

**AT89STK-03**

**Starter Kit for AT8xC5122 & AT83C5123**

.....  
**Hardware User Guide**





# Table of Contents

---

## **Section 1**

Introduction .....	1-1
1.1 Features .....	1-1

---

## **Section 2**

Hardware Description .....	2-5
2.1 Block Diagram .....	2-5
2.2 Power Supply .....	2-6
2.2.1 Regulator Output Voltage Selection & Power On .....	2-7
2.3 C51 Settings .....	2-7
2.3.1 INT1 Switch .....	2-7
2.3.2 EA Jumper .....	2-7
2.3.3 ALE led .....	2-7
2.3.4 Reset .....	2-7
2.3.5 Crystal .....	2-7
2.4 Features Description .....	2-7
2.4.1 LEDs Settings .....	2-7
2.4.2 USB Peripheral .....	2-8
2.4.3 SPI .....	2-8
2.4.4 RS232 .....	2-8
2.4.5 SIM Alternate Card .....	2-8
2.4.6 PLL Filter .....	2-8
2.4.7 EEPROM Settings .....	2-9
2.5 2 x 48 headers .....	2-9

---

## **Section 3**

Part Programming .....	3-11
3.1 Bootloader DPW .....	3-12
3.1.1 ISP mode (In System Programming ) .....	3-12
3.1.2 Execution mode .....	3-13
3.2 Bootloader DRV .....	3-16
3.2.1 ISP Mode .....	3-16
3.2.2 Execution mode .....	3-16
3.3 Bootloader DSU .....	3-21
3.3.1 ISP Mode .....	3-21
3.3.2 Execution mode .....	3-21

---

**Section 4**

Schematics ..... 4-27

    4.1 Bill of Materials.....4-35

---

**Section 5**

Document Revision History ..... 5-37



# Section 1

---

## Introduction

This document describes the MCU demonstration board included in AT89STK-03 Starter Kit dedicated to the AT8xC5122 & AT83C5123 Smart Card Reader microcontrollers.

This board enables to test the microcontrollers by using demonstration software and/or start the development of applications.

---

### 1.1 Features

The AT8xC5122 & AT83C5123 demonstration board provides the following features:

- Support of three packages :
  - PLCC 68
  - VQFP 64
  - PLCC 28
- On-board configurable power supply using a low drop regulator powered by :
  - an external power converter
  - an external battery
  - the USB interface
- ISO7816 Smart Card connector
- SIM Smart Card connector
- On-board reset, INT1, In-System Programming and USB detach switches
- EEPROM 24C256 socket for code storage
- Power, ALE, RS232 Rx /Tx, and 7 general purpose LEDs
- USB, SPI and RS232 hardware connectors
- Test points
- Two 48x2 headers

Figure 1-1. AT8xC5122 & AT83C5123 Evaluation Board

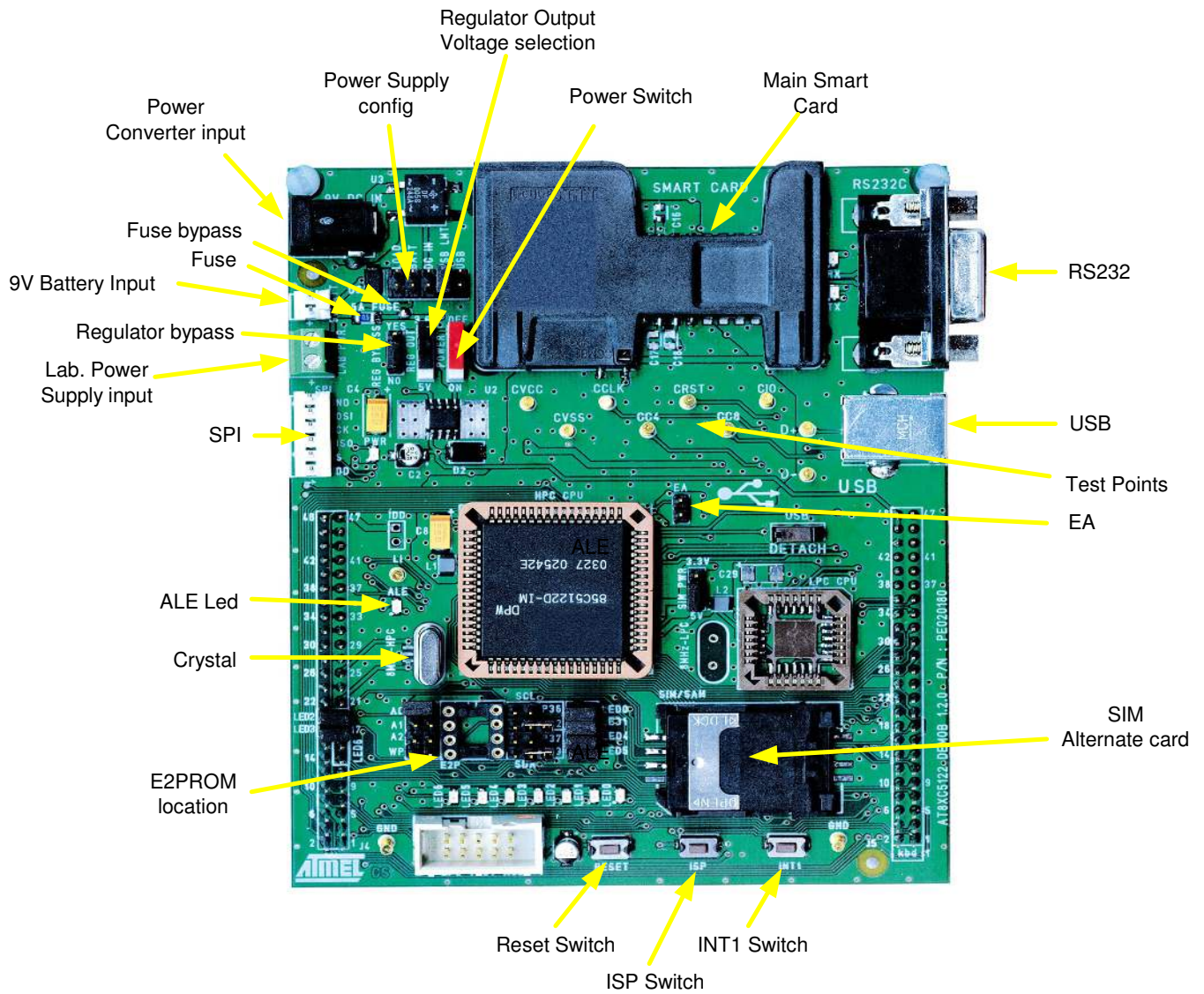


Figure 1-2. Default Configuration : Component Side - Jumper Config

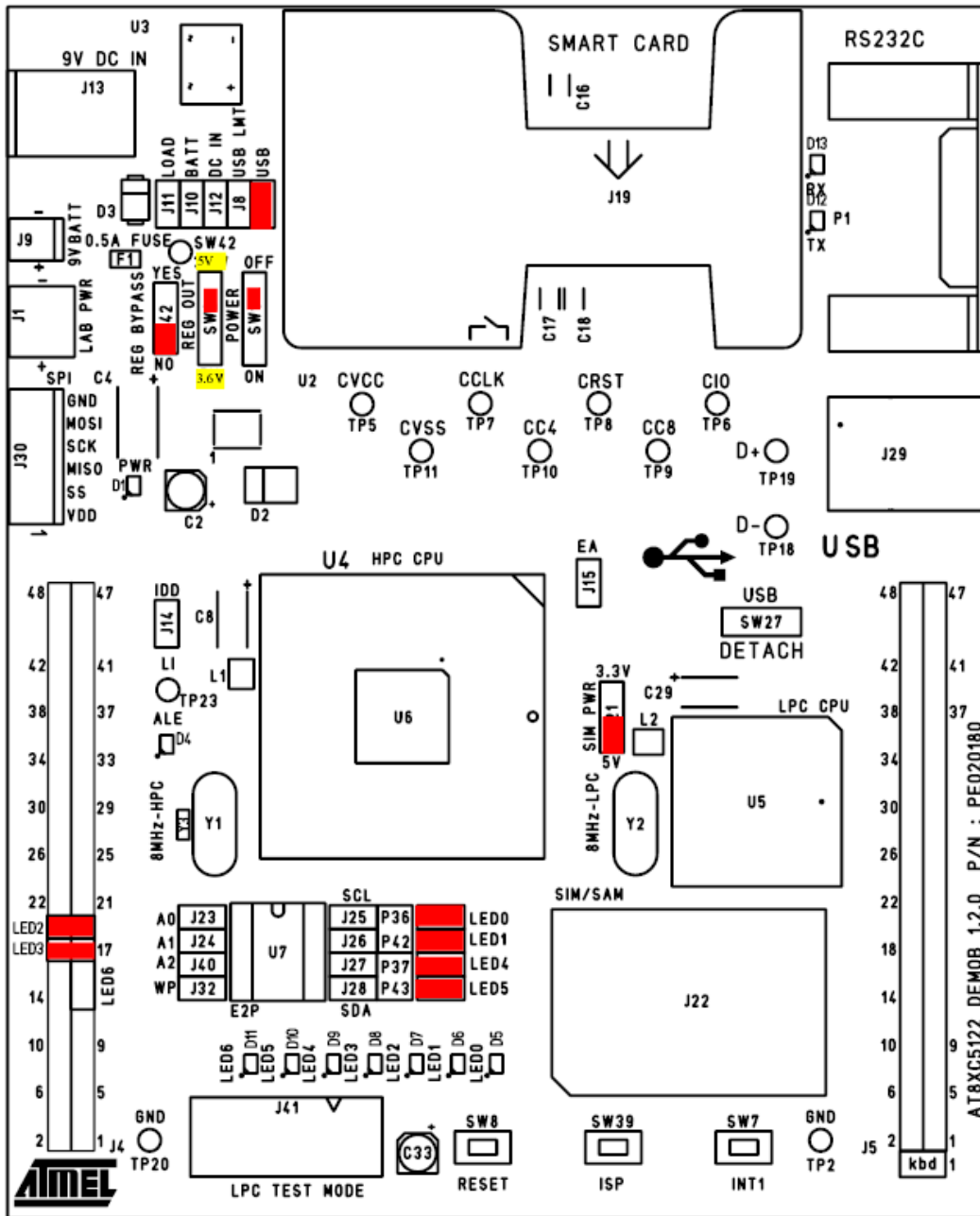
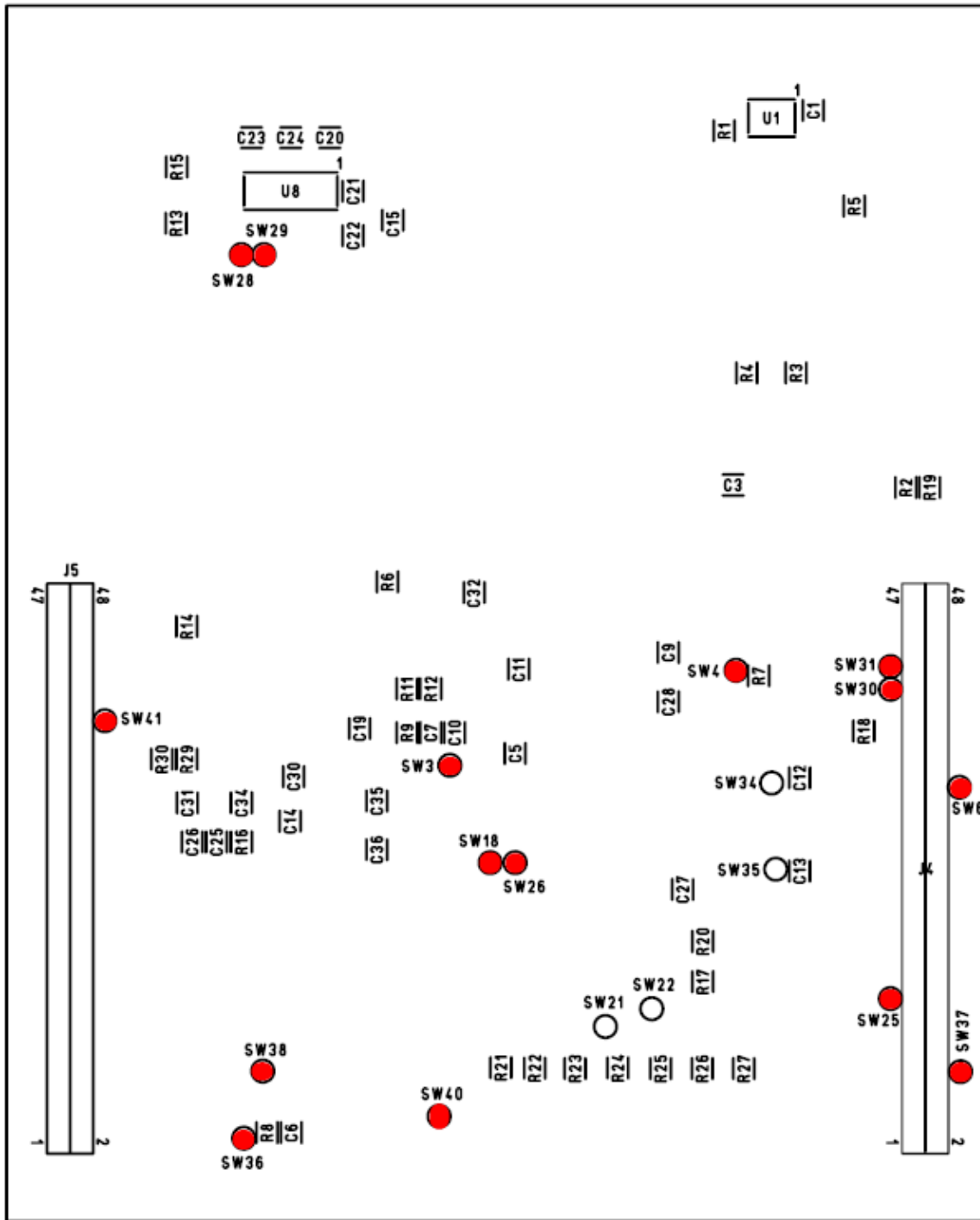
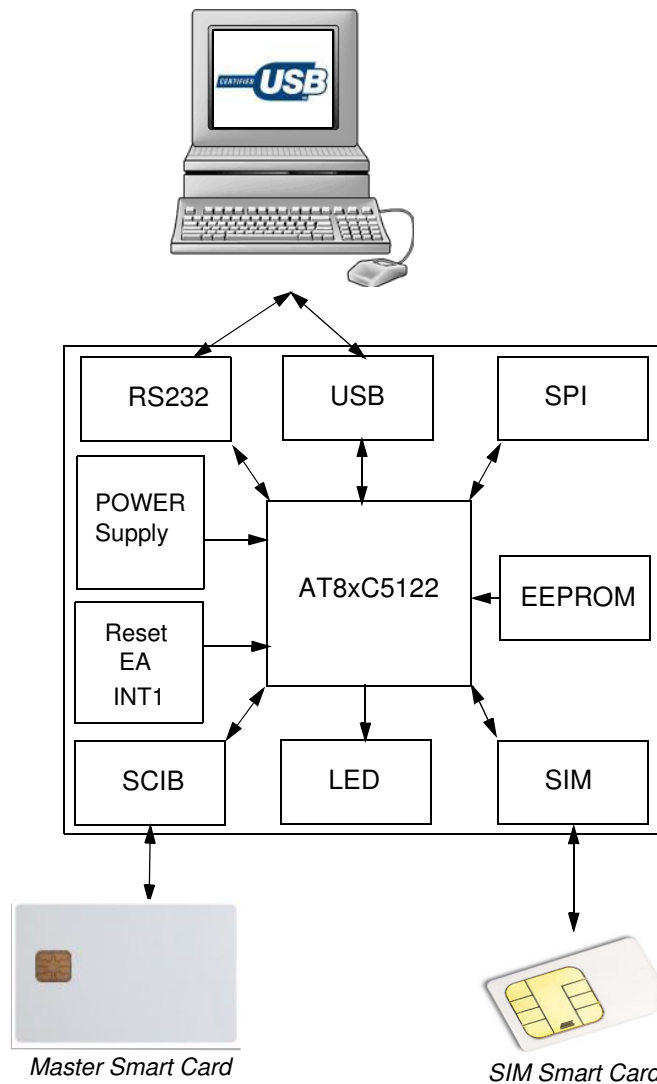


Figure 1-3. Default Configuration : Solder Side - Jumper Configuration



### 2.1 Block Diagram

Figure 2-1. AT8xC5122 & AT83C5123 demonstration board components





## 2.2 Power Supply

The on-board power supply electronics enable various power supply configurations.

The power source can be:

- VBus from USB interface (USB connector)
- An external power converter (9V DC connector)
- 9V Battery (9V BATT connector)
- Laboratory Power Supply (LAB PWR connector)

The power source selection is done by means of a jumper. The jumper must be set accordingly with the selected power source. The input of laboratory power supply is not controlled by the jumper. This source is directly applied to the input of the low drop regulator.

Any power supply (including the Laboratory Power Supply) can be turned on/off by using the red switch (POWER). Once the power is established, the power LED (PWR) becomes red. A low drop regulator is used to deliver a 3.6V or a 5V voltage.

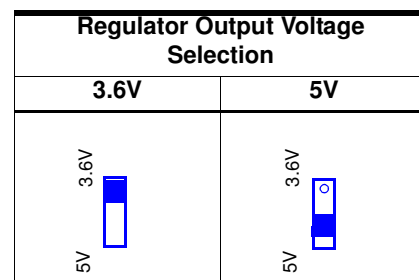
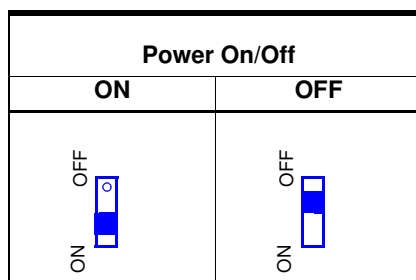
The low drop regulator can be bypassed by means of “REG. BYPASS” jumper. In this case the power source is directly applied on the MCU. Extreme care must be taken in using this feature and it is recommended to use it only with a Laboratory Power Supply perfectly stable and controlled.

The demonstration board is protected by a fuse soldered on board. In case the fuse blows, it is possible to bypass it by adding a solder gout on the solder pad next to the fuse on the component side of the board until its replacement. Once the fuse is replaced, the solder gout must be removed from the bypass pad.

**Table 2-1.** Power supply specifications and configurations

Jumper Position	Power Supply	Min	Max	Unit	Comments
USB	USB interface	4.3	5.5	V	- Default configuration - Applicable when CPU demonstration board is used alone
USB_LMT	USB interface				- Applicable when CPU demonstration board is plugged in the C51 generic board - This configuration prevents the electronics of CPU and C51 generic board from drawing inrush currents from USB interface
DC_IN	External power supply	7.0	9.0	V	Applicable when the board is fed by an external power converter
BATT	Battery	6.5	9.0	V	Applicable when the board is fed by an external battery

## 2.2.1 Regulator Output Voltage Selection & Power On



The 3.6V output can be changed by means of R4 resistor (solder side). The output voltage equation is:

$$R4 = R3 \times \frac{V_{ref}}{V_{out} - V_{ref}}$$

$V_{ref} = 1.20V$ ,  $R3 = 1M\Omega$

## 2.3 C51 Settings

### 2.3.1 INT1 Switch

The INT1 switch can be disabled by removing the solder gout from the solder pad SW36 on the solder side of the CPU demonstration board.

### 2.3.2 EA Jumper

EA Jumper	Description	Comments
ON	Code fetched from an external memory	
OFF	Code fetched from MCU's internal memory	Default configuration

### 2.3.3 ALE led

The ALE led reports the MCU state. It can be disabled by removing the solder gout from the solder pad SW4 on solder side of the board.

### 2.3.4 Reset

The push button is provided to generate a warm reset.

### 2.3.5 Crystal

The 8 Mhz crystal can be removed in order to be placed next to the appropriate package (PLCC68 or PLCC28). The load capacitors are disabled by default because the internal oscillator of the MCU has been designed to work without them. However these capacitors can be enabled by adding a solder gout on the solder pads SW34 and SW35 on the solder side of the board.

## 2.4 Features Description

### 2.4.1 LEDs Settings

11 LEDs are provided.

Name	Color	Description
PWR	Red	Power Supply status
ALE	Green	MCU activity status - This led is enabled by a solder gout on the solder pad SW4
LED0	Green	General purpose led - Connected to P3.2 - Enabled by LED0 jumper
LED1	Green	General purpose led - Connected to P3.4 - Enabled by LED1 jumper

Name	Color	Description
LED2	Green	General purpose led - Connected to P3.6 - Enabled by LED2 jumper or SW21 solder pad
LED3	Green	General purpose led - Connected to P3.7 - Enabled by LED3 jumper or SW22 solder pad
LED4	Green	General purpose led - Connected to P4.3 (Only on PLCC68 / VQFP64 packages) - Enabled by LED4 jumper
LED5	Green	General purpose led - Connected to P4.4 (Only on PLCC68 / VQFP64 packages) - Enabled by LED5 jumper
LED6	Green	General purpose led - Connected to P4.5 (Only on PLCC68 / VQFP64 packages) - Enabled by LED6 jumper or SW25 solder pad
TX	Green	Serial Transmission Line Status - Connected to TXD / P3.1 - Enabled by SW28 solder pad
RX	Red	Serial Reception Line Status - Connected to RXD / P3.0 - Enabled by SW29 solder pad

**2.4.2 USB Peripheral**

All the required hardware to develop a USB firmware is available and includes:

- a USB connector
- 2 test points D+ / D-
- a DETACH pushbutton which enables to connect / disconnect the pull-up on D+ and then to simulate an USB Attach/Detach operation. This avoids removals and insertions of the USB cable.

**2.4.3 SPI**

An SPI interface is available.

**2.4.4 RS232**

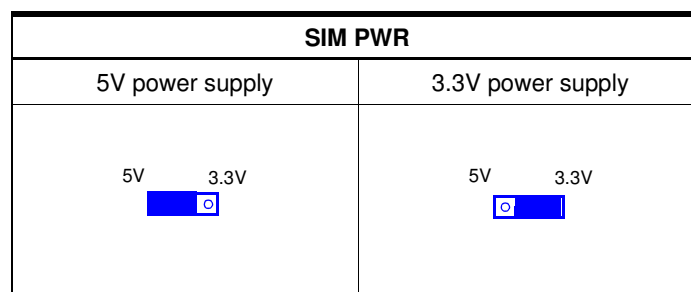
All the required hardware to manage a RS232C interface is available. Tx and Rx can be connected to the LEDs (Rx and Tx) by means of SW28 and SW29 solder pads on solder side of the board.

- 9 pin SUBD connector
- Transceiver MAX202

**2.4.5 SIM Alternate Card**

The following hardware is supplied:

- SIM connector
- power supply jumper



**2.4.6 PLL Filter**

1. PLCC68 / VQFP64

PLL Filter is composed by C7, C10, R9

2. PLCC28

PLL filter is composed by C25, R16, C26.

**2.4.7 EEPROM Settings**

This MCU demonstration board supports different versions of the AT8XC5122 and AT83C23 microcontrollers:

- Flash versions
- CRAM versions
- ROM versions

and different versions of embedded bootloaders (DPW, DRV, DSU).

PLCC68 package is only available in CRAM version but it possible to have it function in flash mode by adding an external 256Kb (32KB x 8) E2PROM on the 8 pin DIP socket. It is possible to write protect the E2PROM by adding a jumper on the WP location.

The demonstration board is configured to work with parts implementing the latest version of the bootloader (DSU or upper versions). If parts implementing older versions are used, headers must be added on A0 ,A1, P36, P37, P42, P43 locations and the configuration of the board must changed accordingly.

**Table 2-2.** E2PROM settings

Configuration of the board	Bootloader		
	DPW	DRV	DSU
Jumper on A0	Yes	No	No
Jumper on A1	No	No	No
Jumper on P36	Yes	No	No
Jumper on P37	Yes	No	No
Jumper on P42	Yes	No	No
Jumper on P43	Yes	No	No
Solder gout on SW18	No	Yes	Yes
Solder gout on SW26	No	Yes	Yes

Refer to bootloader datasheets for further information.

**2.5 2 x 48 headers**

These headers support the optional C51 generic board.



## Section 3

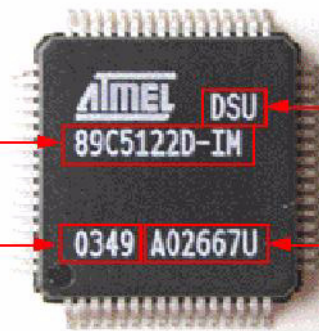
# Part Programming

This section applies only to versions AT85C5122, AT85EC5122 and AT89C5122. The method to program the part depends on the version of the bootloader which is embedded in a ROM memory. The bootloader version is defined by three letters which can be found on the part marking.

### Marking

Part Number

Date Code



Bootloader code

Lot number

### 3.1 Bootloader DPW

Ensure you are familiar with the DPW bootloader datasheet before programming the part.

Configure the MCU demonstration board following Table 2-2 on page 9.

#### 3.1.1 ISP mode (In System Programming)

When no E2PROM is connected to the MCU, the bootloader works automatically in ISP mode. It waits for instructions coming from an external program like FLIP through serial or USB interfaces. FLIP is a software interface which enables to communicate with the bootloader in order to program the part by transferring the code in the internal CRAM (volatile program memory) and execute the code.

**Table 3-1.** Programming the MCU with FLIP

Interface	Perform the following actions	Comments
USB	<p><b>DEMO BOARD CONFIGURATION</b></p> <ul style="list-style-type: none"> <li>- Set the POWER switch in OFF position</li> <li>- Remove any E2P present on the DIP socket</li> <li>- Connect a USB cable between demo board and PC. If an RS232C cable is present, the USB interface is selected by default</li> <li>- Configure the power jumper accordingly to power source which supplies the demo board</li> <li>- Set the POWER switch in ON position</li> </ul> <p>--&gt; <b>Result</b> : PWR ON and ALE leds are switched on</p> <p>--&gt; <b>Trouble shooting</b> : check if 8 Mhz quartz is correctly fitted</p> <p><b>PROGRAMMING PART</b></p> <ul style="list-style-type: none"> <li>- Run FLIP</li> <li>- Select part at8xc5122 for FLIP 1.8.8 or part at8xc5122_dpw for FLIP 2.x.x</li> <li>- Establish an USB communication</li> </ul> <p>--&gt; <b>Results</b> :</p> <ul style="list-style-type: none"> <li>- Bootloader Vers = 1.1.2</li> <li>- DeviceBoot Ids = 02 00</li> <li>- Start Application button becomes red</li> </ul> <ul style="list-style-type: none"> <li>- Select Erase, Blank check, Program and Verify options if part is running for the first time, otherwise just select Program and Verify</li> <li>- Load the code file (Intel hex format) into FLIP buffer</li> <li>- Click on Run button</li> <li>- Click on Start Application button</li> </ul>	<ul style="list-style-type: none"> <li>- Any E2P connected to C5122 prevents from working in ISP mode with FLIP</li> <li>- DPW bootloader is supported by FLIP since version V1.8.8</li> <li>- Area from 7FFD to 07FFF is reserved for bootloader configuration</li> </ul>

Table 3-1. Programming the MCU with FLIP

Interface	Perform the following actions	Comments
RS232C	<p><b>DEMO BOARD CONFIGURATION</b></p> <ul style="list-style-type: none"> <li>- Set the POWER switch in OFF position</li> <li>- Remove any USB cable connected to demoboard</li> <li>- Remove any E2P present on DIP socket</li> <li>- Configure the power jumper accordingly to power source wich supplies the demo board</li> <li>- Connect a RS232C cable between demo board and PC</li> <li>- Set the POWER switch in ON position</li> </ul> <p>--&gt; <b>Result</b> : PWR ON and ALE leds are switched on</p> <p>--&gt; <b>Trouble shooting</b> : check if 8 Mhz quartz is correctly fitted</p> <p><b>PROGRAMMING PART</b></p> <ul style="list-style-type: none"> <li>- Run FLIP</li> <li>- Select part at8xc5122 for FLIP 1.8.8 or part at8xc5122_dpw for FLIP 2.x.x</li> <li>- Select a RS232C communication</li> <li>- Establish a serial communication with the appropriate COM number and a baudrate comprised between 1200 and 38400 .</li> </ul> <p>---&gt; <b>Results</b> :</p> <ul style="list-style-type: none"> <li>- Bootloader Vers = 1.1.2</li> <li>- DeviceBoot Ids = 02 00</li> <li>- Start Application button becomes red</li> </ul> <ul style="list-style-type: none"> <li>- Select Erase, Blank check, Program and Verify options if part is running for the first time, otherwise just select Program.</li> <li>- Load the code file (Intel hex format) into FLIP buffer</li> <li>- Click on Run button</li> <li>- Click on Start Application button</li> </ul>	

### 3.1.2 Execution mode

When the E2PROM is connected to ports P4.2 and P4.3 or P3.6 and P3.7, the boot-loader works automatically in execution mode. At reset, it transfers the code from the E2PROM to the internal CRAM (volatile program memory) and execute it. The E2PROM must contain a valid code. As the bootloader does not support E2PROM programming functions, the E2PROM must be programmed by external programmer or by a specific code which is programmed in CRAM and which behaves as a bootloader. This code programs the E2PROM by the code received from FLIP.

**Table 3-2.** Programming the E2PROM with FLIP

Interface	Perform the following actions	Comments
USB interface	<p><b>DEMO BOARD CONFIGURATION</b></p> <ul style="list-style-type: none"> <li>- Set the POWER switch in OFF position</li> <li>- Remove any E2P present on DIL8 socket</li> <li>- Connect a USB cable between demo board and PC. If an RS232C cable is present, the USB interface is selected by default</li> <li>- Configure the power jumper accordingly to power source which supplies the demo board</li> <li>- Set the POWER switch in ON position</li> </ul> <p>--&gt; <b>Result</b> : PWR ON and ALE leds are switched on</p> <p>--&gt; <b>Trouble shooting</b> : check if 8 Mhz quartz is correctly fitted</p> <p><b>PROGRAMMING EXTERNAL E2P</b></p> <ul style="list-style-type: none"> <li>- Run FLIP</li> <li>- Select part at8xc5122 for FLIP 1.8.8 or part at8xc5122_dpw for FLIP 2.x.x</li> <li>- Establish a USB communication</li> <li>- Select Program and Verify options</li> <li>- Load c5122-at24c-loader-0_6_0.hex file</li> <li>- Click on Run button</li> <li>- Click on Start Application button</li> <li>- Re-establish an USB communication</li> </ul> <p>--&gt; <b>Results</b> :</p> <ul style="list-style-type: none"> <li>- Bootloader Vers =1.1.2</li> <li>- DeviceBoot Ids = E0 00</li> <li>- Start Application button becomes red</li> </ul> <ul style="list-style-type: none"> <li>- Select Program and Verify options</li> <li>- Load the user code into FLIP buffer</li> <li>- Edit FLIP buffer</li> <li>- Configure the area from 07FFD to 07FFF according to the wanted options (Refer to bootloader datasheet). A common configuration is : 07FFD = FDh 07FFE = MSB (user code size) 07FFF = LSB (user code size)</li> <li>- Select Whole buffer in buffer options menu (to program the whole buffer in the E2P)</li> <li>- Insert an 24C256 E2P device into the demoboard's DIP socket without switching off the demo board, otherwise the loader code will be lost.</li> <li>- Click on Run button (the loader running in CRAM programs the external E2P with the code received from FLIP)</li> <li>- No need to click on start application</li> <li>- Reset the MCU by means of RESET switch to execute the code programmed in the external E2P. (If external E2P is present, the bootloader downloads the code from external E2P and execute it)</li> </ul>	DPW bootloader does not provide E2P programming functions. It is necessary to download and execute a loader code into CRAM to program the external E2P by means of FLIP



Table 3-2. Programming the E2PROM with FLIP

Interface	Perform the following actions	Comments
RS232C interface	<p><b>DEMO BOARD CONFIGURATION</b></p> <ul style="list-style-type: none"> <li>- Set the POWER switch in OFF position</li> <li>- Remove any E2P present on DIL8 socket</li> <li>- Connect a USB cable between demo board and PC. If an RS232C cable is present, the USB interface is selected by default</li> <li>- Configure the power jumper accordingly to power source which supplies the demo board</li> <li>- Set the POWER switch in ON position</li> </ul> <p>--&gt; <b>Result</b> : PWR ON and ALE leds are switched on</p> <p>--&gt; <b>Trouble shooting</b> : check if 8 Mhz quartz is correctly fitted</p> <p><b>PROGRAMMING EXTERNAL E2P</b></p> <ul style="list-style-type: none"> <li>- Run FLIP</li> <li>- Select part at8xc5122 for FLIP 1.8.8 or part at8xc5122_dpw for FLIP 2.x.x</li> <li>- Select a RS232C communication</li> <li>- Establish a serial communication with the appropriate COM number and a baudrate comprised between 1200 and 38400 .</li> <li>- Select Program and Verify options</li> <li>- Load c5122DPW-at24c-loader-0_6_0.hex file</li> <li>- Click on Run button</li> <li>- Click on Start Application button</li> <li>- Establish a serial communication with the appropriate COM number and a baudrate comprised between 1200 and 38400 .</li> </ul> <p>---&gt; <b>Results</b> :</p> <ul style="list-style-type: none"> <li>- Bootloader Vers =1.1.2</li> <li>- DeviceBoot Ids = E0 00</li> <li>- Start Application button becomes red</li> </ul> <ul style="list-style-type: none"> <li>- Select Program and Verify options</li> <li>- Load the user code into FLIP buffer</li> <li>- Edit FLIP buffer</li> <li>- Configure the area from 07FFD to 07FFF according to the wanted options (Refer to bootloader datasheet). A common configuration is :  07FFD = FDh  07FFE = MSB (user code size)  07FFF = LSB (user code size)</li> <li>- Select Whole buffer in buffer options menu (to program the whole buffer in the E2P)</li> <li>- Insert an 24C256 E2P device into the demoboard's DIL8 socket without switching off the demo board, otherwise the loader code will be lost.</li> <li>- Click on Run button (the loader running in CRAM programs the external E2P with the code received from FLIP)</li> <li>- No need to click on start application</li> <li>- Reset the MCU by means of RESET switch to execute the code programmed in the external E2P. (If external E2P is present, the bootloader downloads the code from external E2P and execute it)</li> </ul>	

---

**3.2 Bootloader DRV** Ensure you are familiar with the DRV bootloader datasheet before programming the part.

Configure the MCU demonstration board following Table 2-2 on page 9.

**3.2.1 ISP Mode** ■ CRAM version

The bootloader enters automatically in ISP mode with this version after power up or reset sequences and waits for commands from FLIP as there is no code available to run. When the bootloader is running in ISP mode, the ALE led blinks.

■ FLASH version

The ISP mode must be invoked manually by holding down the ISP switch (P3.7=0) during power up or reset sequences. When the bootloader is running in ISP mode, the ALE led blinks.

**3.2.2 Execution mode** ■ CRAM version

The execution mode must be invoked manually from FLIP. ALE Led remains on permanently during this mode.

■ FLASH version

The execution mode does not request specific action. The application code runs automatically after power up or reset sequence. ALE Led remains on permanently during this mode.

Notes: 1. The external E2PROM works only with the PLCC68 package. It enables to obtain a flash equivalent solution as the PLCC68 is only available in CRAM version. For the others packages, order the appropriate part as the E2PROM is implemented inside the package.

2. The programming of internal or external E2PROM is automatically handled by the bootloader.

Table 3-3. Programming with bootloader DRV

External E2P	Interface	Perform the following actions	Comments
Not Present (AT85 like)	USB	<p><b>DEMO BOARD CONFIGURATION</b></p> <ul style="list-style-type: none"> <li>- Set the POWER switch in OFF position</li> <li>- Connect an USB cable between demo board and PC. If an RS232C cable is present, the USB interface is selected by default</li> <li>- Configure the power jumper accordingly to power source which supplies the demo board</li> <li>- Set the POWER switch in ON position.</li> </ul> <p>--&gt; <b>Result</b> : PWR ON led is switched on and ALE led is blinking</p> <p>--&gt; <b>Trouble shooting</b> : Check if 8 Mhz quartz is correctly fitted</p> <p><b>PROGRAMMING PART</b></p> <ul style="list-style-type: none"> <li>- Run FLIP</li> <li>- Select part at8xc5122</li> <li>- Establish a USB communication</li> </ul> <p>---&gt; <b>Results</b> :</p> <ul style="list-style-type: none"> <li>- Bootloader Vers = 1.2.0</li> <li>- DeviceBoot Ids = 00 03</li> <li>- Two start applications buttons are displayed in red colour : <ul style="list-style-type: none"> <li>- CRAM Start Application button (for at85c5122 parts)</li> <li>- FLASH Start Application (for at89c5122 parts)</li> </ul> </li> </ul> <ul style="list-style-type: none"> <li>- Select Erase, Blank check, Program and Verify options if part is running for the first time, otherwise just select Program and Verify</li> <li>- Load the code file (Intel hex format) into FLIP buffer</li> <li>- Edit FLIP buffer and check if configuration byte located at address 07FFD is correct (Refer to bootloader datasheet) - Only security bit can be applied to this part version</li> <li>- Select Whole buffer in buffer options menu</li> <li>- Click on Run button</li> <li>- Click on CRAM Start Application button (FLASH Start Application is not applicable)</li> </ul>	<ul style="list-style-type: none"> <li>- DRV bootloader is supported by FLIP since version V2.1.1</li> <li>- DRV bootloader is unable to differentiate CRAM and FLASH parts. This is why two Start applications are available. The right start application must be clicked accordingly to the used part</li> <li>- There are some areas reserved for the bootloader. Refer to bootloader datasheet.</li> </ul>

Table 3-3. Programming with bootloader DRV

External E2P	Interface	Perform the following actions	Comments
Present (AT89 like)	USB interface	<p><b>DEMO BOARD CONFIGURATION</b></p> <ul style="list-style-type: none"> <li>- Set the POWER switch in OFF position</li> <li>- Connect a USB cable between demo board and PC. If an RS232C cable is present, the USB interface is selected by default</li> <li>- Configure the power jumper accordingly to power source which supplies the demo board</li> <li>- Set the POWER switch in ON position.</li> <li>-- While holding down the ISP switch, press and release the reset switch to invoke the ISP mode</li> </ul> <p><b>(Note :</b> don't miss this step as it is very important to work around a USB bug when invoking the ISP mode)</p> <p>--&gt; <b>Result :</b> PWR ON led is switched on and ALE led is blinking</p> <p>--&gt; <b>Trouble shooting :</b> Check if 8 Mhz quartz is correctly fitted</p> <p><b>PROGRAMMING EXTERNAL E2P</b></p> <ul style="list-style-type: none"> <li>- Run FLIP</li> <li>- Select part at8xc5122</li> <li>- Establish a USB communication</li> </ul> <p>---&gt; <b>Results :</b> if a Software Security Error windows comes up, just click OK. This windows just warns that the part security bit is activated. In this case, an erase operation is mandatory before programming the part</p> <ul style="list-style-type: none"> <li>- Select Erase, Blank Check, Program and Verify options</li> <li>- Load the user code into FLIP buffer</li> <li>- Edit FLIP buffer</li> <li>- Configure the area from 07FFD to 07FFF according to the wanted options (Refer to bootloader datasheet). A common configuration is : 07FFD = 08h 07FFE = MSB (user code size) 07FFF = LSB (user code size)</li> <li>- Select Whole buffer in buffer options menu</li> <li>- Click on Run button</li> <li>- Click on FLASH Start Application button to execute the user code</li> <li>- Check if E2P code is correctly executed after a power on or a reset operation</li> </ul>	<ul style="list-style-type: none"> <li>- Setting hardware condition P3.7=0 during power on or reset will force the bootloader to enter in ISP mode with FLIP</li> <li>- DRV bootloader supports E2P write functions to program the external E2P.</li> <li>- The area from 07FFD to 07FFF can be configured by means of FLIP's GUI. It is recommended to check the configuration by editing the FLIP buffer from 07FFF to 07FFF</li> </ul>

Table 3-3. Programming with bootloader DRV

External E2P	Interface	Perform the following actions	Comments
Not present (AT83 like)	RS232C	<p><b>DEMO BOARD CONFIGURATION</b></p> <ul style="list-style-type: none"> <li>- Set the POWER switch in OFF position</li> <li>- Remove any USB cable connected to demoboard</li> <li>- Connect a RS232C cable between demo board and PC</li> <li>- Configure the power jumper accordingly to power source wich supplies the demo board</li> <li>- Set the POWER switch in ON position.</li> </ul> <p>--&gt; <b>Result</b> : PWR ON and ALE leds are switched on</p> <p>--&gt; <b>Trouble shooting</b> : check if 8 Mhz quartz is correctly fitted</p> <p><b>PROGRAMMING PART</b></p> <ul style="list-style-type: none"> <li>- Run FLIP</li> <li>- Select part at88c51rc2</li> <li>- Select a RS232C communication</li> <li>- Establish a serial communication with the appropriate COM number and a baudrate of 115200 baud</li> </ul> <p>---&gt; <b>Results</b> :</p> <ul style="list-style-type: none"> <li>- Bootloader Vers = 1.2.0</li> <li>- DeviceBoot Ids = 00 03</li> <li>- Start Application button becomes red</li> </ul> <p><b>WARNING : Don't care about and don't change the other available options except the reset option close to the start application command</b></p> <ul style="list-style-type: none"> <li>- Select Erase, Blank Check, Program and Verify options if part is running for the first time, otherwise just select Program (Verify command is long to execute)</li> <li>- Load the code file (Intel hex format) into FLIP buffer</li> <li>- Edit FLIP buffer and check if configuration byte located at address 07FFD is correct (Refer to bootloader datasheet) - Only security bit can be applied to this part version</li> <li>- Select Whole buffer in buffer options menu</li> <li>- Click on Run button</li> <li>- Select reset option (if this option is not selected, the user code is not executed)</li> <li>- Click on Start Application button</li> </ul>	<ul style="list-style-type: none"> <li>- Versions of FLIP equal or greater than 2.1.1 do not support RS232C interface for at8xc5122 device. A workaround consists to select part at89c51rc2 and RS232C interface.</li> <li>- Bootloader DRV only supports a baudrate of 115200 baud for RS232C interface.</li> </ul>

Table 3-3. Programming with bootloader DRV

External E2P	Interface	Perform the following actions	Comments
Present (AT85 like)	RS232C interface	<p><b>DEMO BOARD CONFIGURATION</b></p> <ul style="list-style-type: none"> <li>- Set the POWER switch in OFF position</li> <li>- Remove any USB cable connected to demoboard</li> <li>- Connect a RS232C cable between demo board and PC</li> <li>- Configure the power jumper accordingly to power source wich supplies the demo board</li> <li>- While holding down the ISP switch, set the POWER switch to the ON position.</li> </ul> <p>--&gt; <b>Result</b> : PWR ON and ALE leds are switched on</p> <p>--&gt; <b>Trouble shooting</b> : Check if 8 Mhz quartz is correctly fitted</p> <p><b>PROGRAMMING EXTERNAL E2P</b></p> <ul style="list-style-type: none"> <li>- Run FLIP</li> <li>- Select part at88c51rc2</li> <li>- Select a RS232C communication</li> <li>- Establish a serial communication with the appropriate COM number and a baudrate of 115200 baud</li> </ul> <p>---&gt; <b>Results</b> :</p> <ul style="list-style-type: none"> <li>- Bootloader Vers = 1.2.0</li> <li>- DeviceBoot Ids = 00 03</li> <li>- Start Application button becomes red</li> </ul> <p><b>WARNING : Don't care about and don't change the other available options except the reset option close to the start application command</b></p> <ul style="list-style-type: none"> <li>- Select Erase, Blank Check, Program and Verify options if part is running for the time, otherwise avoid to select Verify command as it is long to execute</li> <li>- Load the user code into FLIP buffer</li> <li>- Edit FLIP buffer</li> <li>- Configure the area from 07FFD to 07FFF according to the wanted options (Refer to bootloader datasheet). A common configuration is : 07FFD = 08h 07FFE = MSB (user code size) 07FFF = LSB (user code size)</li> <li>- Select Whole buffer in buffer options menu</li> <li>- Click on Run button</li> <li>- Select reset option (if this option is not selected, the user code is not executed)</li> <li>- Click on the start application button to execute the user code</li> <li>- Check if E2P code is correctly executed after a power on or reset operation</li> </ul>	<ul style="list-style-type: none"> <li>- Setting hardware condition P3.7=0 during power on or reset will force the bootloader to enter in ISP mode with FLIP</li> <li>- DRV bootloader supports E2P write functions to program the external E2P.</li> </ul>

---

### 3.3 Bootloader DSU

The DSU bootloader fixes a USB bug present in the DRV bootloader. Therefore the DSU bootloader has the same features as the DRV bootloader.

Ensure you are familiar with the DSU bootloader datasheet before programming the part.

Configure the MCU demonstration board following Table 2-2 on page 9.

#### 3.3.1 ISP Mode

##### ■ CRAM version

The bootloader enters automatically in ISP mode with this version after power up or reset sequences and waits for commands from FLIP as there is no code available to run. When the bootloader is running in ISP mode, the ALE led blinks.

##### ■ FLASH version

The ISP mode must be invoked manually by holding down the ISP switch (P3.7=0) during power up or reset sequences. When the bootloader is running in ISP mode, the ALE led blinks.

#### 3.3.2 Execution mode

##### ■ CRAM version

The execution mode must be invoked manually from FLIP. ALE Led remains on permanently during this mode.

##### ■ FLASH version

The execution mode does not request specific action. The application code runs automatically after power up or reset sequence. ALE Led remains on permanently during this mode.

- Notes:
1. The external E2PROM works only with the PLCC68 package. It enables to get a flash equivalent solution as the PLCC68 is only available in CRAM version. For the others packages, order the appropriate part as the E2PROM is implemented inside the package.
  2. The programming of internal or external E2PROM is automatically handled by the bootloader

Table 3-4. Programming with bootloader DSU

E2PROM	Interface	Perform the following actions	Comments
Not present (AT85 like)	USB	<p><b>DEMO BOARD CONFIGURATION</b></p> <ul style="list-style-type: none"> <li>- Set the POWER switch in OFF position</li> <li>-- Connect a USB cable between demo board and PC. If an RS232C cable is present, the USB interface is selected by default</li> <li>- Configure the power jumper accordingly to power source which supplies the demo board</li> <li>- Set the POWER switch in ON position.</li> <li>--&gt; <b>Result</b> : PWR ON led is switched on and ALE led is blinking</li> <li>--&gt; <b>Trouble shooting</b> : Check if 8 Mhz quartz is correctly fitted</li> </ul> <p><b>PROGRAMMING PART</b></p> <ul style="list-style-type: none"> <li>- Run FLIP</li> <li>- Select part AT8XC5122</li> <li>- Establish a USB communication</li> <li>---&gt; <b>Results</b> : <ul style="list-style-type: none"> <li>- Bootloader Vers = 1.2.1</li> <li>- DeviceBoot Ids = 85 04</li> <li>- Start application buttons is displayed in red colour</li> <li>- The configuration of the firmware present on the device is displayed : <ul style="list-style-type: none"> <li>+ Boot Loader Jump Bit (checked=1, unchecked=0)</li> <li>+ Lock bit (checked=1, unchecked=0)</li> <li>+ Size bit (checked=1, unchecked=0)</li> <li>+ User Code Length = size of the user's code</li> </ul> </li> </ul> </li> </ul> <p><b>Note</b> : only lock bit parameter is applicable for this version of the part.</p> <ul style="list-style-type: none"> <li>- If a Software Security Error window comes up, just click OK. This window just warns that the part security bit is activated. In this case, an Erase operation is mandatory before programming the part</li> <li>- Select Erase, Blank check, Program and Verify options if part is running for the first time, otherwise just select Program and Verify</li> <li>- Load the code file (Intel hex format) into FLIP buffer</li> <li>- Click on Run button.</li> <li>- Click on Start Application button or press the reset switch to execute the code</li> </ul> <p><b>Note</b> : it is necessary to program the part after each power off as the CRAM is a volatile program memory</p>	<ul style="list-style-type: none"> <li>- DSU bootloader is supported by FLIP since version V2.2.0</li> <li>- There are some areas reserved for the bootloader. Refer to bootloader datasheet.</li> </ul> <p>This operating mode is applicable from FLIP 2.4.0</p> <p>85 means CRAM version</p>



Table 3-4. Programming with bootloader DSU

E2PROM	Interface	Perform the following actions	Comments
Present (AT89 like)	USB	<p><b>DEMO BOARD CONFIGURATION</b></p> <ul style="list-style-type: none"> <li>- Set the POWER switch in OFF position</li> <li>- Connect a USB cable between demo board and PC. If an RS232C cable is present, the USB interface is selected by default</li> <li>- Configure the power jumper accordingly to power source which supplies the demo board</li> <li>- While holding down the ISP switch, set the POWER switch in ON position to invoke the ISP mode</li> <li>--&gt; <b>Result</b> : PWR ON led is switched on and ALE led is blinking</li> <li>--&gt; <b>Trouble shooting</b> : Check if 8 Mhz quartz is correctly fitted</li> </ul> <p><b>PART PROGRAMMING</b></p> <ul style="list-style-type: none"> <li>- Run FLIP</li> <li>- Select part at8xc5122</li> <li>- Establish a USB communication</li> <li>---&gt; <b>Results</b> : <ul style="list-style-type: none"> <li>- Bootloader Vers = 1.2.1</li> <li>- DeviceBoot Ids = 89 04</li> <li>- Start application buttons is displayed in red colour</li> <li>- The configuration of the firmware present on the device is displayed : <ul style="list-style-type: none"> <li>+ Boot Loader Jump Bit (checked=1, unchecked=0)</li> <li>+ Lock bit (checked=1, unchecked=0)</li> <li>+ Size bit (checked=1, unchecked=0)</li> <li>+ User Code Length = size of the user's code</li> </ul> </li> <li>- If a Software Security Error window comes up, just click OK. This window just warns that the part security bit is activated. In this case, an Erase operation is mandatory before programming the part</li> </ul> </li> <li>- Select Erase, Blank Check, Program and Verify options</li> <li>- Load the user code into FLIP buffer</li> </ul> <p>FLIP parses the hex file and automatically updates the buffer's area from 07FFDh to 07FFFh - No operation is requested from the user, but the user is free to change the default parameters set by FLIP :</p> <ul style="list-style-type: none"> <li>+ Sizebit = 1</li> <li>+ User Code Length = size of the user's code</li> </ul> <ul style="list-style-type: none"> <li>- Click on Run button <ul style="list-style-type: none"> <li>+ Boot Loader Jump Bit, Lock bit, size bit and User Code Length are updated on the GUI of FLIP after part programming and according to the configuration set the FLIP's buffer from 07FFDh to 07FFFh</li> <li>+ When the part is programmed, it is possible to change the individual bits by checking the box on the GUI if FLIP. They are changed on the fly in the internal E2PROM by the bootloader.</li> </ul> </li> <li>+ <b>Note</b> : when the lock bit is set, the part is set in protected mode at once and it is not possible to read, write and perform any parameter change just after.</li> </ul> <p>The only way to remove the protection mode is to erase the part.</p> <ul style="list-style-type: none"> <li>- Click on the start application button or press the reset switch or power the demo board off and on to execute the code</li> </ul>	<ul style="list-style-type: none"> <li>- Setting hardware condition P3.7=0 during power on or reset will force the bootloader to enter in ISP mode with FLIP</li> <li>- DSU bootloader supports E2P write functions to program the external E2P.</li> <li>- The area from 07FFD to 07FFF can be configured by means of FLIP's GUI. It is recommended to check the configuration by editing the FLIP buffer from 07FFF to 07FFF</li> </ul>

Table 3-4. Programming with bootloader DSU

E2PROM	Interface	Perform the following actions	Comments
Not present (AT85 like)	RS232C	<p><b>DEMO BOARD CONFIGURATION</b></p> <ul style="list-style-type: none"> <li>- Set the POWER switch in OFF position</li> <li>- Remove any USB cable connected to demoboard</li> <li>- Connect a RS232C cable between demo board and PC</li> <li>- Configure the power jumper accordingly to power source wick supplies the demo board</li> <li>- Set the POWER switch in ON position.</li> </ul> <p>--&gt; <b>Result</b> : PWR ON led is switched on and ALE led is blinking</p> <p>--&gt; <b>Trouble shooting</b> : Check if 8 Mhz quartz is correctly fitted</p> <p><b>PART PROGRAMMING</b></p> <ul style="list-style-type: none"> <li>- Run FLIP</li> <li>- Select AT8XC5122 part</li> <li>- Select a RS232C communication</li> <li>- Establish a serial communication with the appropriate COM number and a baudrate of 115200 baud</li> </ul> <p>---&gt; <b>Results</b> :</p> <ul style="list-style-type: none"> <li>- Bootloader Vers = 1.2.1</li> <li>- DeviceBoot lds = 85 04</li> <li>- Start Application button becomes red</li> <li>- The configuration of the firmware present on the device is displayed : <ul style="list-style-type: none"> <li>+ Boot Loader Jump Bit (checked=1, unckecked=0)</li> <li>+ Lock bit (checked=1, unckecked=0)</li> <li>+ Size bit (ckecked=1, unckecked=0)</li> <li>+ User Code Lengh = size of the user's code</li> </ul> </li> </ul> <p><b>Note</b> : only lock bit parameter is applicable for this version of the part.</p> <ul style="list-style-type: none"> <li>- If a Software Security Error windows comes up, just click OK. This window just warns that the part security bit is activated. In this case, an Erase operation is mandatory before programming the part</li> </ul> <ul style="list-style-type: none"> <li>- Select Erase, Blank Check, Program and Verify options if part is running for the first time, otherwise just select Program as Verify command is long to execute</li> <li>- Load the code file (Intel hex format) into FLIP buffer</li> <li>- Click on Run button</li> <li>- Click on Start Application button or press the reset switch to execute the code</li> </ul> <p><b>Note</b> : it is necessary to program the part after each power off as the CRAM is a volatile program memory</p>	<ul style="list-style-type: none"> <li>- Versions of FLIP equal or greater than 2.1.1 do not support RS232C interface for at8xc5122 device. A workaround consists to select part at89c51rc2 and RS232C interface.</li> <li>- Bootloader DSU only supports a baudrate of 115200 baud for RS232C interface.</li> </ul> <p>85 means CRAM version</p>

Table 3-4. Programming with bootloader DSU

E2PROM	Interface	Perform the following actions	Comments
Present (AT89 like)	RS232C	<p><b>DEMO BOARD CONFIGURATION</b></p> <ul style="list-style-type: none"> <li>- Set the POWER switch in OFF position</li> <li>- Remove any USB cable connected to the demoboard</li> <li>- Connect a RS232C cable between demo board and PC</li> <li>- Configure the power jumper accordingly to power source wich supplies the demo board</li> <li>- While holding down the ISP switch ,set the POWER switch in ON position.</li> </ul> <p>--&gt; <b>Result</b> : PWR ON led is switched on and ALE led is blinking</p> <p>--&gt; <b>Trouble shooting</b> : Check if 8 Mhz quartz is correctly fitted</p> <p><b>PART PROGRAMMING</b></p> <ul style="list-style-type: none"> <li>- Run FLIP</li> <li>- Select part AT8XC5122</li> <li>- Select the RS232C communication</li> <li>- Establish a serial communication with the appropriate COM number and a baudrate of 115200 baud</li> </ul> <p>---&gt; <b>Results</b> :</p> <ul style="list-style-type: none"> <li>- Bootloader Vers = 1.2.1</li> <li>- DeviceBoot Ids = 89 04</li> <li>- Start Application button becomes red</li> <li>- The configuration of the fimware present on the device is displayed : <ul style="list-style-type: none"> <li>+ Boot Loader Jump Bit (checked=1, unckecked=0)</li> <li>+ Lock bit (checked=1, unckecked=0)</li> <li>+ Size bit (ckecked=1, unckecked=0)</li> <li>+ User Code Lengh = size of the user's code</li> </ul> </li> <li>- Select Erase, Blank Check, Program and Verify options if part is running for the first time, otherwise avoid to select Verify command as it is long to execute</li> <li>- Load the user code into FLIP buffer</li> </ul> <p>FLIP parses the hex file and automatically updates the buffer's area from 07FFDh to 07FFFh - No operation is requested from the user, but the user is free to change the default parameters set by FLIP :</p> <ul style="list-style-type: none"> <li>+ Size bit = 1</li> <li>+ User Code Lengh = size of the user's code</li> </ul> <ul style="list-style-type: none"> <li>- Click on Run button <ul style="list-style-type: none"> <li>+ Boot Loader Jmp Bit, Lock bit, size bit and User Cod Length are updated oln the GUI of FLIP after part programming and according to the configuration set the FLIP's buffer from 07FFDh to 07FFFh</li> <li>+ When the part is programmed, it is possible to change the individual bits by checking the box on the GUI if FLIP. They are changed on the fly in the internal E2PROM by the bootloader.</li> </ul> </li> <li>+ <b>Note</b> : when the lock bit is set, the part is set in protected mode only when the serial communication with FLIP is stopped and not at once like in USB interface. This is due to a minor bug in the bootloader. When the protection mode is active it is not possible to read, write and perform any parameter change. The only way to remove the protection mode is to erase the part.</li> <li>- Click on the start application button or press the reset switch or power the demo board off and on to execute the code</li> </ul>	<ul style="list-style-type: none"> <li>- Setting hardware condition P3.7=0 during power on or reset will force the bootloader to enter in ISP mode with FLIP</li> <li>- DSU bootloader supports E2P write functions to program the external E2P.</li> </ul> <p>89 means flash version</p> <p>Size bit =1 will force the bootloader to download only the valid code instead the whole 32K memory. This decreases the time that the bootloader needs to run the code.</p>



## Section 4

---

# Schematics

Figure 4-1. Power Supply Section

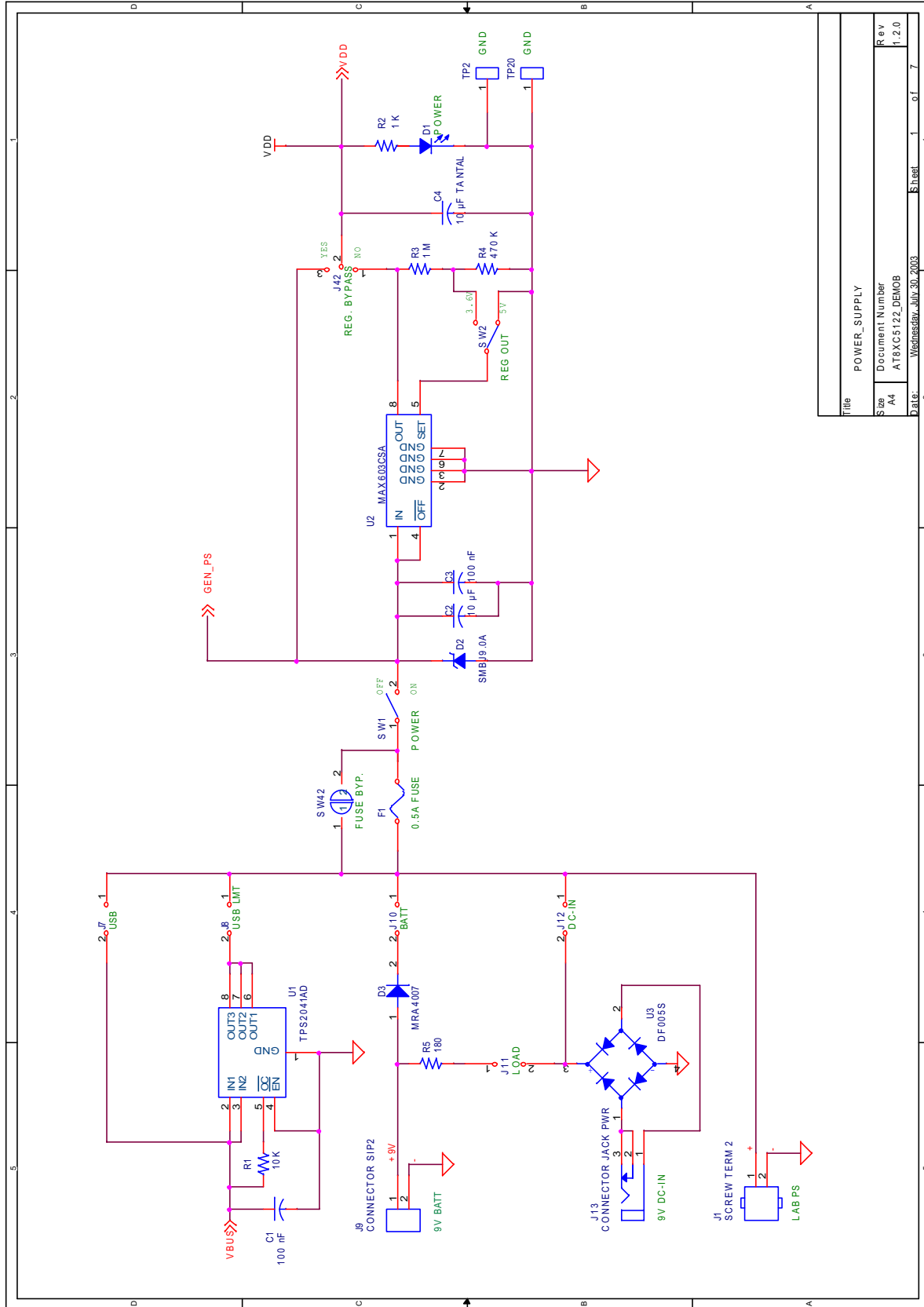
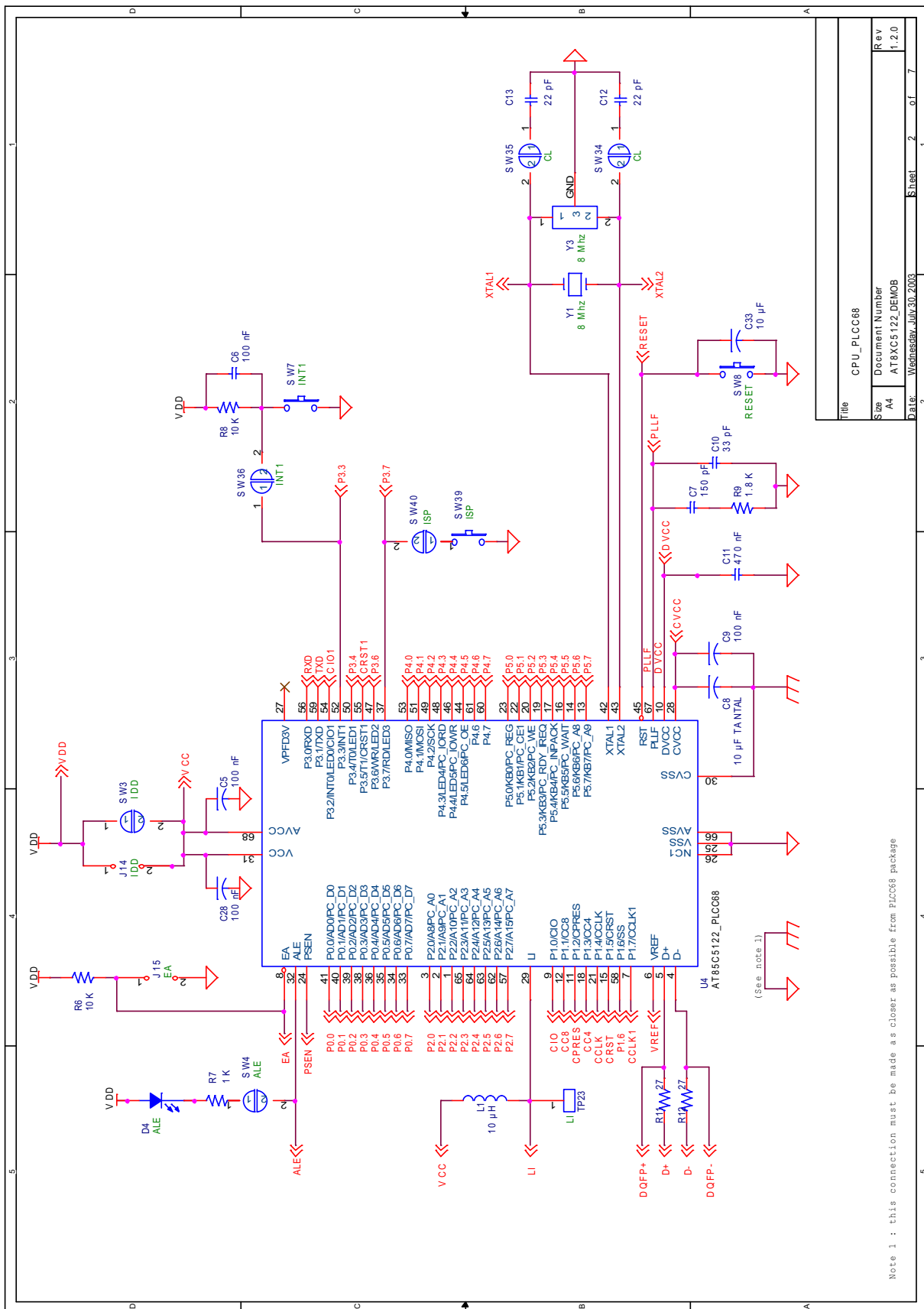


Figure 4-2. PLCC68 CPU Section

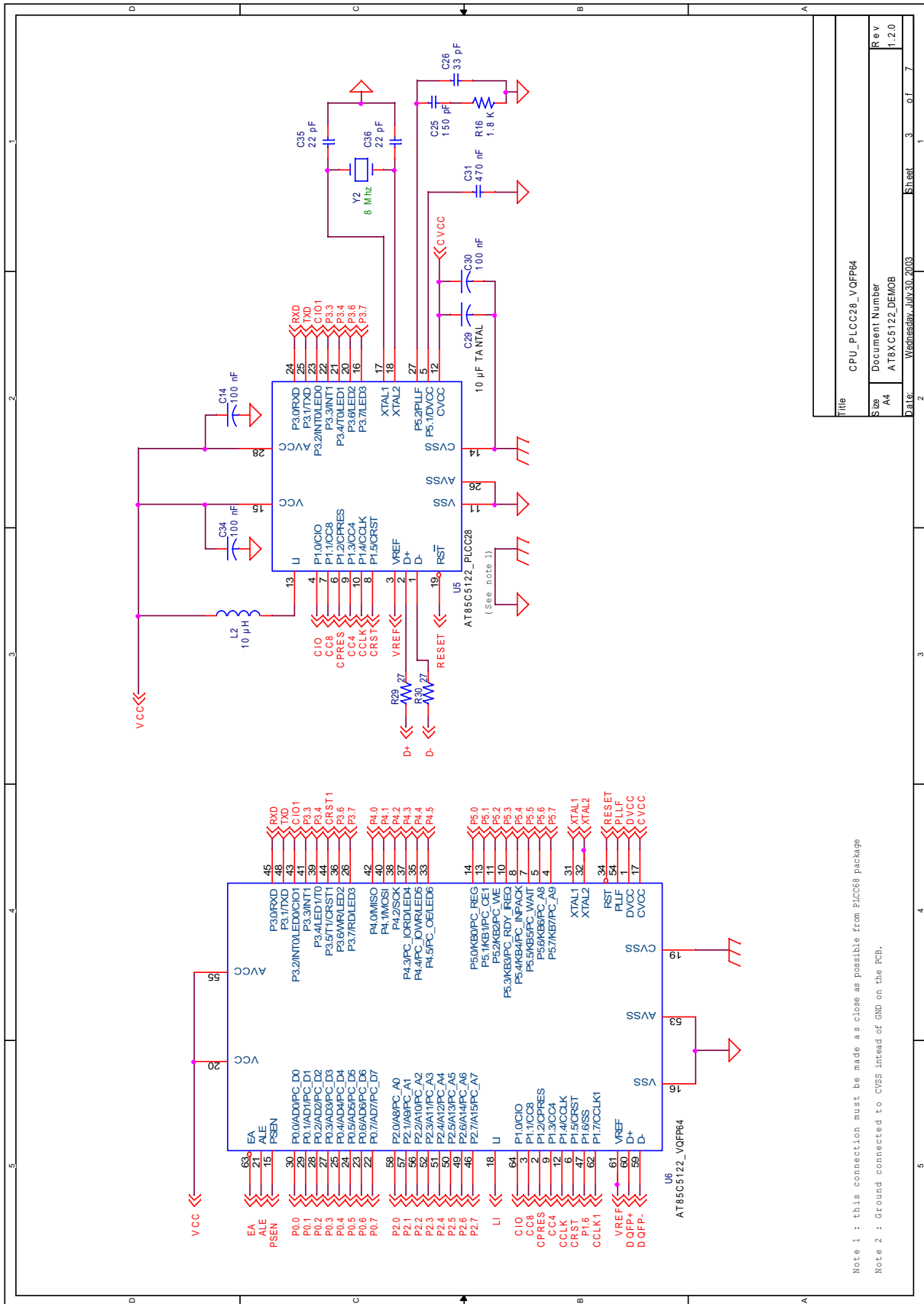


Note 1 : this connection must be made as closer as possible from PLCC68 package

Title		CPU_PLCC68	
Size	Document Number	AT89C5122_DEIMOB	
Rev	Date	1.2.0	of 7
	Sheet	2	of 7
	Date	Wednesday, July 30, 2003	



Figure 4-3. PLCC28\_VQFP64 CPU Section

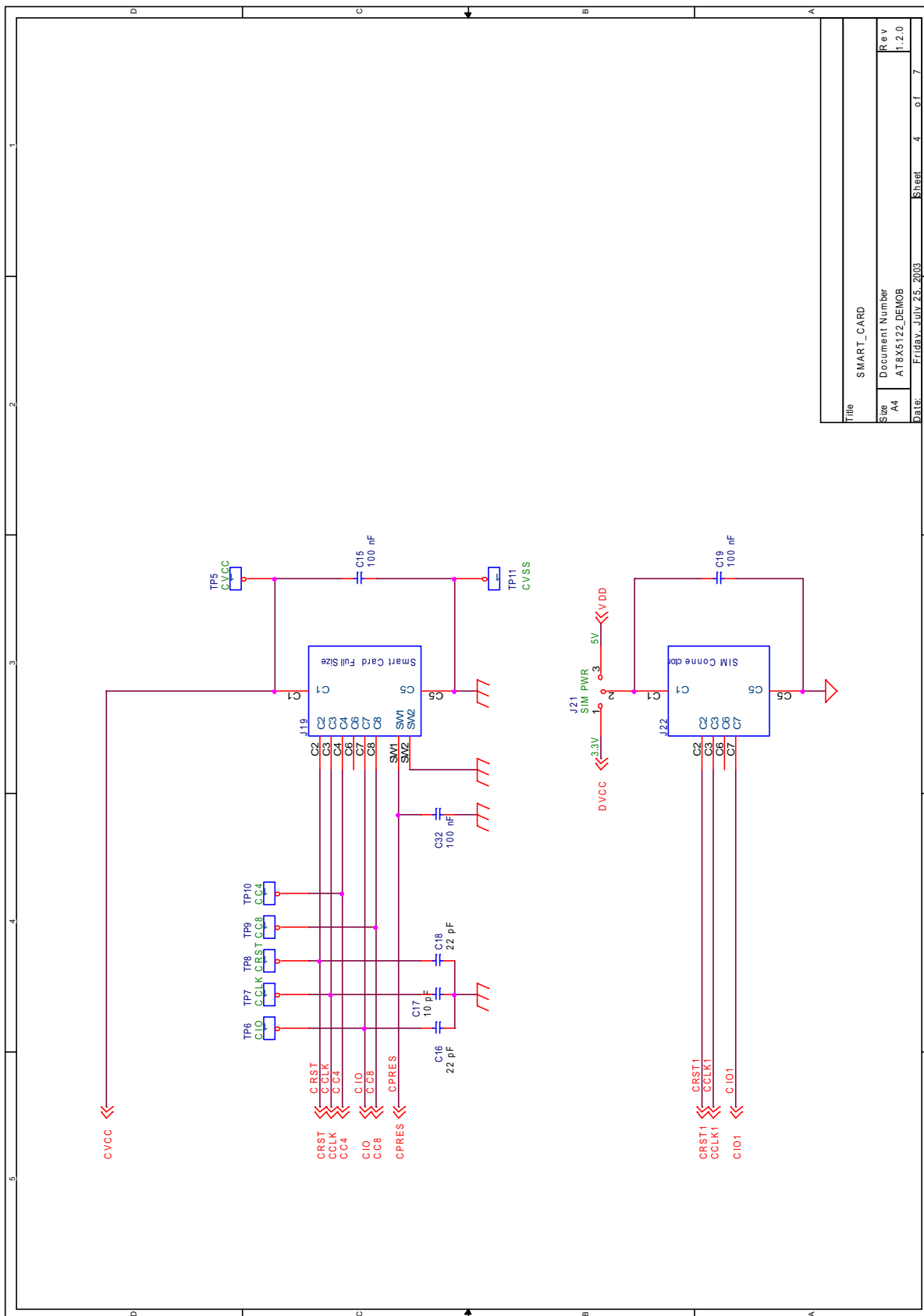


Title		CPU_PLCC28_VQFP64	
Size	Document Number	AT85C5122_DEMOB	
Rev	Date	Sheet	3 of 7
1.2.0	Wednesday, July 30, 2003		

Note 1 : this connection must be made as close as possible from PLCC68 package  
 Note 2 : Ground connected to CVSS instead of GND on the PCB.



Figure 4-4. Smart Card Schema



Title		SMART_CARD
Size	Document Number	AT89S122_DEMOB
Rev	1.2.0	
Date:	Fridav, July 25, 2003	
Sheet	4	of 7





Figure 4-5. E2P LEDs Section

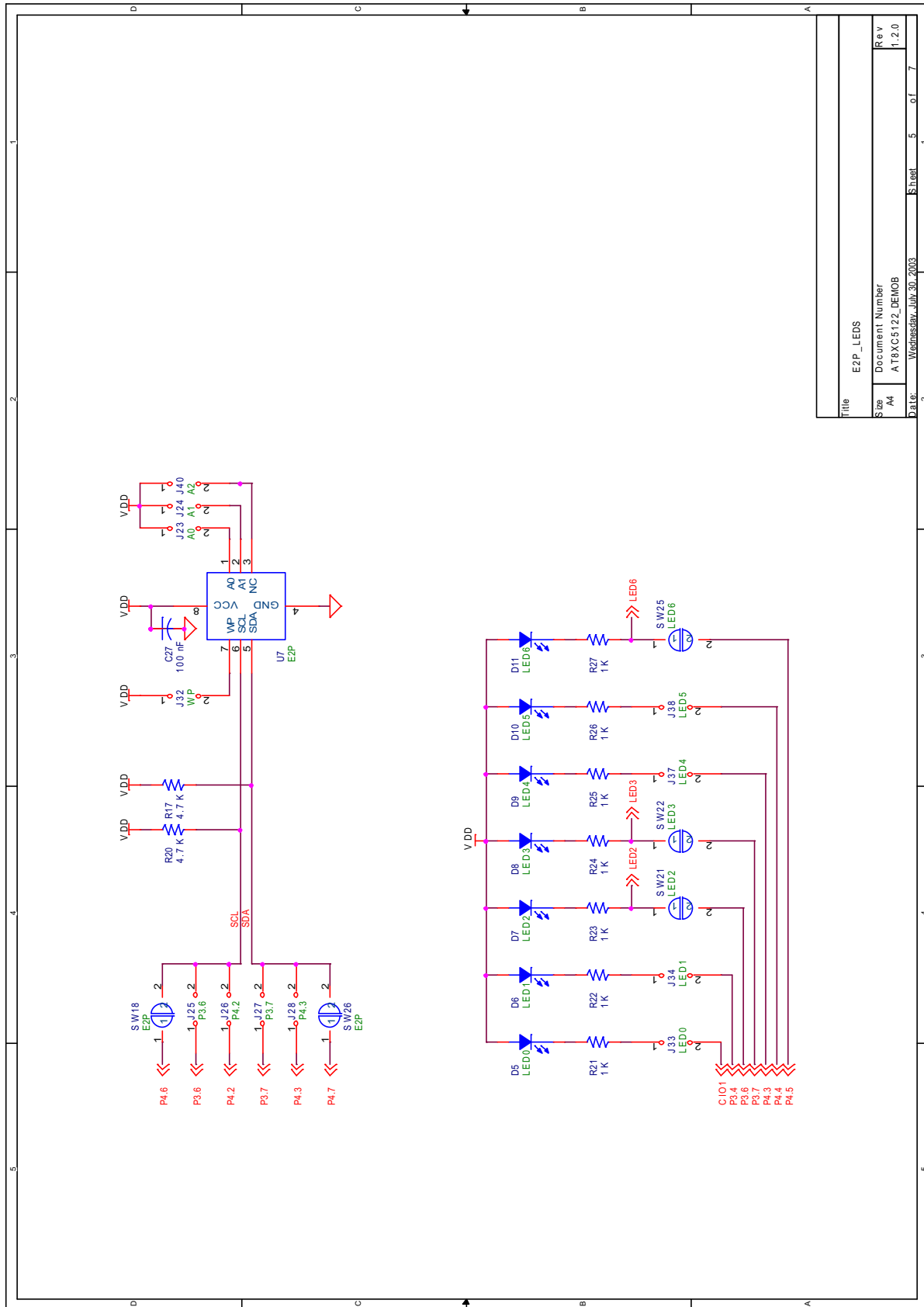
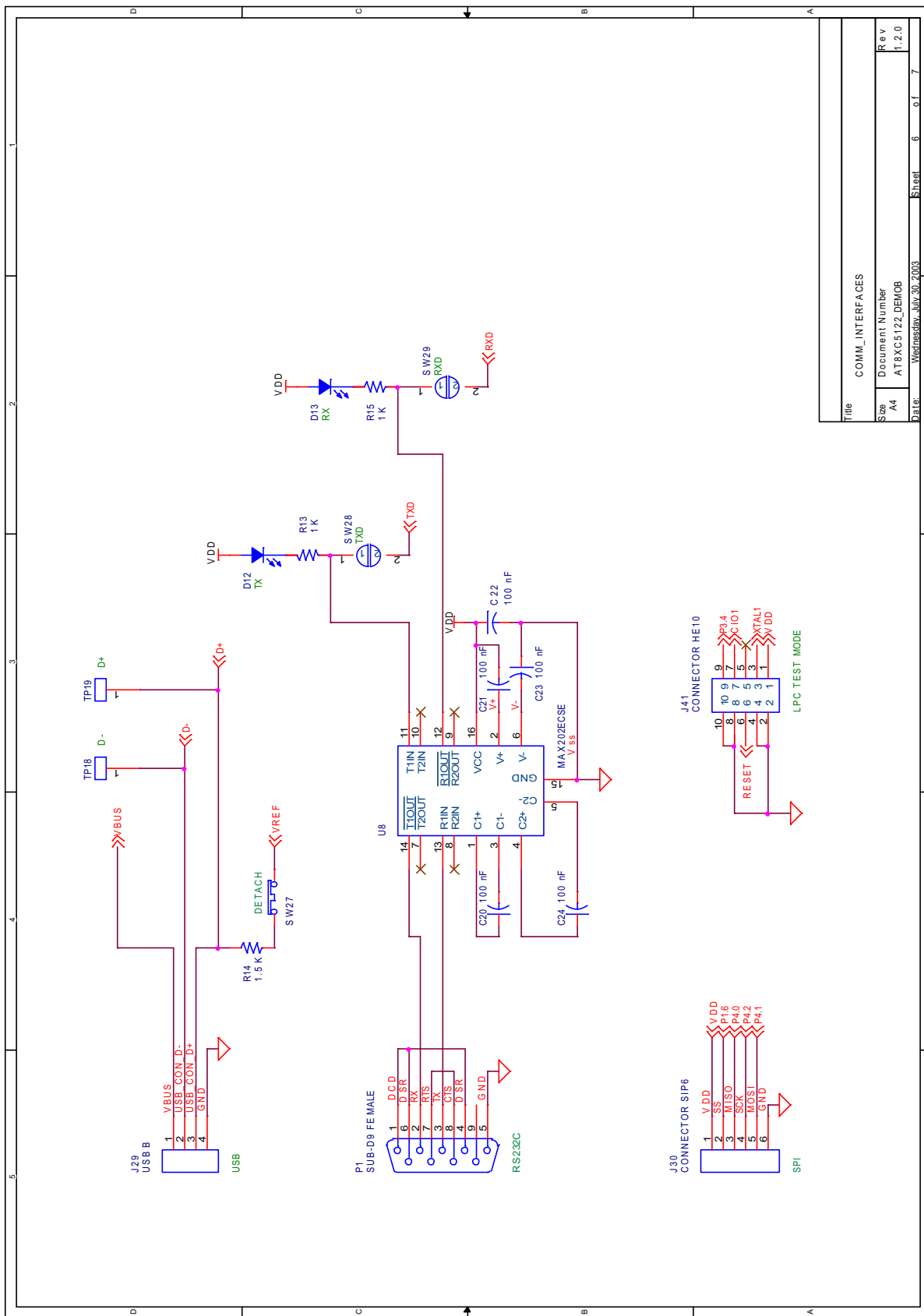


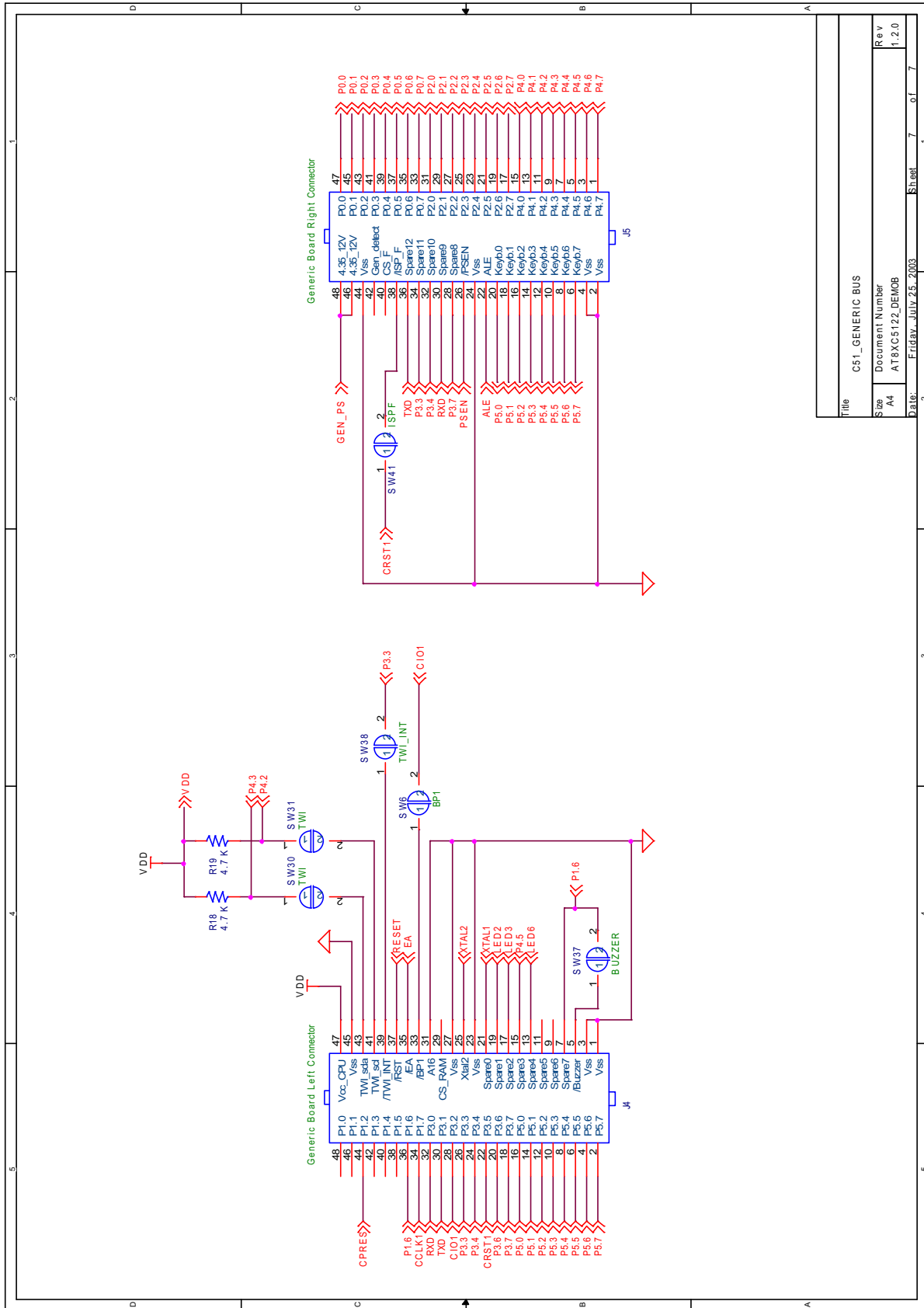
Figure 4-6. Comm Interfaces Section



Title		COMM_INTERFACES
Size	Document Number	A4 AT89STK-03 DEMOB
Date	Version	1.2.0
Sheet	of	6 7



Figure 4-7. C51 Generic Bus



## 4.1 Bill of Materials

Reference	Part	Part Description
C12,C13,C35,C36	22 pF	50V-5% Ceramic Capacitor
C26,C10	33 pF	50V-5% Ceramic Capacitor
C25,C7	150 pF	100V-5% Ceramic Capacitor
C1,C3,C5,C6,C9,C14,C15, C19,C20,C21,C22,C23,C24, C27,C28,C30,C32,C34	100 nF	50V-5% Ceramic Capacitor
C31,C11	470 nF	16V-10%-Ceramique
C2	10 $\mu$ F	16V +/-20%
C8,C4	10 $\mu$ F TANTAL	16V +/-10%
R11,R12,R29,R30	27 Ohms	1/16W-5% Resistor SMD
R5	180 Ohms	1/16W-5% Resistor SMD
R2,R7,R13,R15,R21,R22, R23,R24,R25,R26,R27	1 K Ohms	1/16W-5% Resistor SMD
R14	1.5 K Ohms	1/16W-5% Resistor SMD
R16,R9	1.8 K Ohms	1/16W-5% Resistor SMD
R17,R18,R19,R20	4.7 K Ohms	1/16W-5% Resistor SMD
R1,R6,R8	10 K Ohms	1/16W-5% Resistor SMD
R4	470 K Ohms	1/16W-5% Resistor SMD
R3	1 M Ohms	1/16W-5% Resistor SMD
L2,L1	10 $\mu$ H	10% Inductance 300 mA SMD
D4,D5,D6,D7,D8,D9,D10, D11,D12	LED GREEN 30mA	LPM670 LED Green Mini TOPLED 30mA SMD
D13,D1	LED RED 30mA	LSM676 LED Red Mini TOPLED 30mA SMD
D3	MRA4007	DIODE Rectifier
D2	SMBJ9.0A	DIODE TRANSIL 9V
U8	MAX202ECSE	RS-232 Transceiver
U3	DF005S	Bridge rectifier
J19	Smart Card Full Size	ITT CCM01
J22	SIM Connector	ITT CCM03
Y1	Crystal 8 Mhz	
F1	FUSE 500mA	SMD FUSE 500mA
U1	TPS2041AD	Current Limit Switches
U2	MAX603CSA	5V Adj. Regulator
U4	PLCC68 THole	Socket PLCC68 Through Hole
U5	PLCC28 CMS	Socket PLCC28 CMS



## Section 5

---

# Document Revision History

### Changes from 4302A to 4302B

1. Updated document to reflect version 1.2.0 of MCU demonstration board.



## Atmel Corporation

2325 Orchard Parkway  
San Jose, CA 95131, USA  
Tel: 1(408) 441-0311  
Fax: 1(408) 487-2600

## Regional Headquarters

### Europe

Atmel Sarl  
Route des Arsenalux 41  
Case Postale 80  
CH-1705 Fribourg  
Switzerland  
Tel: (41) 26-426-5555  
Fax: (41) 26-426-5500

### Asia

Room 1219  
Chinachem Golden Plaza  
77 Mody Road Tsimshatsui  
East Kowloon  
Hong Kong  
Tel: (852) 2721-9778  
Fax: (852) 2722-1369

### Japan

9F, Tonetsu Shinkawa Bldg.  
1-24-8 Shinkawa  
Chuo-ku, Tokyo 104-0033  
Japan  
Tel: (81) 3-3523-3551  
Fax: (81) 3-3523-7581

## Atmel Operations

### Memory

2325 Orchard Parkway  
San Jose, CA 95131, USA  
Tel: 1(408) 441-0311  
Fax: 1(408) 436-4314

### Microcontrollers

2325 Orchard Parkway  
San Jose, CA 95131, USA  
Tel: 1(408) 441-0311  
Fax: 1(408) 436-4314

La Chantrerie  
BP 70602  
44306 Nantes Cedex 3, France  
Tel: (33) 2-40-18-18-18  
Fax: (33) 2-40-18-19-60

### ASIC/ASSP/Smart Cards

Zone Industrielle  
13106 Rousset Cedex, France  
Tel: (33) 4-42-53-60-00  
Fax: (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.  
Colorado Springs, CO 80906, USA  
Tel: 1(719) 576-3300  
Fax: 1(719) 540-1759

Scottish Enterprise Technology Park  
Maxwell Building  
East Kilbride G75 0QR, Scotland  
Tel: (44) 1355-803-000  
Fax: (44) 1355-242-743

### RF/Automotive

Theresienstrasse 2  
Postfach 3535  
74025 Heilbronn, Germany  
Tel: (49) 71-31-67-0  
Fax: (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.  
Colorado Springs, CO 80906, USA  
Tel: 1(719) 576-3300  
Fax: 1(719) 540-1759

### Biometrics/Imaging/Hi-Rel MPU/ High Speed Converters/RF Datacom

Avenue de Rochepleine  
BP 123  
38521 Saint-Egreve Cedex, France  
Tel: (33) 4-76-58-30-00  
Fax: (33) 4-76-58-34-80

---

### Literature Requests

[www.atmel.com/literature](http://www.atmel.com/literature)

**Disclaimer:** The information in this document is provided in connection with Atmel products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Atmel products. **EXCEPT AS SET FORTH IN ATMEL'S TERMS AND CONDITIONS OF SALE LOCATED ON ATMEL'S WEB SITE, ATMEL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ATMEL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ATMEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.** Atmel makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Atmel does not make any commitment to update the information contained herein. Unless specifically provided otherwise, Atmel products are not suitable for, and shall not be used in, automotive applications. Atmel's products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.

© Atmel Corporation 2006. All rights reserved. Atmel®, logo and combinations thereof, and Everywhere You Are® are the trademarks or registered trademarks, of Atmel Corporation or its subsidiaries. Other terms and product names may be trademarks of others.



Printed on recycled paper.