



## Section 2

---

# AT89S/AT90S Series Flash Microcontroller Starter Kit

### 2.1 Introduction

This manual describes the AT89S/AT90S Series Flash MCU Starter Kit.

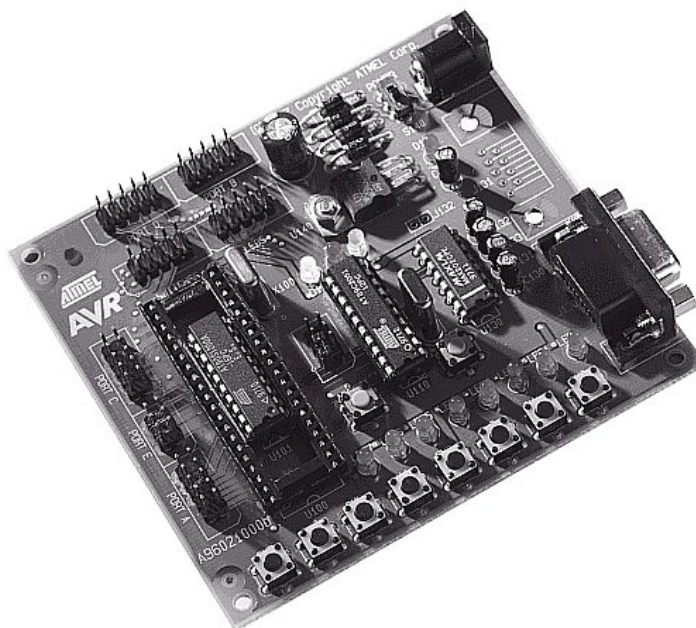
The AVR Development Board is designed to help new MCU users get quickly acquainted with the Atmel baseline microcontrollers. The AVR Development Board can also be used to “breadboard” and try new designs before a final PCB is made.

The Starter Kit has the following features:

- Regulated power supply for both DC and AC voltage sources.
- 8 push buttons for general use.
- 8 LEDs for general use.
- All AVR ports are easy accessible through header connectors.
- RS-232 level converter for general use.
- Serial programming of AT90S Series AVR MCUs.
- Serial programming of AT89S series MCUs.
- Development board for AT89C/S series MCUs.



Each part of the AVR Development Board is described in detail in the following sections. Note that when the term AVR device is used in the text it also covers both the AT90S series AVR and AT89S series MCU devices, unless specifically noted.



**Figure 2-1.** AT89S/AT90S Series Flash Microcontroller Starter Kit.

## 2.2 The Power Supply Circuit

The power supply input connector (J140) must be connected to a DC or AC voltage source. The input circuit is a full bridge rectifier so the polarity of the input voltage does not really matter, but the general standard on other AVR tools without the full bridge rectifier is to connect minus to the center pin. The supplied power cable will get minus on the center pin if the black/gray lead is connected to minus.

The power supply switch is labeled S140 and is placed close to J140. When a voltage source is connected to J140, and the power switch is turned on, the power regulator will supply the AVR Development Board with a steady 5V voltage. The red LED (D144) will light to indicate that power is on. The input voltage should be within the following range:

$$V_{DC}: 9.0 \text{ V} - 20.0 \text{ V}$$

$$V_{AC}: 6.5 \text{ V}_{RMS} - 15.0 \text{ V}_{RMS}$$

The cooling properties of the PCB against the power regulator and the power consumption of the development board set the voltage range

**2.3 The RS-232 Ports** The AVR Development Board is made with two serial ports. One port (DSUB connector J130) is used by the programming section and the other (DSUB connector J131) can be freely used. Both ports uses U130 (MAX202) for level converting. This circuit can handle baudrates up to 120 kbit/s.

Each development board is supplied with an RS-232 cable to connect it to a serial port on a PC. The cable connection is a one-to-one serial cable where pin 1 (male D-SUB) is connected to pin 1 (female D-SUB), and so on. A null modem cable (twisted RX and TX pins) will not work.

**Note:** J131 and J132 are not mounted on the supplied Development Board. If the user wants to use the available RS-232 port, these connectors need to be mounted first.

To use the available RS-232 port, just connect J132 to the AVR pins that are used for the serial port. For AVR devices with built-in UART, this is pins PD0 (RX) and PD1 (TX). For AT89C/S series controllers, the pins are P3.0 (RX) and P3.1 (TX).

**2.4 The General Push-Button Switches** The push-button switches (marked SW0 - SW7 on the PCB) are each supplied with a 10 kW pull-up resistor. When a switch is pushed down, the corresponding signal will be forced low (to GND level), otherwise the signal will be high (VDD).

The push-buttons are connected to the header connector (J150, marked SWITCHES on the PCB) as shown in the table below:

SIGNAL	J150		SIGNAL
SW0	Pin 1	Pin 2	SW1
SW2	Pin 3	Pin 4	SW3
SW4	Pin 5	Pin 6	SW5
SW6	Pin 7	Pin 8	SW7
GND	Pin 9	Pin 10	GND

**Note:** J150 is positioned right below J103 (AVR Port D) so only a short flat-cable with 10 pin header connectors is required to connect the push-buttons to AVR port D. Port D should then be configured as an input port.

**2.5 The Red LEDs for General Use** The red LEDs (Marked LED0 - LED7 on the PCB) are supplied with 680R series resistors, and can be connected directly to any AVR output port. To light a LED, the corresponding signal has to be forced low (to GND level). The LEDs are connected to header connector (J160, marked as LEDS on the PCB) as shown in the table below:

SIGNAL	J160		SIGNAL
LED0	Pin 1	Pin 2	LED1
LED2	Pin 3	Pin 4	LED3
LED4	Pin 5	Pin 6	LED5
LED6	Pin 7	Pin 8	LED7
GND	Pin 9	Pin 10	GND

**Note:** J160 is positioned right below J101 (AVR Port B) so only a short flat-cable with 10 pin header connectors is required to connect the push-buttons to AVR port B. Port B should then be configured as an output port.

## 2.6 The AVR Port Connections

U100/U101 are sockets for 40 pin and 20 pin AVR devices, respectively. Since U101 is placed inside U100, only one AVR device can be inserted at a time. All AVR ports and special pins are connected to header connectors for easy access. All header connectors are supplied with GND, to ease connections to external circuits.

The following table shows an overview of the port connections:

AVR port	Connector	AVR device	Comments
Port A	J100	40 pin only	This port is not available on 20 pin devices.
Port B	J101	20/40 pin	
Port C	J102	40 pin only	This port is not available on 20 pin devices.
Port D	J103	20/40 pin	
Port E	J104	40 pin only	ALE, ICP and OC1B pins.

Since there is some differences in naming conventions between AVR and AT89S devices, the table below shows the AT89S series port connections:

AT89S port	Connector	AT89S device	Comments
Port 0	J100	40 pin only	This port is not available on 20 pin devices.
Port 1	J101	20/40 pin	
Port 2	J102	40 pin only	This port is not available on 20 pin devices.
Port 3	J103	20/40 pin	
Port 4	J104	40 pin only	ALE/PROG, EA/VPP and PSEN pins.

Each header connector is defined as follows:

SIGNAL	J10x		SIGNAL
Pn0	Pin 1	Pin 2	Pn1
Pn2	Pin 3	Pin 4	Pn3
Pn4	Pin 5	Pin 6	Pn5
Pn6	Pin 7	Pin 8	Pn7
GND	Pin 9	Pin 10	GND

Where n is one of the ports A,B,C, or D (ports 0,1,2, or 3 for AT89S devices) and x is one of the header connectors J100, J101, J102, or J103.

The special pins on AVR 40 pin devices are connected to J104 as shown in the table below:

SIGNAL	J104		SIGNAL
OC1B	Pin 1	Pin 2	ALE
ICP	Pin 3	Pin 4	GND
GND	Pin 5	Pin 6	GND

Since there is some differences in naming conventions between AVR and AT89S devices, the table below shows the AT89S series special pins connections:

SIGNAL	J104		SIGNAL
$\overline{\text{PSEN}}$	Pin 1	Pin 2	ALE/ $\overline{\text{PROG}}$
$\overline{\text{EA}}/\text{VPP}$	Pin 3	Pin 4	GND
GND	Pin 5	Pin 6	GND

**Note:** Port C, port D, and port E is placed on a 2.54 mm grid (100 mil) so that a single 34 pin flat cable header connector will connect all pins to an external application with a single flat cable. This make it simple to interface to external memories since all external memory pins are available on these ports.

The same can be done with port B and D. A 26 pin header connector is required to cover both ports.

## 2.7 The AVR Development Board Programming Section

The development board can be used together with the supplied utility programs to serially program AVR series devices and AT89S series devices. There are two sets of programs for supplied for programming devices, one for Windows95 and Windows NT users, and MS-DOS command line programs for MS-DOS and Win 3.1x users.

Both AT90S1200 and AT90S1200A may be programmed. For AT90S1200 devices, the external crystal should be at least 1.0 MHz.

There are three programming status LEDs on the AT90DEVBOARD. These are numbered D110 (Yellow, marked Y on the PCB), D111 (Green, marked R on the PCB), and D112 (Red, marked G on the PCB). Note that there is an error in the silkscreen marking. The red LED is marked G when it should be R, and the green LED is marked R, when it should be G. These LEDs will show the programming status according to the following description:

- The yellow LED will be lit when programming/verification is in progress.
- The green LED will be lit when verification is OK.
- The red LED will be lit when verification fails.

When the AT90DEVBOARD is turned on, all status LEDs will be turned on, and turned off one at a time. This is an indication that the board is OK. The same will happen if the programming MCU (U110) is reset.

Note that the programming MCU (U110) will take control over the reset line of the AVR device when programming/verification is performed. As soon as programming/verification is done, the reset line will be released and the AVR device will start to run its program.

**Note:** When programming AT89S8252, the port connectors should not be connected to the LEDs or the switches. This may cause programming or verification failure.

### 2.7.1 Windows95/Windows NT

In order to program an AVR device under Windows95 or Windows NT, the program **Avr-Prog** should be used. For a detailed description of the programming SW, see the **Avr-Prog** User Guide.

2.7.2 MS-DOS/  
Windows 3.1x

If Windows95 or Windows NT is not available, MS-DOS programs can be used to program the AVR device. To program and verify the Flash, the program *progf.exe* must be used. To program and verify the EEPROM the program *proge.exe* must be used. There are also two programs to read the Flash and EEPROM contents, *readf.exe* and *reade.exe* respectively. In addition there are also two programs to set the fuse bits and the lock bits. These programs are called *progfuse.exe* and *proglock.exe* respectively. For a detailed description of these programs, see the **AvrProg** User Guide.

2.8 The AVR  
Development  
Board reset  
systems

In order to make the development board work with both AVR devices and AT89S series microcontrollers a flexible reset system has to be used. The AT89S devices uses active high reset polarity, while the AVR devices uses active low reset polarity. The common reset switch (S100) is not connected directly to U100/U101 sockets. Instead it is connected to the external interrupt pin (pin 6) on the programming MCU (U110). Pin 7 on U110 is connected the reset pins on U100/U101. U110 will sense the level pin 17 at power-on to determine which reset polarity to use. The reset polarity is set according to following definition:

If pin 17 is high (VDD) at power-on, the reset polarity is active low for AVR devices.

If pin 17 is low (GND) at power-on, the reset polarity is active high for AT89S devices.

The voltage level on pin 17 can be set by using the supplied jumper on header connector J110. If the jumper is set between pins 2 and 4 (or is omitted), the level on pin 17 will be high. If the jumper is set between pins 4 and 6, the level on pin 17 will be low.

Note that the level on pin 17 has to be set prior to power-on, in order to select the right reset polarity for the devices. If the level on pin 17 is changed when power is on, this will NOT affect the reset polarity of the board.

For advanced users, there is also a possibility to change the programming MCU (U110) to either AVR or AT89S devices. U110 is currently mounted with AT89C2051. To change this to an AT90S1200 or AT90S2313, the reset polarity has to be changed. This can be done by moving R115 (0R) to R114 and R116 (0R) to R117. This will reverse the reset polarity on U110. R118 and R119 (10K) pull-up resistors can also be removed since they are only required by AT89C2051.

U110 (mounted with AVR device) is also prepared for In-System-Programming (ISP) through J110 mainly for ease of production. The AVR ISP programmer that will be available soon can be plugged direct into J110 to program U110. The pin description for J110 is shown in the table below:

SIGNAL	J104		SIGNAL
DATAOUT	Pin 1	Pin 2	VDD
CLKIN	Pin 3	Pin 4	DATAIN
RESET	Pin 5	Pin 6	GND

**Note:** All signal directions refer to U110.

2.9 The clock  
systems

U110 is currently mounted with an 11.0592 MHz crystal to fit the AT89C2051 UART. This may be changed when an AVR device is used later.

The crystal for U100/U101 (X100) is set to 4.00MHz. To change this, it need to be soldered out and replaced with another value. Note that the mounted load capacitors are selected to fit parallel resonant crystals. It may be necessary to change the load capacitors to fit any replaced crystals. Check the datasheet for recommended values on the crystal you are using. This note is also valid for the crystal to U110 (X110).

- 
- 2.10 Running 40 pin AT89S devices** The 40 pin AT89S series can use external program memory. In order to do this the pin must be set low. The pin in on U100 is connected to J104, pin 1 and is floating. In order to run reliable on internal program memory, the pin should be set high. The easiest way to do this is to connect it to pin 2 on J110, which is connected to VDD. This should not be any problem since the jumper (reset polarity setting) on J110 should be connected between pins 4 and 6 for AT89S devices.
- 
- 2.11 Troubleshooting** A lot of effort has been used to make this development board as reliable as possible, but there are some situations where it may fail. The following list is a simple checklist to go through when things do not seem to work the way they should do.
- If the development board does not work at all, check this:
- Is a power supply applied to the target board?
  - Is the power switch turned on?
- Note:** Check if the power LED on the target is lit.
- Is the reset polarity jumper set correct?
  - Was the reset polarity jumper set correct at power-on?
  - Is the  $\overline{EA}$  pin set correct on AT89S 40 pin devices.
  - Is a device inserted into U100 or U110 socket?
  - Is the device inserted the right way in the socket?
  - Are all the device pins inside the socket?
  - Is the correct device inserted to the socket?
- Check that the device is thoroughly inserted into the target board programming socket.
- 2.11.1 If the development can not be found by the AvrProg SW, check this:**
- Is the serial cable well connected in both ends?
  - Is the serial cable of the right type?
- Note:** Only a one-to-one cable will work. Null modem cables with twisted RX/TX pins will NOT work.
- Does more then one program accessing the serial port in a Windows environment?
- A common problem with Windows environments is that two programs can not control the same serial port at the same time. If a DOS window is open and a program which used the serial port has been run (even if the program has been exit), the Windows version of **AvrProg** will not work. The same situation will occur if the Windows version of **AvrProg** is still open, and the MS-DOS version is run in a DOS window. The solution is to close down (completely) all programs (and DOS windows) that may use or may have used the serial port. This should solve the problem.
- 2.11.2 Programming/ verify failures** If **AvrProg** show either of the “Programming failed!” or “Verified failed!” messages, check the following list:
- Have you selected the correct device in the device menu on the Windows version?
  - Have you remembered to use the -d option on the MS-DOS version?
  - Have you typed the device name right (-d option) on the MS-DOS version?
- Note:** If you have tried to program a device with a wrong device selection, the target board SW may crash and get totally locked. It may be seen on the AT90DEVBOARD that the yellow LED is constantly lit. On the AT90PROG1200 board, the green LED may be constantly lit. If this happens, use the respective reset buttons on the target boards, and try again. If this does not help, turn off the power to the target boards and close down the **AvrProg** programs. If the MS-DOS version is used in a DOS window in a Windows environment, close

down the DOS window. Then turn no the power to target board again and restart **AvrProg**. This should resolve the situation.

Note that parts may get damaged in your lab due to a bad ESD environment or bad handling.

NB! Always handle devices with proper ESD precautions.

If you suspect a faulty part, try several devices to check if they behave in the same manner. If they don't you may have faulty part. If they do, the target board may be damaged. The target boards should be handled with proper ESD precautions as well. Don't exceed the input power supply range of the target boards. This may cause permanent damage to the boards. Also be very careful to avoid short circuits on the target boards. This may cause permanent damage to the boards as well.

### **2.11.3 Where to get help**

If you have any problems with the Development Board or **AvrProg**, or any of the *AVR* tools, please contact your local ATMEL distributor for help. You may also send an E-mail to the *AVR* support service at [avr@atmel.com](mailto:avr@atmel.com). Include all information about the problem, all HW/SW versions, and which *AVR* tools where used. Also include information on the PC system you are using, and which operating system you are using (MS-DOS, Windows95/WindowsNT).

---

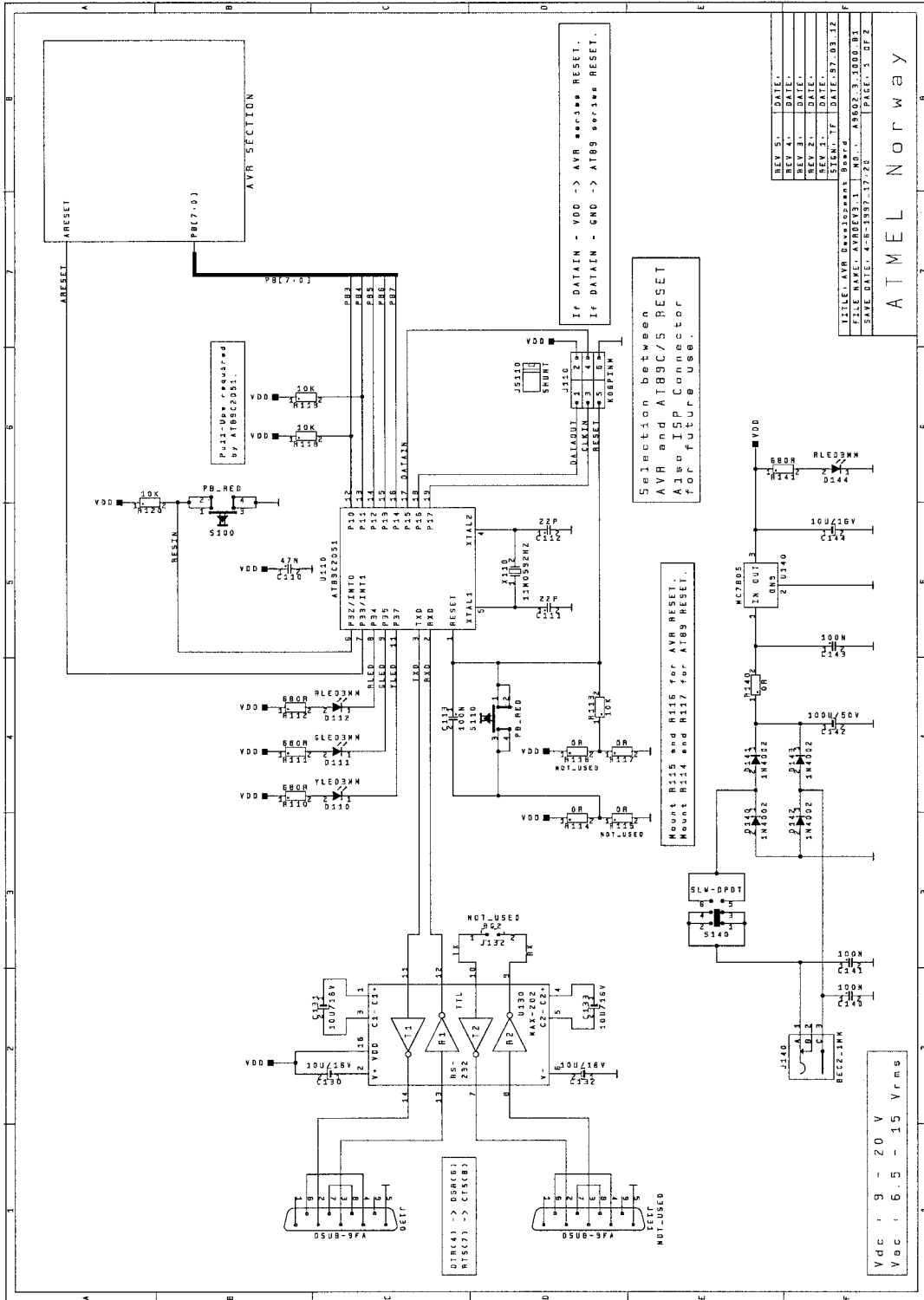
## **2.12 The AVR Development Kit packing list**

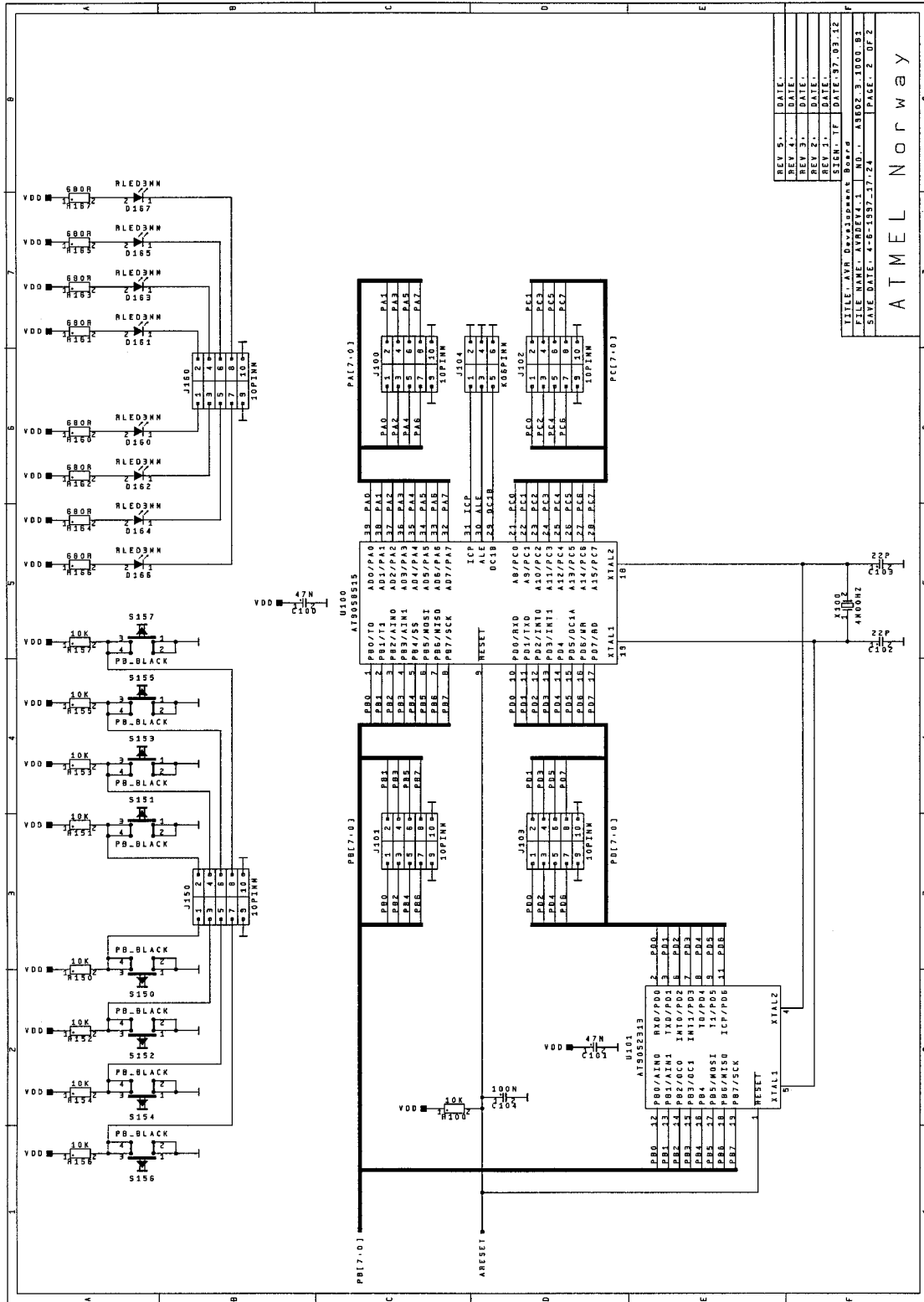
The *AVR* Development kit should be supplied with the following items:

- The *AVR* Development Board.
- Two flat-cables with 10 pin female header connectors.
- One power cable.
- One RS-232 cable.
- The ATMEL CD-ROM data books.
- The **AvrProg** SW (one diskette).
- The *AVR* Studio SW (two diskettes).
- Development Tools User Guide (this document).



2.13 Appendix: AVR Development Board schematic diagram





REV 5	DATE
REV 4	DATE
REV 3	DATE
REV 2	DATE
REV 1	DATE
SIGN. TF	DATE: 97.03.12
TITLE: AVR Development Board	
DRAWN: ...	
DATE: 4-6-1997-37-24	
PAGE: 2 OF 2	

ATMEL Norway

