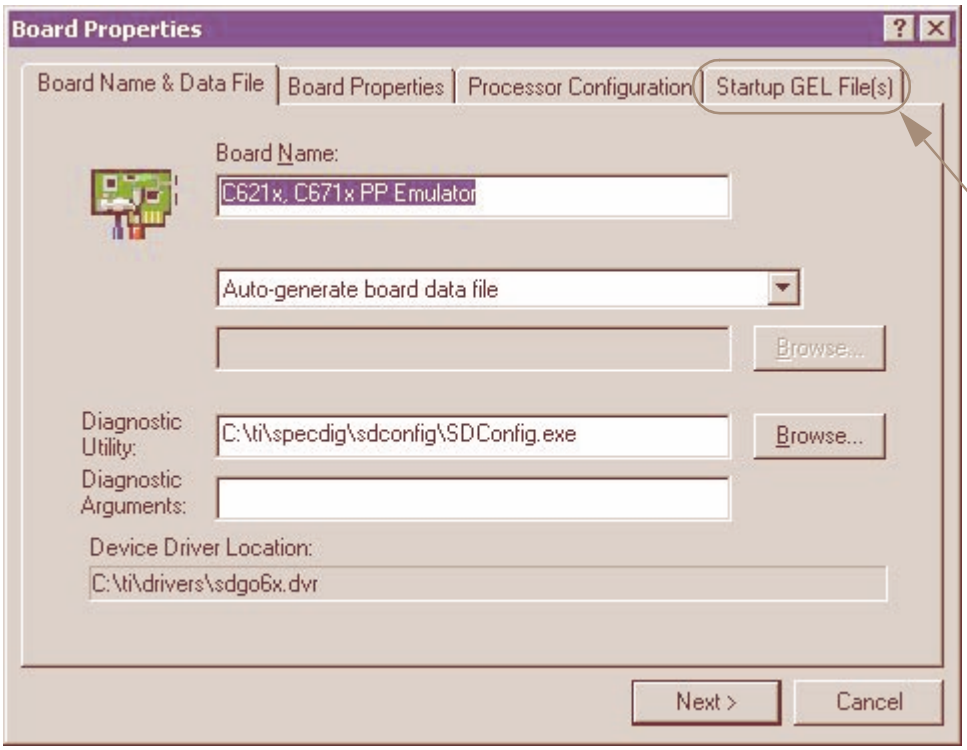
When you are done, the CCS Setup window should look similar to Figure B-9.

FIGURE B-9 **System Configuration Pane After Processor Setup**



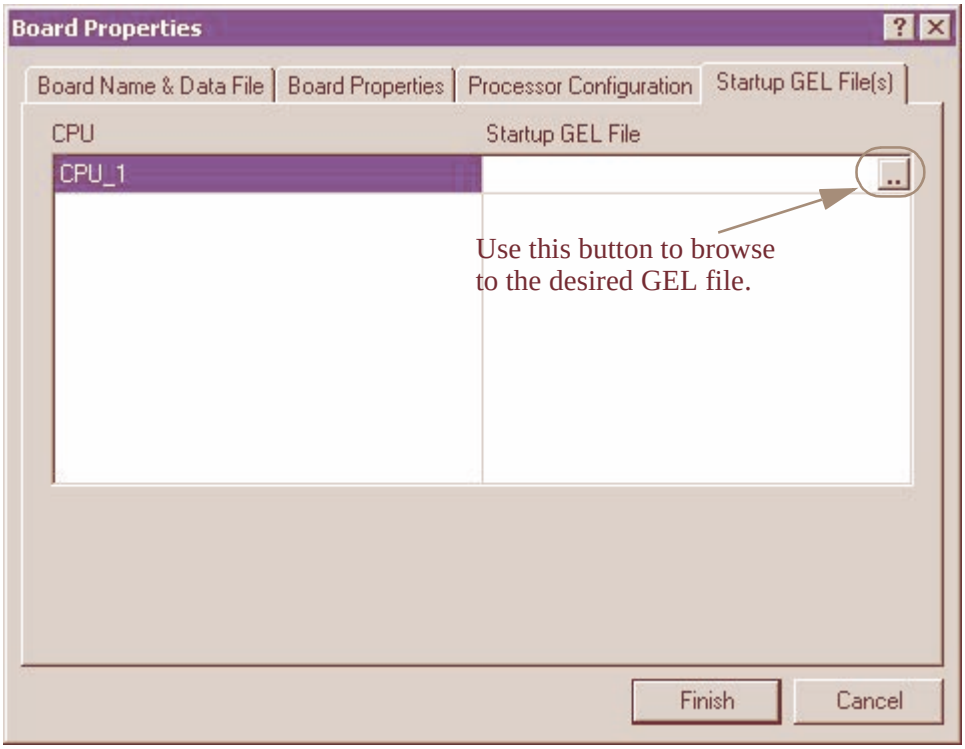
11. Now, back in the “System Configuration” pane, right-click the “C621x, C671x PP Emulator” entry and choose **Properties...** This opens the “Board Properties” dialog as shown in Figure B-10.

FIGURE B-10 Board Properties Dialog



12. In the “Board Properties” dialog, select the “Startup GEL File(s)” tab. The contents of this tab are shown in Figure B-11.

FIGURE B-11 **Startup GEL File(s) Tab**



13. Use the "**..**" button (all the way to the right of the "CPU_1" line) to browse to one of these files:

    ```
    T:\pa\gel\board_evmda601_d601.gel # PC601
    T:\pa\gel\board_EVMDA607_D607.gel # PC607
    ```

    The directory `P:` here is that defined during the installation of DA6XXEVM software. Double-click the file, or select it and hit **Open** to choose it.
14. Click **finish** in the "Board Properties" dialog.
15. Use the **File / Save** menu to save the setup.
16. Exit the Code Composer Studio Setup application. Answer "No" when asked whether or not to Start Code Composer Studio on exit.

B.3 Use

This section provides basic information on how to use the installed tools to download PA17 firmware.

B.3.1 Field Upgrade

The hardware distributed with the PA17/DA6XXEVM Kit includes a pre-programmed, flash memory device containing PA17 firmware. This firmware provides the powerup features exhibited by the DA6XXEVM hardware. In case the contents of this memory device become corrupted, it can be reprogrammed using the techniques described below.

The software and firmware portion of the PA17/DA6XXEVM Kit may be redistributed as an update to that originally provided via a PA17 Deliverable. In this case, the contents of the memory device can be reprogrammed with the new firmware, thus providing a field upgrade of the DA6XXEVM hardware to the new version of the PA17 firmware.

In addition, a PA17 Deliverable may be applied to the DA6XXEVM hardware obtained not as part of a PA17/DA6XXEVM Kit, but as part of some other product. In this case, the contents of the memory device can be reprogrammed with the PA17 firmware, thus providing a field upgrade that changes use of the DA6XXEVM hardware from its previous to PA17.

The sections below describe how to download firmware to the DA6XXEVM to provide field upgrade for PA17.

B.3.1.1 Firmware Programming

This section provides an overview of the firmware structure with respect to programming of the flash device.

The firmware to be programmed as the contents of the flash memory has three components:

1. Boot Loader
2. Program
3. Version ID

The Program section is equivalent to the PA17 software. The Boot Loader and Version ID sections provided are included automatically when programming the flash device.

During power-up operation, the Boot Loader section provides the functions of copying the Program section from flash memory to RAM and starting execution of it from RAM. The Version ID section contains PA17 version information.

As stated above, the Boot Loader and Version ID sections are automatically included during flash memory programming by the programming tool provided. As these sections do not change, further description of the Boot Loader and Version ID is not required for description of programming the flash device.

Programming of the flash device is performed using CCS and FlashBurn. Required GEL, FlashBurn configuration, FlashBurn Target Component (FBTC) and hex files are provided. The hex files are the sections described above. As FlashBurn is executed from a shell command line for programming of PA17 firmware, a DOS batch file is provided which runs the entire programming session.

Flash programming is 100% verified using checksum processing.

B.3.1.2 Firmware Upload

Depending upon the user application, it may be important to upload the contents of the flash memory to be saved in a file on the PC. This is an important debugging tool:

- The original flash contents can be reprogrammed exactly in that unit despite intervening reprogramming.
- The flash contents can be compared with those of another Mozart II unit to determine if hardware or firmware differences cause operation of two units to differ.
- The flash contents can be compared with those downloaded to reverify proper programming.

This section describes how to upload firmware from the flash memory.

1. Using Code Composer Studio, navigate through the pull-down menu to select **File/Data/Save**. This will open a Store Data dialog box.
2. In the **Store Data** dialog box, specify a data file name. This dialog offers the following options:

   Save in: Lists the available folders and files. To see how the current folder fits in the hierarchy on your computer, click the down arrow. To see what's inside a folder, click it.

   The box below this field shows the folders and files in the selected location. Double-click a folder to open it. When a file is selected, the file name is displayed in the **File** name field. Double-click a file to save it.

   File name: Provides a space for you to enter the name of the file. You can use "`*`" as a wildcard. For example, you can type "`*.*`" to see a list of all files. You can also type the full path of a file. For example, you can type:
   `c:\ti\myprojects\sample\myfile.dat`.

   Save as type: Specifies the type of file you are saving. The list includes all the available file types that Code Composer Studio can recognize. By default, Hex (`*.dat`) is selected.
3. Click **Save**.
4. In the **Storing Memory into File** dialog box, specify the starting **Address** and the **Length** of the data you want to store.
5. Click **OK**.

B.3.1.3 Firmware Development

Once custom firmware has been developed, programming it into the flash device can be performed by converting the associated "`.out`" file to hex format, then substituting the custom firmware hex file for the hex firmware file provided with the PA17 SDK Deliverable and performing the Firmware Download procedure.

The Hex Conversion Utility (`coff2hex`), provided with the PA17 SDK Deliverable creates the custom firmware hex file. The procedure for converting a custom firmware "`.out`" file to hex format is provided below. The procedure for subsequent download of the resulting custom firmware hex file to flash memory is described in "Firmware Download" on page 83.

This procedure assumes the custom file has the same name as the firmware file provided with the PA17 SDK Deliverable (`pa17isdk.out`), as it suits the CCS project file provided with the PA17 SDK Deliverable. If desired, there is the option to rename the file within the project file and accordingly within this procedure. If your custom firmware file is named `pa17isdk.out`, you may want to save the PA17 SDK Deliverable version by renaming it or copying it to a different directory.

1. Copy your custom firmware `pa17isdk.out` file into the directory `P:\EVMDA607_D607A003_Release`, where `P:\` is as defined in section "Software Deliverable Installation" in the *PAn SDK User's Guide*, and the directory name is that present in the deliverable. The target directory contains `pa17isdk.out` provided with the PA17 SDK Deliverable.

Note: ***Choice of Directory***

In step [1] above, and step [5] below, we mention the directory `P:\EVMDA607_D607A003_Release`. This corresponds to the location of the `pa17isdk.out` file for the DA607 processor. However, if the DA601 processor is being used, you would use directory `P:\EVMDA607_D601A003_Release`.

---

2. Open a shell for entering commands, such as a DOS shell.
3. Change to the `P:\` drive:
   ```
   C:\> P:
   ```
4. Add the bin directory `P:\bin` to the path:
   ```
   P:\> SET PATH=%PATH%;P:\bin;
   ```
5. Change to the directory containing your custom firmware `pa17isdk.out` file:
   ```
   cd P:\EVMDA607_D607A003_Release
   ```
6. To create a hex equivalent of the custom firmware, enter:
   ```
   coff2hex pa17isdk.out fl_program.hex
   ```
7. This creates file `fl_program.hex`, which can be directly substituted for the Program section provided with the PA17 SDK Deliverable (`fl_pa17isdk.hex`).
8. To program the custom firmware into flash memory using `fl_program.hex`, perform the Firmware Download procedure described in the next section.

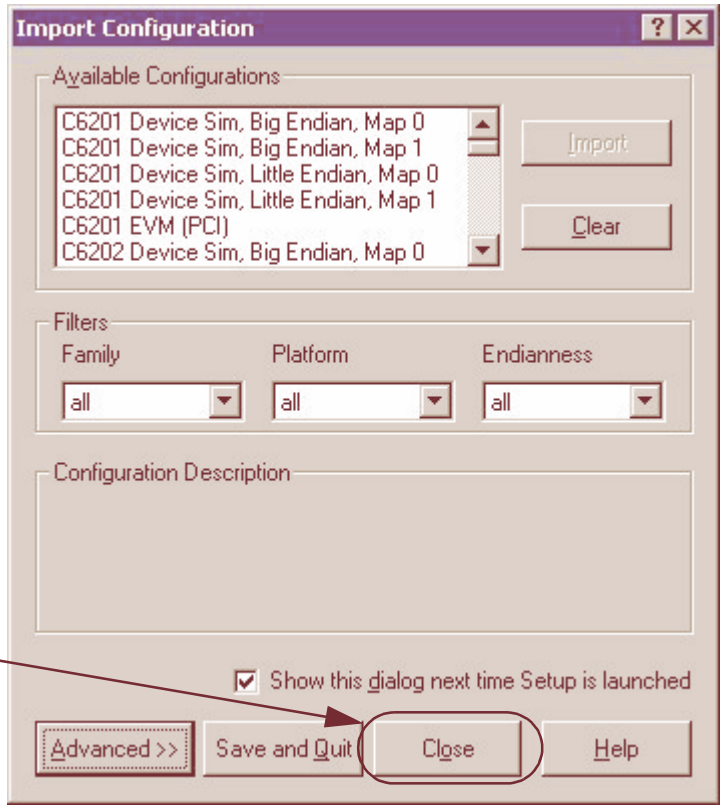
B.3.1.4 Firmware Download

This section describes how to download firmware to the flash memory. This process is also referred to above as "program" and more commonly as "burn."

Firmware download requires a change to the CCS configuration established during installation of XDS510PP. Once the new configuration is saved, a DOS batch file provided for running the entire download session is executed. The procedure is as follows:

1. As performed for installation of XDS510PP, run the CCS Setup Program. This can be done by double-clicking the CCS Setup icon on the desktop. The CCS Setup icon is shown in Figure B-2 (CCS Desktop Icons).
2. After you start the program, you will be in the **Import Configuration** dialog box. Select the **Close** button as indicated in Figure B-12.

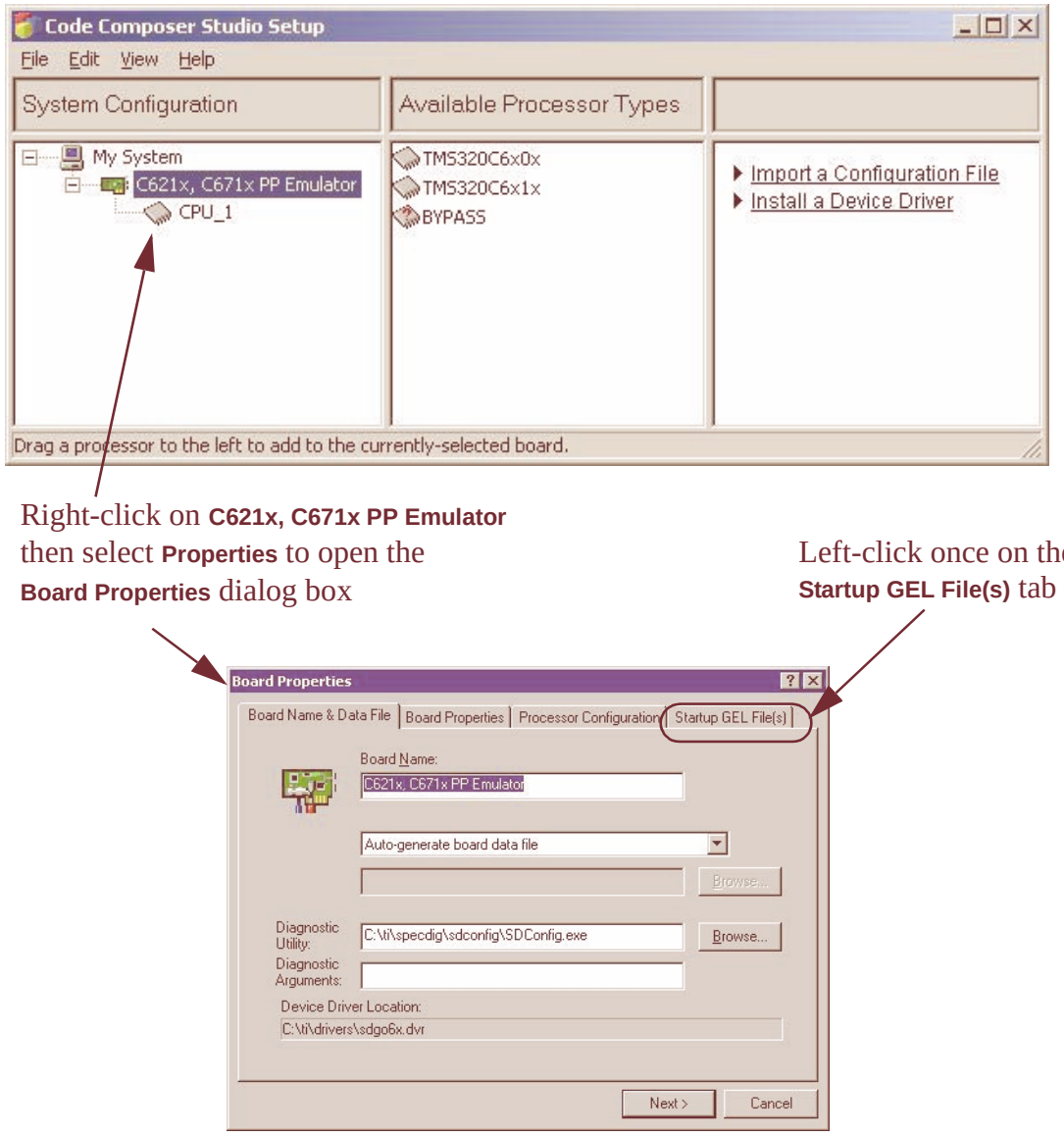
FIGURE B-12 Import Configuration Dialog



Click on this button to close the Import Configuration dialog box.

3. As shown in Figure B-13 (Board Properties Dialog), in the “System Configuration” pane, right-click once on “C621x, C671x PP Emulator” entry to pop open a menu. Then left-click once on **Properties...**.This opens the **Board Properties** dialog box.

FIGURE B-13 Board Properties Dialog



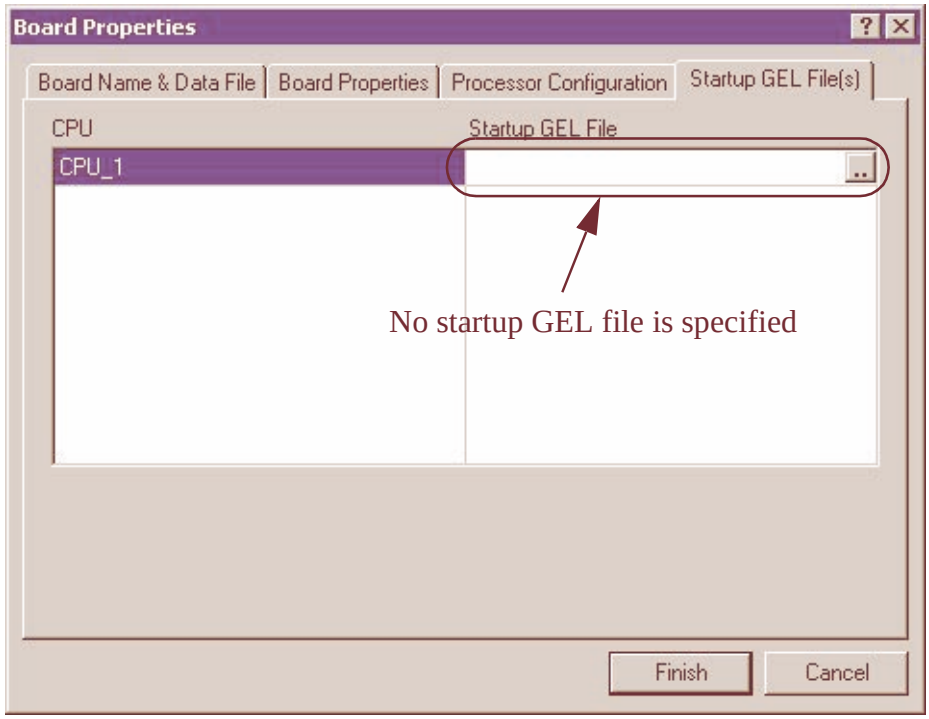
4. Left-click once on the **Startup GEL File(s)** tab.
5. In the window immediately left of the "..." button, under "Startup GEL File," delete the file name there (expected default file):

```
T:\pa\gel\board_EVMDA601_D601.gel # PC601
T:\pa\gel\board_EVMDA607_D607.gel # PC607
```

As shown in Figure B-14 (No Startup GEL File), no startup GEL file should be specified. Not specifying a Startup GEL File allows the DOS batch file to select a different GEL file for firmware download processing.

Note: This same procedure, steps [1]-[8], should be used to restore the above Startup GEL File after firmware download is completed and you're ready to restore the default CCS configuration for XDS510PP.

FIGURE B-14 No Startup GEL File



6. Click **finish** in the "Board Properties" dialog.
7. Use the **File / Save** menu to save the setup.
8. Exit the Code Composer Studio Setup application. Answer "No" when asked whether or not to Start Code Composer Studio on exit.
9. Open a shell for running the DOS batch file and establish the required path by executing:

```
c:\ti\DosRun.bat
SET PATH=%PATH%;c:\ti\bin\utilities\flashburn
```

10. Change to directory `P:\`, where `P:\` is as defined in section "Software Deliverable Installation" in the *PAn User's Guide* and *PAn SDK User's Guide*.
11. Run the DOS batch file flash programming tool:

    To download firmware provided with PA17 (default flash program in DA6XXEVM) or PA17 SDK Deliverable, enter:

```
bin\burn\burn_D601 # PC601
bin\burn\burn_D607 # PC607
```

    To download custom firmware for the default device (specified by the device in the default build configuration), enter:

```
bin\burn_custom
```

12. After approximately one to two minutes, messages similar to these should be displayed:

```
## MM/DD/YY HH:MM:SS
FlashBurn of 1024 bytes is complete.
Time: 451 mSec.
## MM/DD/YY HH:MM:SS
FlashBurn of 61624 bytes is complete.
Time: 16464 mSec.
## MM/DD/YY HH:MM:SS
FlashBurn of 62 bytes is complete.
Time: 380 mSec.
```

    The sequence of “FlashBurn of *n* bytes is complete” messages correspond to Boot Loader, Program and Version ID sections.

13. Provided no errors are indicated, firmware download is successful. Checksum verification of all data written is performed as part of this firmware download process.

    Note: If desired, steps [1]-[8] of this procedure can now be performed to restore default CCS configuration for XDS510PP. Step 5 requires selecting the Startup GEL File for restoration, as it was deleted for firmware download.

APPENDIX C *Flash Boot*

This appendix discusses “booting” the DA6XXEVM from flash memory on the DA601/607 Processor Card. This is usually referred to as the “Flash Boot” process.

C.1 Flash Boot Overview

The Processor Card of the DA6XXEVM contains on-board non-volatile flash memory that can be used to “boot” the system. This means that you can put a program into the flash memory and have the system automatically run that code when you turn on the power to the DA6XXEVM.

C.1.1 Flash Programming

Before programming[20] the flash, you need to have a working program that you want to place into flash memory. Once you have that, you can use a batch file and programs we have created to help you program the code into flash memory. The batch file uses the “FlashBurn” utility program produced by Texas Instruments to actually download the code to the flash memory, via an XDS JTAG Emulator. Once flash is programmed, you can disconnect the JTAG emulator then “Flash Boot” the DA6XXEVM to automatically run your code following power-on.

For details of how to program DA6XXEVM flash memory, see “Field Upgrade” on page 81 in Appendix B.

C.1.2 Flash Boot Operation

After flash memory is programmed, Flash Boot may be performed by following these procedures found in the Quick Start Guide:

20. Programming of flash memory is also referred to as “burning.”

- Setup Hardware: DA6XXEVM on page QSG-2.
- Setup Software: Run From Flash on page QSG-6.

APPENDIX D *Errata*

This appendix gives a list of errata associated with the DA6XXEVM.

D.1 Errata

1. If a Host MCU is not available for use with the DA6XXEVM audio card's Host MCU interface, i.e. before the customer's microcontroller code is working, the customer may use the DA6XX SPI/I^{2}C Card (SPI Dongle) as an alternate method of host communications (to send alpha code).
2. The DA6XXEVM Audio Card's Digital Input Receiver (DIR1703) may have problems with low quality digital input signals. For example, using the Digital Audio Labs CardDeluxe at a 32 kHz sample rate directly into the Audio Card does not work properly; if the signal is cleaned up first by going through a coaxial-to-optical box (for example), it works properly.
3. The volume up/down buttons are not yet supported in software.
4. Current support is for only one output stream.
5. Due to a relay issue on the board, a "pop" may be heard in the output audio at power-on if speakers are connected. It is advised that speakers be disabled during power-on to prevent potential damage.
6. Repeatedly plugging/unplugging the digital input during playback may cause audible distortion. The same may occur if the digital input selector switch is toggled repeatedly during playback. Development for correction of this issue is in progress.

APPENDIX E *Revision History of this Manual*

This Appendix lists changes from previous releases of this document.

E.1 V1.1, April 28, 2004

- Corrected hypertext links to be functional.
- Updated PC601 schematic diagrams.
- Updated references/paths to files provided on the distribution CD, as the structure of the CD image has been reorganized; updated associated text descriptions.

E.2 V1.0, February 18, 2004

- Corrected Parallel Debug Mode procedure in Quick Start Guide and CCS Appendix; minor cleanup of Quick Start Guide.
- Updated PC601 and PC607 chapters.
- Updated DSP version references from A002 (IROM2) to A003 (IROM3).
- First standard release.

E.3 V0.91, February 16, 2004

- Updated all imported schematic diagrams.
- Final preliminary release.

E.4 V0.9, February 11, 2004

- Updated CCS description and added FlashBurn description, including update of version number requirements for both; added description and links for TMS320C67x Fast RTS Library and User Guide; updated procedures, including GEL file names and locations, to suit PA17 SDK 040205 release.
- Updated PC601 Chapter; corrected device identification for PC601 (flash, accessible SDRAM size).
- Added link to new DA6XXEVM data sheet.

E.5 V0.8, January 28, 2004

- Minor correction of SPI/I^2C Card procedures: updated Quick Start Guide; added Firmware Download procedure for programming SPI/I^2C Card's flash memory; reorganized Chapter 5, DA6XX SPI/I^2C Card (SPI Dongle) Details.

E.6 V0.7, January 21, 2004

- Added SPI/I^2C Card documentation and procedures: major revision of Quick Start Guide; major revision of Chapter 5, DA6XX SPI/I^2C Card (SPI Dongle) Details.

E.7 V0.6, January 14, 2004

- Updated DA6XXEVM Audio Loopback Project.
- Updated Appendix B (Code Composer Studio for DA6XXEVM): Section B.3.1 (Field Upgrade).
- Updated Appendix C (Flash Boot).
- Minor text updates throught the entire document.

E.8 V0.5, January 7, 2003

First preliminary release.