AVR® STK501

User Guide

ATMEL®

Table of Contents

Section 1

Introduction 1-1

1.1 Features 1-2

Section 2

Using the STK501 Top Module 2-1

2.1 Connecting the STK501 to the STK500 Starter Kit 2-1

2.1.1 Placing an ATmega103(L) or ATmega128(L) on the STK500 2-1

2.2 PORT Connectors 2-2

2.2.1 PORT E/PORT F 2-2

2.2.2 PORT G/AUX 2-3

2.2.2.1 PG0 - PG4 2-3

2.2.2.2 A16 2-3

2.2.2.3 SRAMEN 2-3

2.2.2.4 PEN 2-3

2.3 Programming the ATmega103(L)/ 128(L) 2-3

2.3.1 In-System Programming 2-3

2.3.2 High-voltage Programming 2-4

2.4 JTAG Connector 2-5

2.5 External SRAM 2-6

2.5.1 A16 2-7

2.5.2 SRAMEN 2-7

2.6 Ram High Address Jumpers 2-8

2.7 A[7:0] Connector 2-8

2.8 Using the SRAM Interface with AT90S/LS8515 and ATmega161 2-8

2.9 TOSC Switch 2-9

2.10 RS-232C Port 2-9

Section 3

Troubleshooting Guide 3-1

Section 4

Technical Specifications 4-1

Section 5

Technical Support 5-1

Section 6

Complete Schematics 6-1

Section 1

Introduction

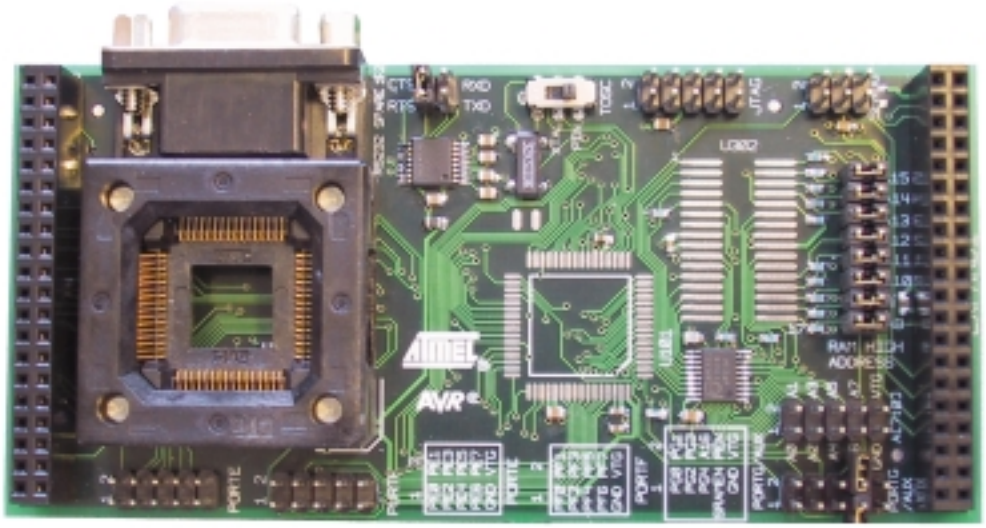
The STK501 board is a top module designed to add ATmega103(L) and ATmega128(L) support to the STK500 development board from Atmel Corporation. With this board the STK500 is extended to support all current AVR devices in a single development environment.

The STK501 includes connectors, jumpers and hardware allowing full utilization of the new features of the ATmega128(L) while the Zero Insertion Force (ZIF) socket allows easy use of TQFP packages for prototyping.

This user guide acts as a general getting started guide as well as a complete technical reference for advanced users.

In addition to adding support for new devices, it also adds new support for peripherals previously not supported by the STK500. An additional RS-232 port and external SRAM interface are09/01 among the new features. Devices with dual UART or XRAM interface can all take advantage of the new resources on the STK501 board.

Figure 1-1. STK501 Top Module for STK500



1.1 Features

- STK500 Compatible
- AVR Studio® Compatible
- Supports ATmega103(L) and ATmega128(L)
- Zero Insertion Force Socket for TQFP Packages
- TQFP Footprint for Emulator Adapters
- Supports all Added Features in ATmega128(L)
- JTAG Connector for On-chip Debugging Using JTAG ICE (ATmega128(L))
- Additional RS-232C Port with Available RTS/CTS Handshake Lines
- Adds External SRAM Support to the STK500 Board (Usable for all Devices with XRAM Interface)
- On-board 32 kHz Crystal for Easy RTC Implementations

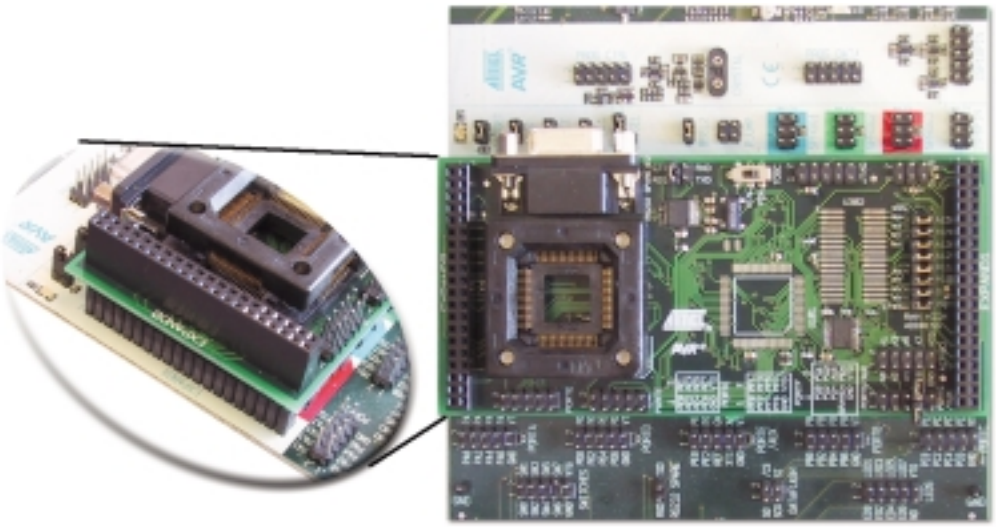
Section 2

Using the STK501 Top Module

2.1 Connecting the STK501 to the STK500 Starter Kit

The STK501 should be connected to the STK500 expansion header 0 and 1. It is important that the top module is connected in the correct orientation as shown in Figure 2-1. The EXPAND0 written on the STK501 top module should match the EXPAND0 written beside the expansion header on the STK500 board.

Figure 2-1. Connecting STK501 to the STK500 Board



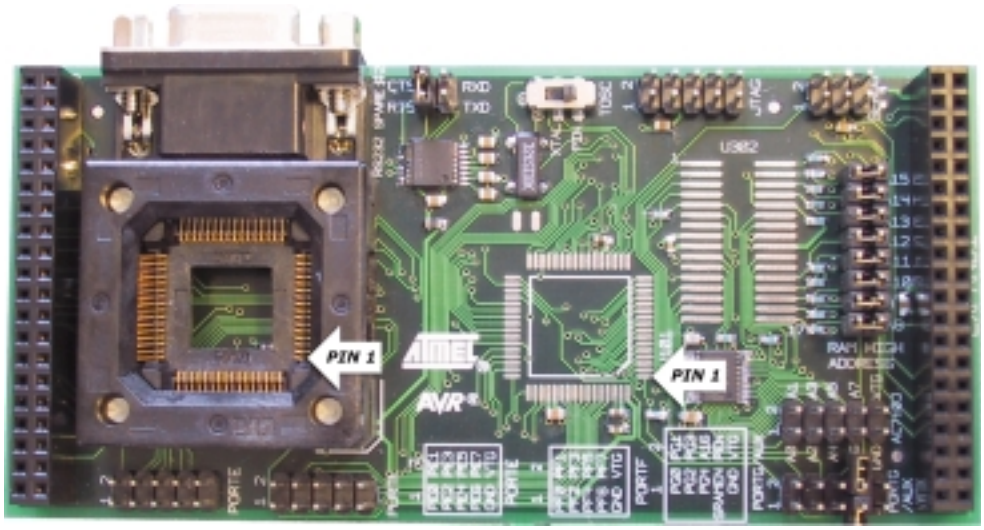
Note: Connecting the STK501 with wrong orientation may damage the board.

2.1.1 Placing an ATmega103(L) or ATmega128(L) on the STK500

The STK501 contains both a ZIF socket, and the pinout for a TQFP package; which allows an easy way of soldering an emulator adapter directly into the STK501. Care should be taken so that the device (or adapter) is mounted with the correct orientation. Figure 2-2 shows the location of pin 1 for the ZIF socket and the TQFP footprint.

Caution: Do not mount an ATmega103(L) or ATmega128(L) on the STK501 at the same time as an AVR is mounted on the STK500 board.

Figure 2-2. Pin1 on ZIF Socket and TQFP Footprint



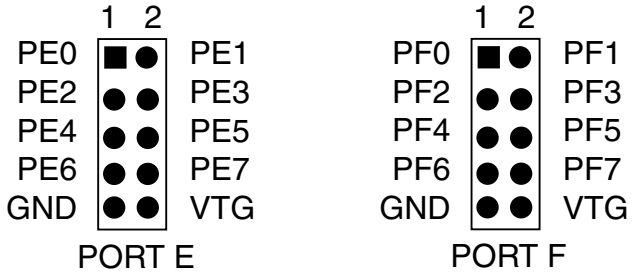
2.2 PORT Connectors

Since the ATmega103(L) and ATmega128(L) have additional ports not available on the STK500, these ports are located on the STK501 board. They have the same pinout and functionality as the ports on the STK500 board. Port A to Port D which are already present on the STK500 board are not duplicated on the STK501.

2.2.1 PORT E/PORT F

Figure 2-3 shows the pinout for the I/O port headers Port E and Port F.

Figure 2-3. General I/O Ports

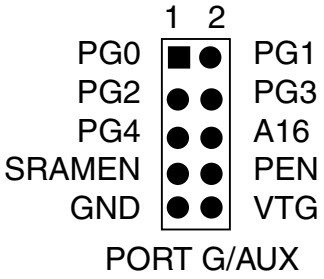


Note: Port E is also present on the STK500, but only PE0 to PE2 (3 least significant bits) are accessible there. To access all Port E bits the connector on the STK501 must be used.

2.2.2 PORT G/AUX

In addition to the normal Port G pins, this connector has some extra signals. See Figure 2-4.

Figure 2-4. PORTG/AUX



2.2.2.1 PG0 - PG4

These are general I/O ports for the ATmega128(L) and connect to the ZIF socket and the TQFP footprint. The PG3 and PG4 signals are routed through the TOSC switch since these pins also are inputs for a 32 kHz oscillator. For a description on the TOSC switch see Section 2.9.

Note: ATmega103(L) does not have Port G.

2.2.2.2 A16

This line goes to A16 (most significant address bit) on the SRAM. See Section 2.5 for more information about this signal. can be connected to any AVR pin.

2.2.2.3 SRAMEN

The SRAMEN signal controls if the SRAM is enabled or not. To enable the SRAM a LOW level should be applied to this pin. See “External SRAM”, Section 2.5, for more information on how to use this signal. This signal is pulled high by default.

2.2.2.4 PEN

The PEN pin is connected to the PEN pin on the ATmega103(L)/128(L). This pin is described in the programming section of the ATmega103(L) and ATmega128(L) datasheets.

2.3 Programming the ATmega103(L)/128(L)

The ATmega103(L) and ATmega128(L) can be programmed using both SPI and High-voltage Parallel Programming. This sub section will explain how to connect the programming cables to successfully use one of these two modes. The AVR Studio STK500 software is used in the same way as for other AVR parts.

Note: The ATmega128(L) also supports Self Programming. See AVR109 application note for more information on this topic.

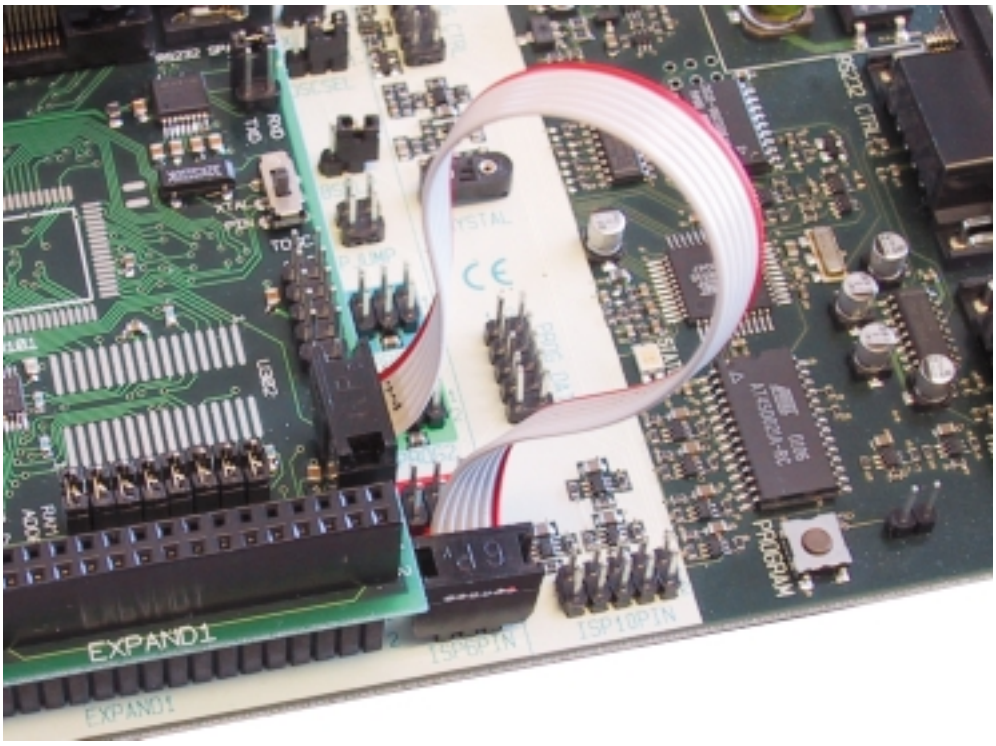
2.3.1 In-System Programming

To program the ATmega103(L) or ATmega128(L) using ISP programming mode, connect the 6-wire cable between the ISP6PIN connector on the STK500 board and the ISP connector on the STK501 board as shown in Figure 2-5.

The device can be programmed using the serial programming mode in the AVR Studio STK500 software.

Note: See the STK500 User Guide for information on how to use the STK500 front-end software for ISP programming.

Figure 2-5. In-System Programming



2.3.2 High-voltage Programming

To program the ATmega103(L) or ATmega128(L) using High-voltage (Parallel) Programming, connect the PROGCTRL to PORTD and PROGDATA to PORTB on the STK500 as shown in Figure 2-6.

As described in the STK500 User Guide, the BSFL2 jumper must be mounted when High-voltage Programming ATmega devices. This also applies to the High-voltage Programming of ATmega103(L) and ATmega128(L).

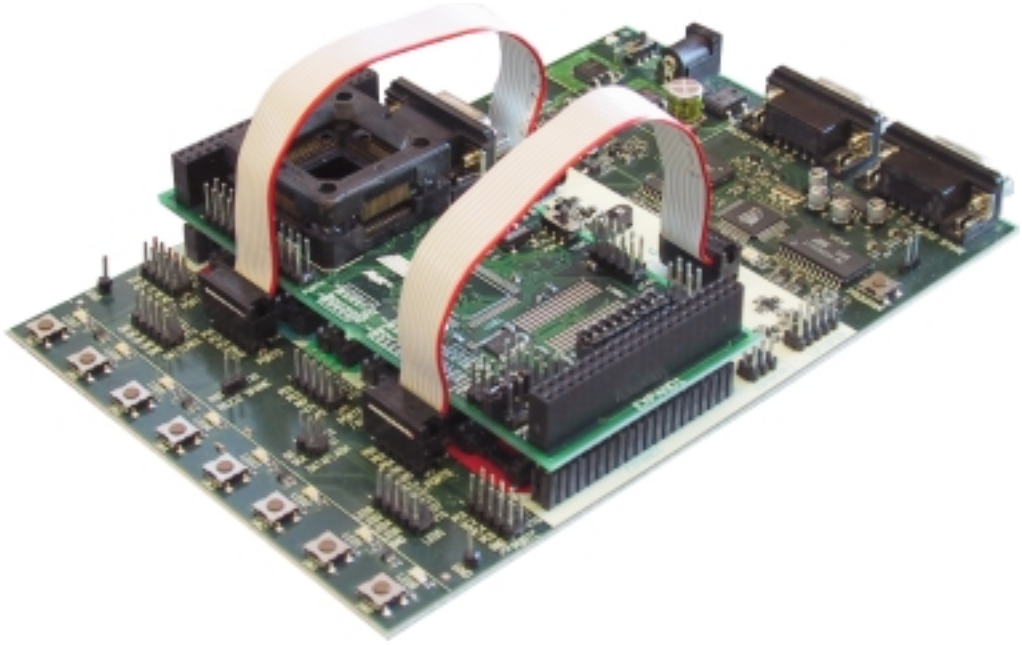
The device can now be programmed using the High-voltage Programming mode in AVR Studio STK500 software.

Note: See the STK500 User Guide for information on how to use the STK500 front-end software in High-voltage Programming mode.

Note: For the High-voltage Programming mode to function correctly, the target voltage must be higher than 4.5V.

Caution: Make sure the SRAM (if mounted) can handle this voltage.

Figure 2-6. High-voltage (Parallel) Programming



2.4 JTAG Connector

The JTAG connector is intended for the ATmega128(L) that has a built-in JTAG interface. The pinout of the JTAG connector is shown in Figure 2-7 and is compliant with the pinout of the JTAG ICE available from Atmel. Connecting a JTAG ICE to this connector allows On-chip Debugging of the ATmega128(L).

More information about the JTAG ICE and On-chip Debugging can be found in the AVR JTAG ICE User Guide, which is available at the Atmel web site, www.atmel.com.

Figure 2-7. JTAG Connector

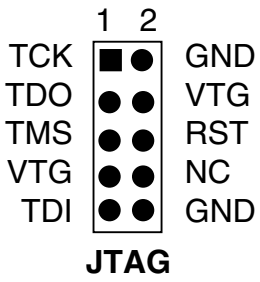
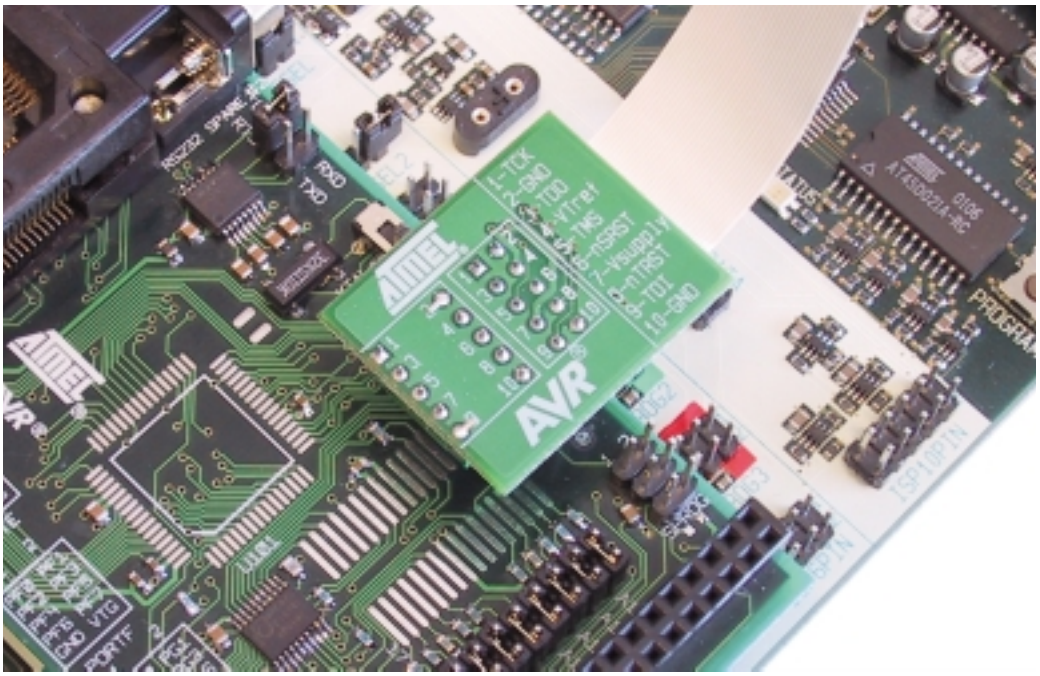


Figure 2-8 shows how to connect the JTAG ICE probe on the STK501 board.

Figure 2-8. Connecting JTAG ICE to the STK501



2.5 External SRAM

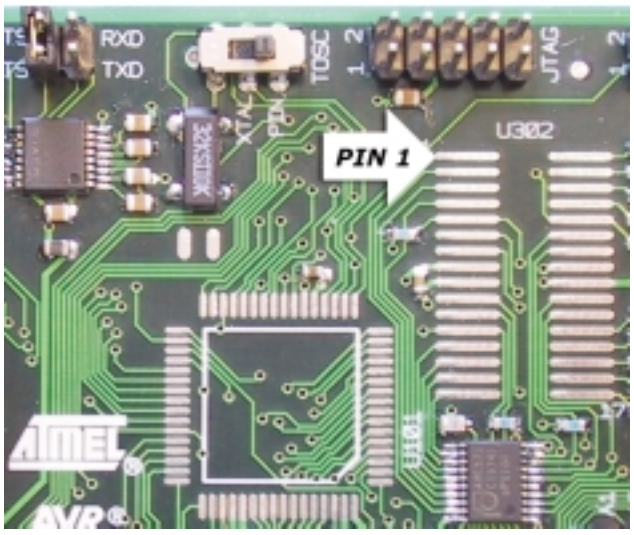
The STK501 contains a footprint where an external SRAM device can be mounted. Make sure the SRAM device has the same voltage range as the rest of the design.

Caution: Special care should be taken if a low voltage SRAM is used, since High-voltage Programming requires a programming voltage higher than 4.5V. Low-voltage SRAM may be damaged if High-voltage Programming of the target AVR is done.

Table 2-1 shows a list of recommended SRAM devices, and typical range of operation. It is important that the SRAM device is soldered with the correct orientation as shown in Figure 2-9.

Note: The SRAM is disabled by default. To enable SRAM support, put a jumper between the SRAMEN and GND pin on the PORTG/AUX connector.

Figure 2-9. Pin1 on SRAM Footprint and Pinout



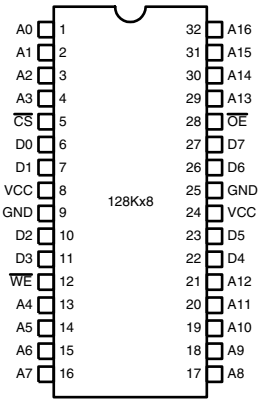


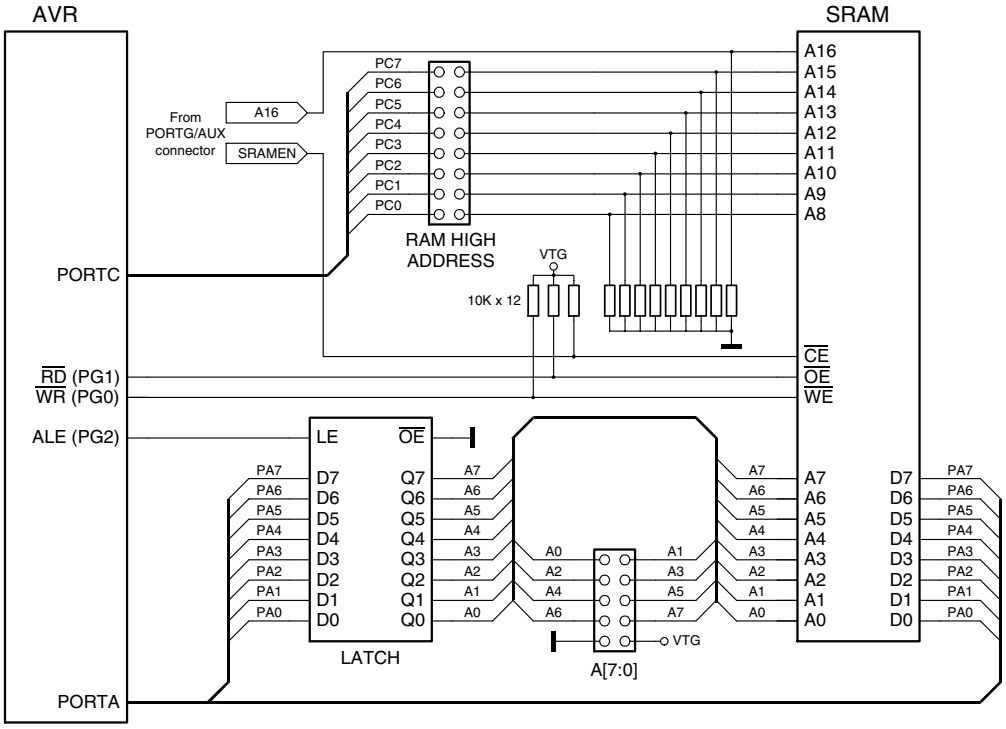
Table 2-1. Recommended SRAM Devices

Manufacturer	Part Number	Supply Voltage (V)	Package
ISSI	IS63LV1024-T	3.3	TSOP-II
ISSI	IS63LV1024-J	3.3	SOJ 300-mil
ISSI	IS63LV1024-K	3.3	SOJ 400-mil
IDT	IDT71124-Y	5.0	SOJ 400-mil
IDT	IDT71V124SA-TY	3.3	SOJ 300-mil
IDT	IDT71V124SA-Y	3.3	SOJ 400-mil
IDT	IDT71V124SA-PH	3.3	TSOP-II

2.5.1 A16

The A16 pin on the PORTG/AUX connector is connected to A16 (address pin 16) on the SRAM. ATmega103(L) and ATmega128(L) support up to 60 KB of external SRAM. The STK501 SRAM footprint is for a 128 KB SRAM. Implementing software control of the A16 line will increase the memory range from 64 KB to 128 KB. This line is pulled low by default, addressing the lower 64 KB of the SRAM.

Figure 2-10. SRAM Block Schematic

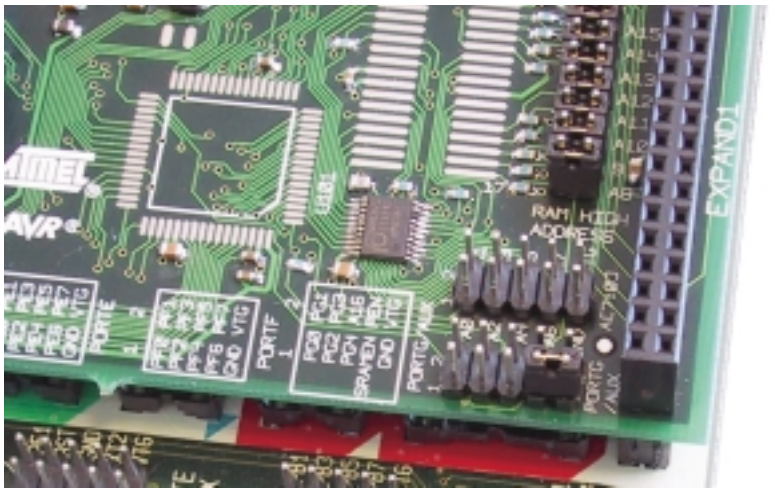


2.5.2 SRAMEN

The SRAMEN pin on the PORTG/AUX connector is connected to the Chip-enable (CE) pin of the SRAM. This signal controls if the SRAM should be enabled or not. To enable the SRAM, a low level should be applied to this pin. This pin is pulled high by default, through a 10 kΩ resistor. Figure 2-10 shows a simplified block schematic on how the SRAM interface is implemented. Figure 2-11 shows how to enable the SRAM by shorting SRAMEN and GND on the PORTG/AUX connector using one of the supplied jumpers.

This signal can also be controlled by software or by some external control logic.

Figure 2-11. SRAMEN Connected to GND



2.6 Ram High Address Jumpers

When External Memory is enabled in an AVR, all Port C pins are by default used for the high address byte. If the full 60 KB address space is not required to access the external memory, some, or all, Port C pins can be released for normal port pin function as described in the ATmega128(L) datasheet. AT90S/LS8515, ATmega103(L) and ATmega161 do not have this feature, and all jumpers should be connected if using the XRAM interface with these devices.

If some or all of the Port C pins are released for normal port pin functions, the corresponding “RAM High Address” jumper should be removed to avoid any Port C activity to reach the SRAM address pins thus corrupting the address.

If a jumper is removed, the corresponding address line will be pulled low giving a logic zero on that address bit on the SRAM. See the block schematic on Figure 2-10.

2.7 A[7:0] Connector

The connector marked A[7:0] contains the 8 least-significant bits of the external SRAM address bus. The purpose of the connector is to provide easy access to the address bus. The 8 most significant bits can be found on the “Ram High Addresses” jumpers or the Port C connector.

The connector is placed after the latch as shown in Figure 2-10.

This connector is handy when using the SRAM interface to interface external devices.

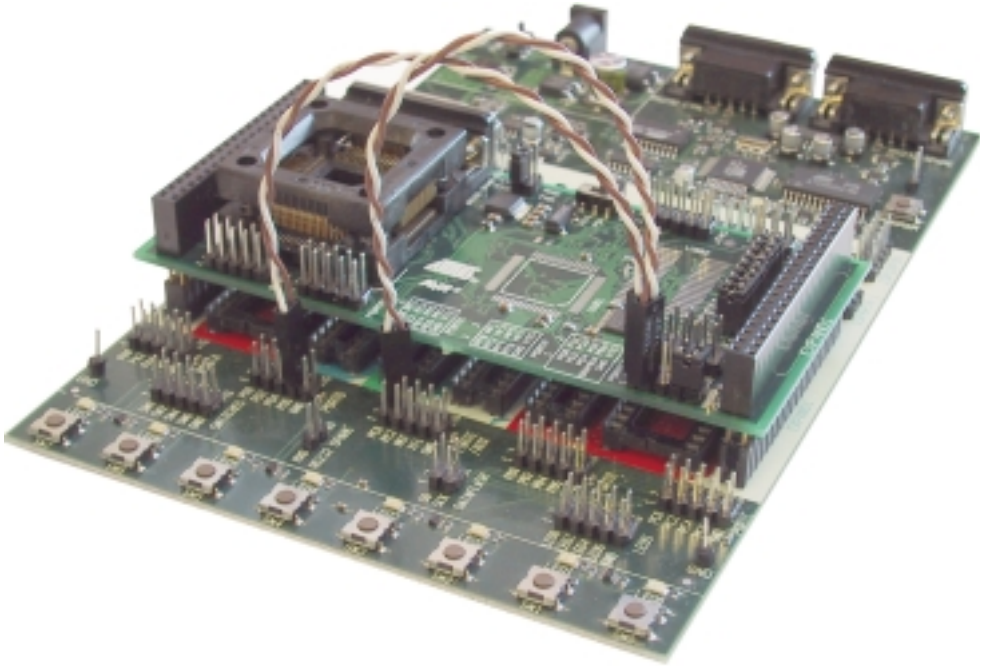
2.8 Using the SRAM Interface with AT90S/LS8515 and ATmega161

When using the SRAM interface with devices placed in the STK500 board, some additional straps are required. The reason is that the $\overline{RD}$, $\overline{WR}$, and ALE signals are not on the same port pins for the AT90S8515/ATmega161(L) and ATmega103(L)/ ATmega128(L), so these signals must be routed manually using two of the 2-wire cables.

Table 2-2. Signal Routing Required for AT90S8515A and ATmega161(L)

Connections	STK500	STK501	Description
Write Signal WR	PD6	PG0	Connect PD6:STK500 to PG0:STK501
Read Signal RD	PD7	PG1	Connect PD7:STK500 to PG1:STK501
Address Latch Enable ALE	PE1	PG2	Connect PE1:STK500 to PG2:STK501

Figure 2-12. Enabling SRAM Interface for Devices in STK500



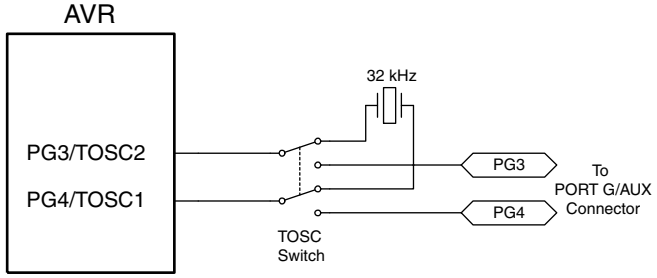
2.9 TOSC Switch

On the ATmega128(L) the TOSC1 and TOSC2 lines are shared with Port G (PG4 and PG3). The TOSC switch select if the 32 kHz crystal, or the Port G connector pins should be connected to the pins on the device.

Figure 2-13 shows a simplified block schematic on how this is implemented.

Note: Port G is not available on the ATmega103(L), the switch will thus only select if the 32 kHz crystal should be connected or not.

Figure 2-13. TOSC Block Schematic

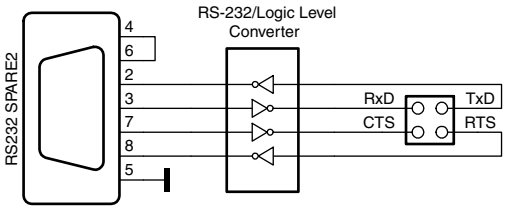


2.10 RS-232C Port

The ATmega128(L) has an additional UART compared to the ATmega103(L). The RS-232 port on the STK501 board has in addition to the RXD and TXD lines support for RTS and CTS flow control. Figure 2-14 shows a simplified block schematic on how this is implemented.

Note: The UART in ATmega128(L) does not support hardware RTS or CTS control. If such functionality is needed, it must be implemented in software.

Figure 2-14. UART Block Schematic



This UART can also be used from devices placed in the STK500 board. Simply connect the appropriate port pins to RXD and TXD on the STK501 board.

Note: If no software RTS/CTS flow control is implemented, a jumper shorting RTS and CTS will ensure correct communication with an external application that uses such flow control.

Section 3

Troubleshooting Guide

Table 3-1. Troubleshooting Guide

Problem	Reason	Solution
SRAM does not work properly.	The SRAM is not connected.	Verify all solderings, and make sure the Pin1 on the SRAM matches the one on the footprint. Make sure the SRAM pinout is correct.
	SRAMEN is not mounted.	Make sure that the SRAMEN is connected to GND on the AUX connector.
	XRAM interface is not enabled in the AVR device.	Verify that the code actually enables the XRAM interface.
	Some of the ADDRESS HIGH BYTE jumpers may be set incorrectly.	Connect some or all of ADDRESS HIGH BYTE jumpers.
SRAM does not work when used by devices on the STK500 board.	$\overline{WR}$, $\overline{RD}$ and ALE signals must be strapped using two 2-wire cables.	Use two 2-wire cables, and connect these signals to the appropriate pins.
After doing a High-voltage Programming of the AVR, the SRAM does not work properly.	The SRAM might be damaged due to the High-voltage needed to program the AVR.	Make sure the SRAM handles 5V, if High-voltage Programming mode should be used.

Section 4

Technical Specifications

System Unit

Physical Dimensions . . . . . 56 x 119 x 27 mm

Weight . . . . . 70 g

Operating Conditions

Voltage Supply . . . . . .2.7V - 5.5V

Connections

Serial Connector . . . . . .9-pin D-SUB female

Serial Communications Speed . . . . . 250 kbps

Section 5

Technical Support

For Technical support, please contact avr@atmel.com. When requesting technical support, please include the following information:

- Which target AVR device is used (complete part number)
- Target voltage and speed
- Clock source and fuse setting of the AVR
- Programming method (ISP or High-voltage)
- Hardware revisions of the AVR tools, found on the PCB
- Version number of AVR Studio. This can be found in the AVR Studio help menu.
- PC operating system and version/build
- PC processor type and speed
- A detailed description of the problem

Section 6

Complete Schematics

On the following pages the complete schematics and assembly drawing of the STK501 revision B are shown.

Figure 6-1. Schematics, 1 of 3

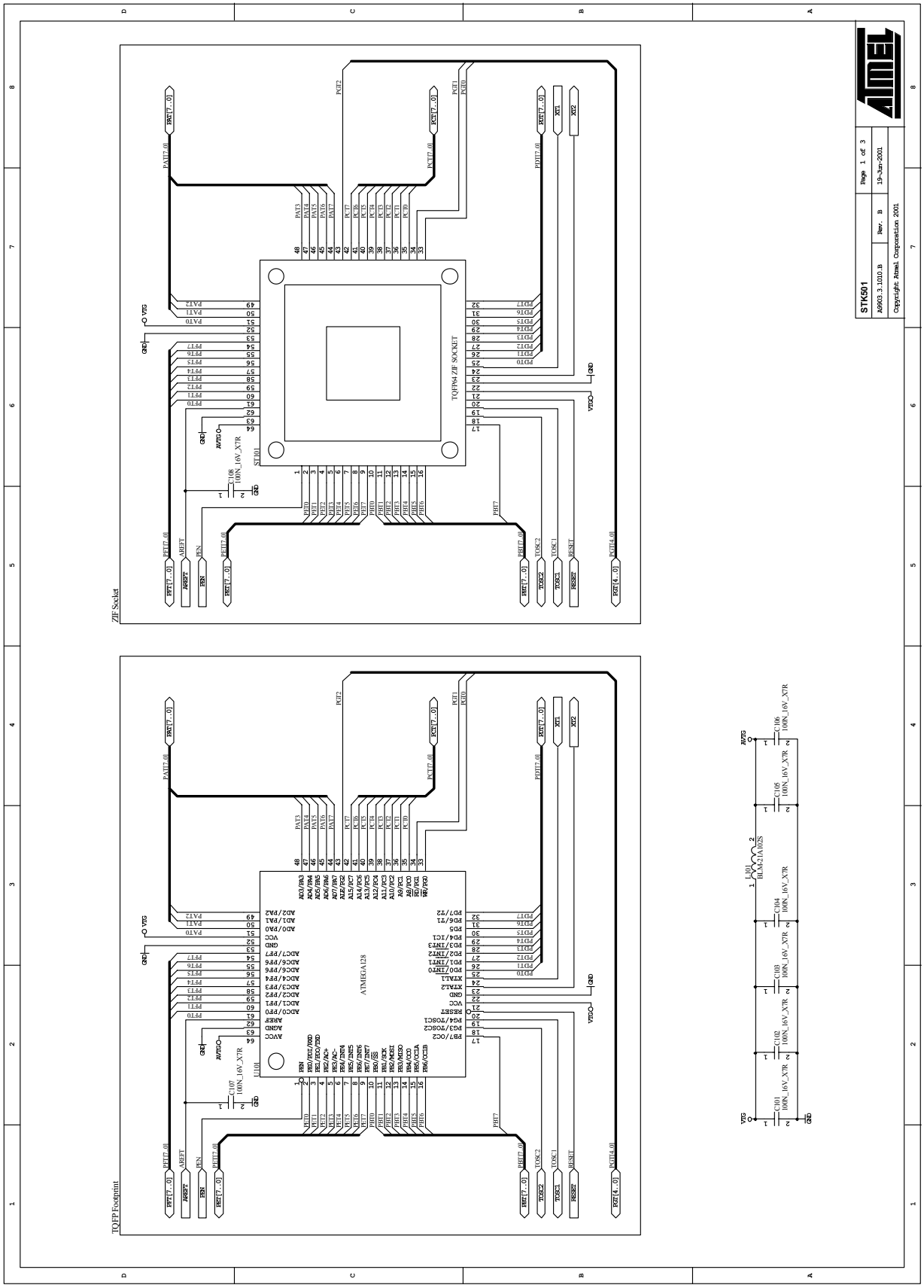


Figure 6-2. Schematics, 2 of 3

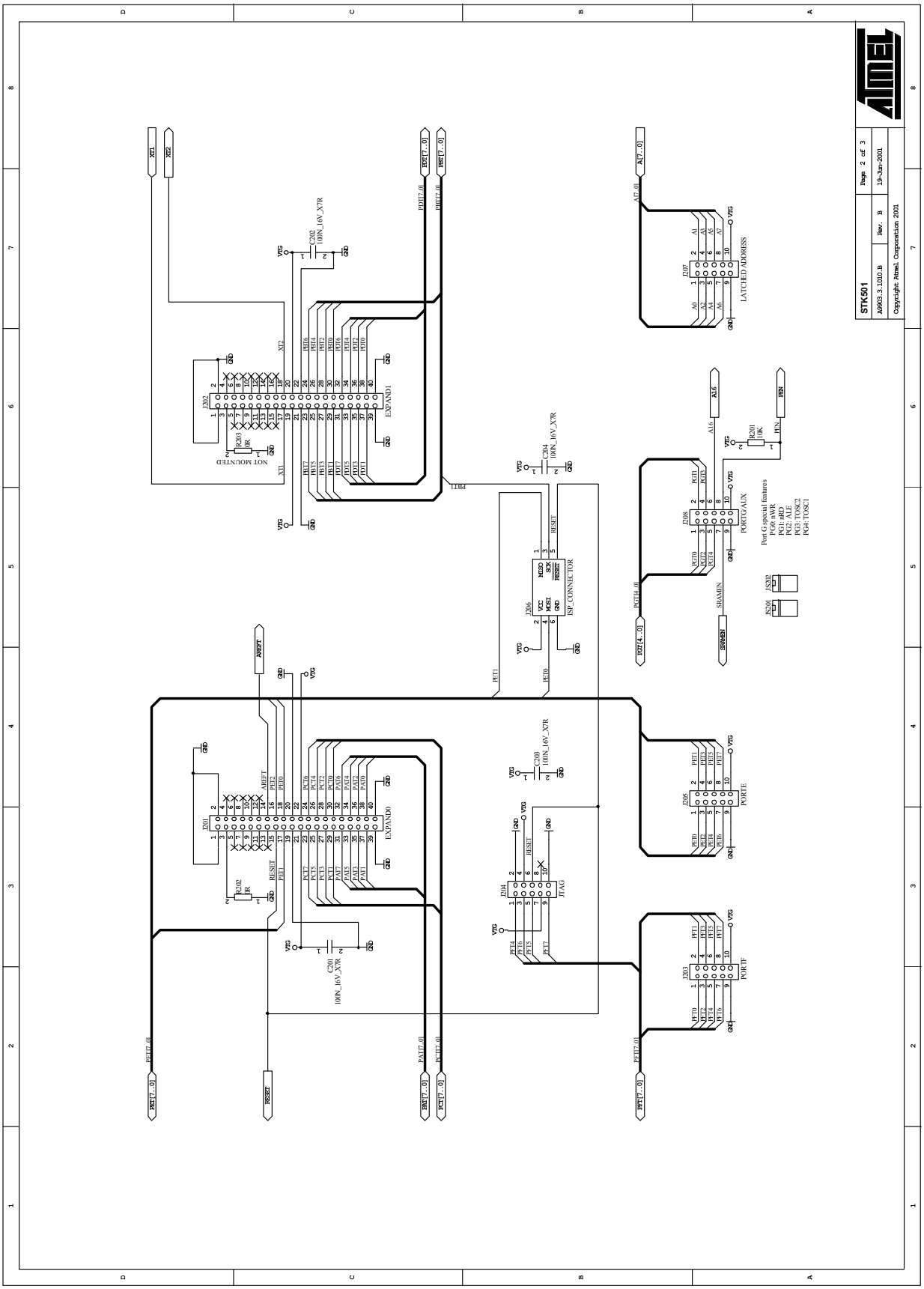


Figure 6-3. Schematics, 3 of 3

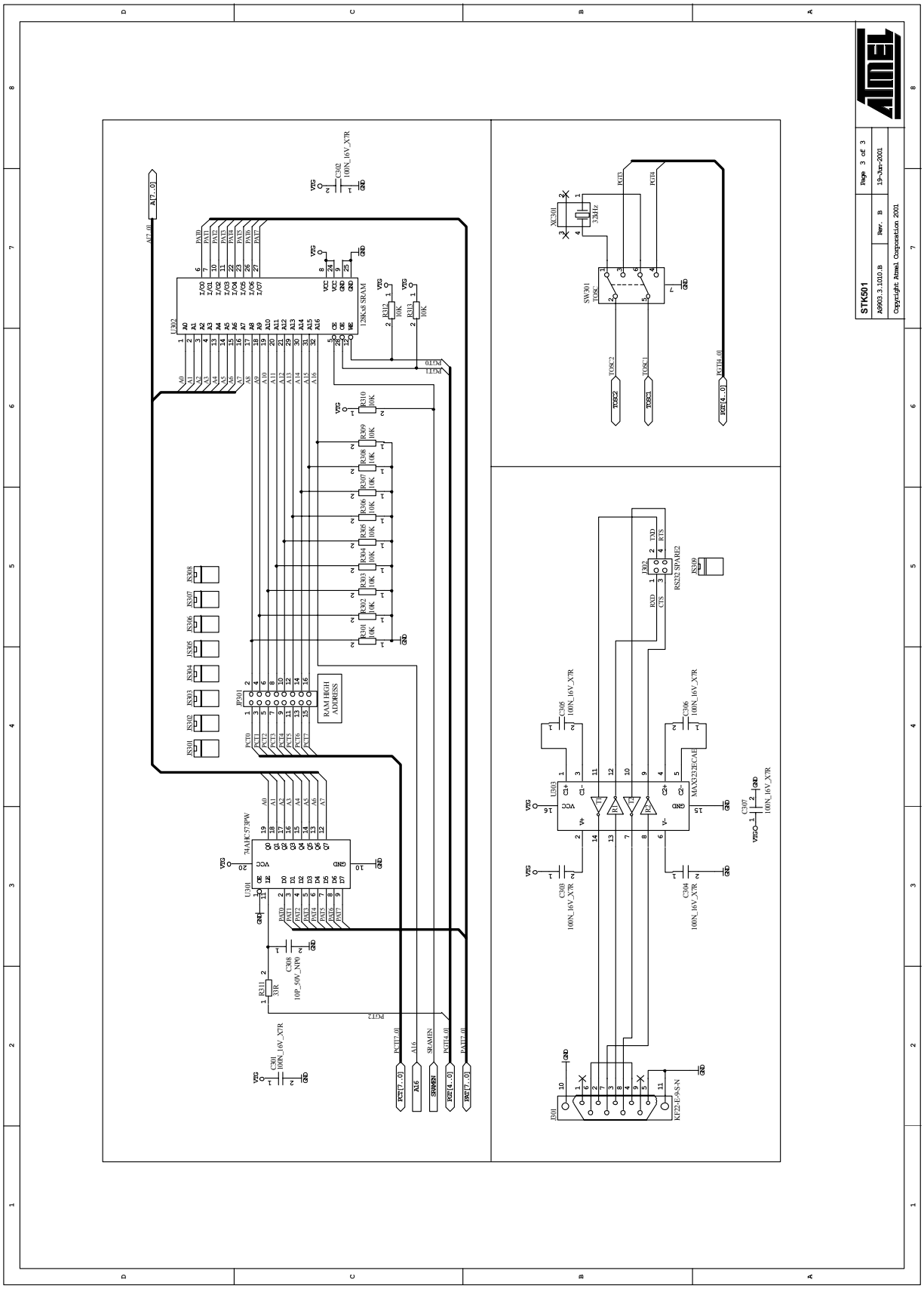
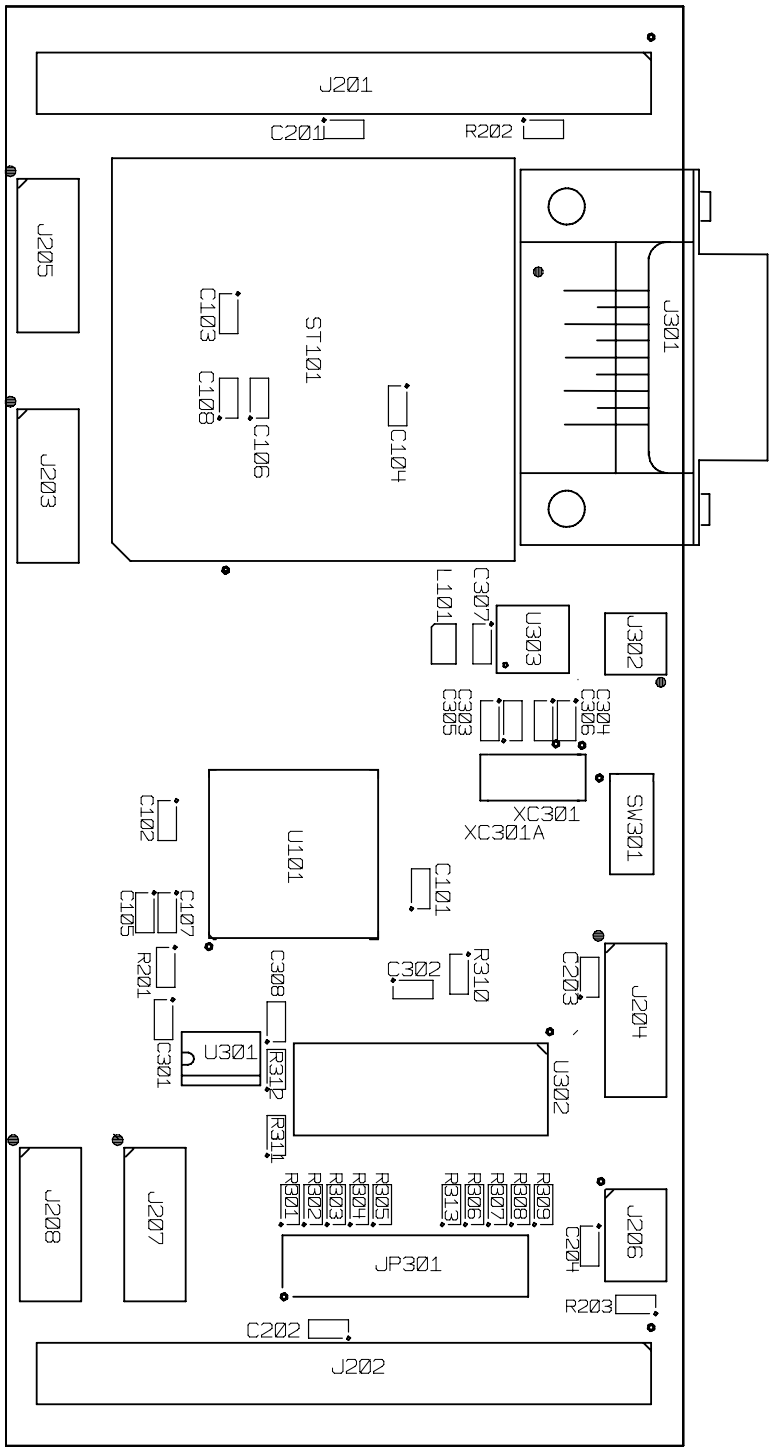


Figure 6-4. Assembly Drawing, 1 of 1



Atmel Headquarters

Corporate Headquarters
2325 Orchard Parkway
San Jose, CA 95131
TEL (408) 441-0311
FAX (408) 487-2600

Europe
Atmel SarL
Route des Arsenaux 41
Casa Postale 80
CH-1705 Fribourg
Switzerland
TEL (41) 26-426-5555
FAX (41) 26-426-5500

Asia
Atmel Asia, Ltd.
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimhatsui
East Kowloon
Hong Kong
TEL (852) 2721-9778
FAX (852) 2722-1369

Japan
Atmel Japan K.K.
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 Product Operations

Atmel Colorado Springs
1150 E. Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
TEL (719) 576-3300
FAX (719) 540-1759

Atmel Grenoble
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
TEL (33) 4-7658-3000
FAX (33) 4-7658-3480

Atmel Heilbronn
Theresienstrasse 2
POB 3535
D-74025 Heilbronn, Germany
TEL (49) 71 31 67 25 94
FAX (49) 71 31 67 24 23

Atmel Nantes
La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
TEL (33) 0 2 40 18 18 18
FAX (33) 0 2 40 18 19 60

Atmel Rousset
Zone Industrielle
13106 Rousset Cedex, France
TEL (33) 4-4253-6000
FAX (33) 4-4253-6001

Atmel Smart Card ICs
Scottish Enterprise Technology Park
East Kilbride, Scotland G75 0QR
TEL (44) 1355-357-000
FAX (44) 1355-242-743

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

BBS
1-(408) 436-4309

© Atmel Corporation 2001.
Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of Atmel are granted by the Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not authorized for use as critical components in life support devices or systems.

Atmel®, AVR® and AVR Studio® are the registered trademarks of Atmel.
Windows® and Windows NT® are registered trademarks of Microsoft Corporation.
Terms and product names may be trademarks of others.

Printed on recycled paper.

2491A–09/01/2M