

# How to Program 062 Oscilloscope

Model: 06201P, 06201KP, 06202KP

The firmware of the scope resides in its core controller, a ATmega64. This is an AVR series micro-controller from Atmel. There are two ways to program the chip for the 06201 scope. One method is using a hardware programmer. The other method is via the pre-loaded bootloader. This design note explains both.

## Program the Scope with a Hardware Programmer

To do this you need to prepare the following tools:

- 1) An AVR hardware programmer
- 2) Software to control programmer

The AVR is one of the most popular micro-controller series. It shouldn't be too difficult to find a ready-made programmer for it. There are also plenty of resources on the web for building one yourself if you don't want to purchase one. Normally, they also include information on the controlling software. Also, you can use FPGA/CPLD programmers from some big names (such as Altera or Xilinx) as an AVR programmer. You can also use the AVR STK200 parallel programmer (or compatible). An advantage of the STK200 programmer is that it works with PonyProg, a famous free programming software ([www.lancos.com](http://www.lancos.com)). This article will use the Altera programmer ByteBlasterMV as an example since the programming header on the 06201 board was designed to be compatible with it. For any other programmer you just need to make a simple dongle to route signals to their correct pins. (JYE Tech provides programmers that are compatible with the AVR STK200 and the Altera ByteBlasterMV. Please see <http://www.jyetechnology.com/en/default.html>.)

The controlling software for Altera's ByteBlasterMV was designed by Chan. It can be downloaded from <http://elm-chan.org/works/avr/reports.html>. This program runs in DOS and detailed instructions for usage can be found within the download.

Here are the programming steps for use with the ByteBlasterMV:

- 1) Identify the programming header (J4) and its orientation (Fig. 1 and Fig 2).

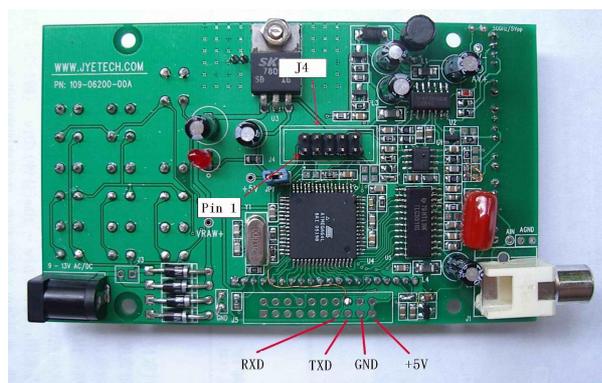


Fig. 1

### J4 pin out and signal names

|      |   |    |     |
|------|---|----|-----|
| SCK  | 1 | 2  | GND |
| MISO | 3 | 4  | VCC |
| nRST | 5 | 6  |     |
|      | 7 | 8  |     |
| MOSI | 9 | 10 | GND |

Fig. 2

- 2) Correctly connect J4 to programmer with programming cable.
- 3) Connect the programmer to your PC parallel port.
- 4) Power up the scope.
- 5) Check if the programmer and connections work properly by typing in the command "avrsp -pl1 -rf". This command reads the fuse information of the ATmega64. If it returns valid results then go to the next step (see below for proper values).
- 6) Run command

```
avrsp -pl1 113-06201-050.hex [113-06202-010.hex]
to burn in the main firmware (and bootloader).
```

- 7) Run command
  - avrsp -pl1 113-06201-050.eep**  
to burn in the EEPROM data (You don't need to do this for firmware version -092 or higher).
- 8) Program fuse bytes in the following order (This is only necessary when the chip is programmed for the first time or the fuses were changed).
  - avrsp -pl1 -fx11111111** (Fuse for extended byte, 0xFF)
  - avrsp -pl1 -fh11000000** (Fuse for high byte, 0xC0)
  - avrsp -pl1 -fl10101110** (Fuse low byte, 0xAE)
- 9) If no error was reported then programming is done. Power down scope and remove programming cable.

### Update the Scope Firmware via Bootloader

The scope was preloaded with a bootloader. Each time the scope is powered up it enters the bootloader first and stays there for about 2 – 3 seconds. During this period the bootloader will check the serial port for a communications request. If a request is found and communication established it will accept the firmware download and flash the newly downloaded firmware into the ATmega64. If nothing happens, it will boot into scope main firmware and start normal operation.

To use the bootloader the following tools are needed:

- 1) Serial level converter.
- 2) Download program running on PC.

A level converter is needed because the scope serial port signal levels (0 - +5V) are not compatible with standard serial port signal levels (+/- 3V - +/- 25V). A simple level converter can be easily made with a few common components. The schematic is available at the JYE Tech web site ([http://www.jyetech.com/Products/FunctionModules/SerialAdapter/105-04000-00B\\_S.pdf](http://www.jyetech.com/Products/FunctionModules/SerialAdapter/105-04000-00B_S.pdf)). The finished product can also be purchased from the same web site (PN: 100-04000-00B).

Alternatively many USB-Uart converters can be used for the same purpose. JYE Tech has two USB-Uart converters available (PN 07301 and PN 08101). Please visit [www.jyetech.com](http://www.jyetech.com) for details.

The download program for the PC (avrubd.exe) was designed by [Shao Ziyang](http://blog.ednchina.com/shaoziyang/) (<http://blog.ednchina.com/shaoziyang/>). This program communicates with the bootloader and transfers code to be burned to the bootloader. It can be downloaded from the JYE Tech web site (<http://www.jyetech.com/Products/LcdScope/avrubd.rar>).

Steps to use the bootloader:

- 1) Connect serial level converter to the serial port pins (shown in Fig. 1) of the scope board. Do not power up the scope yet.
- 2) Connect the other end of the level converter to PC (Fig. 3).

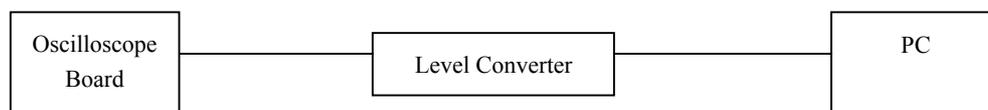


Fig. 3

- 3) Run AVRUBD.EXE. Click “Language” and select “English”.
- 4) Click on “Options” at the main menu. Change settings on the tab “AVR” to those shown in Fig. 4. Connect Key has to be “HEX 64”. Flash size should be 64K.
- 5) Change settings on tab “System” to those shown in Fig. 5. Communications protocol should be set to “Lite Xmodem”.
- 6) Change settings on tab “Com Port” to those shown in Fig. 6. Com port number should

be chosen according to the actual port used. Communication format is 9600 bps, 8N1 with no parity and flow control.

- 7) Load HEX file to be downloaded to scope. Click “File – Load” and select your file.

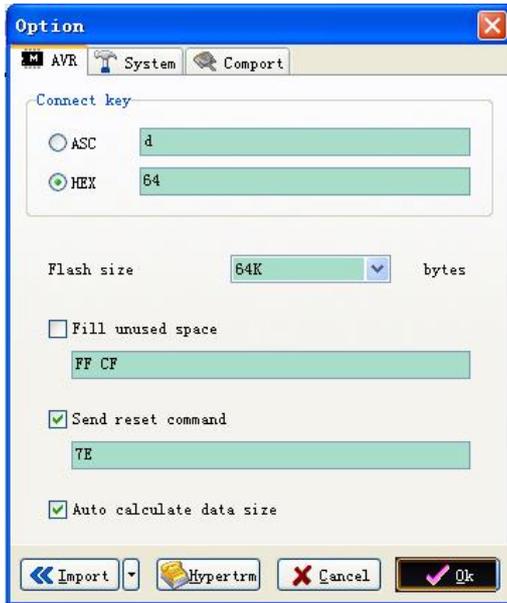


Fig. 4

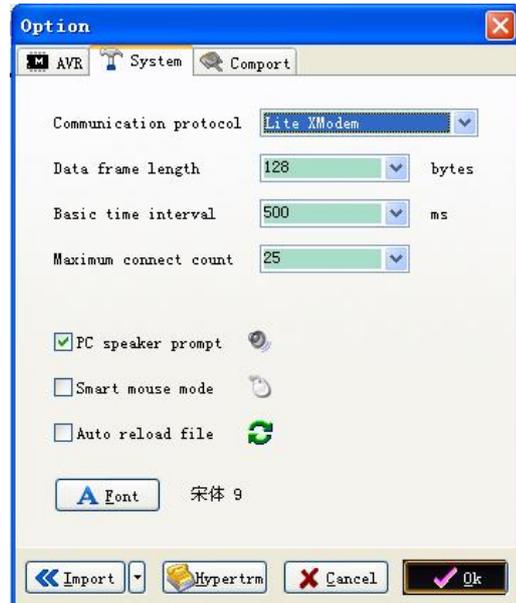


Fig. 5

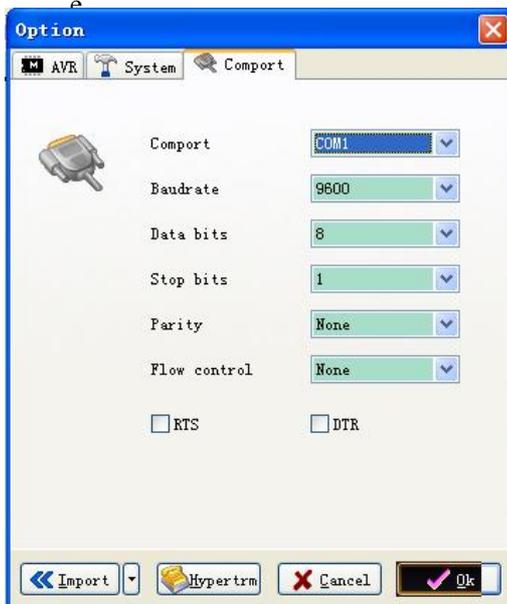


Fig. 6

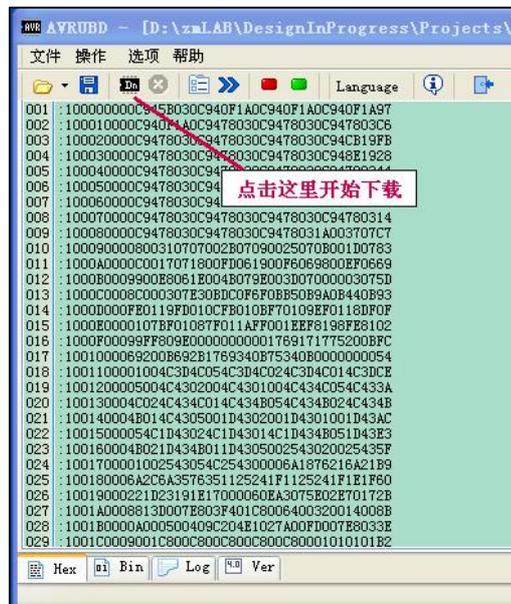


Fig. 7

ck on “Operate – Download” from the main menu) Power up the scope immediately after clicking. You will see AVRUBD is trying to communicate with the scope board. If the handshake failed you will see a screen like that of Fig. 8.

- 9) Otherwise, download is successful if you see a screen like Fig. 9.

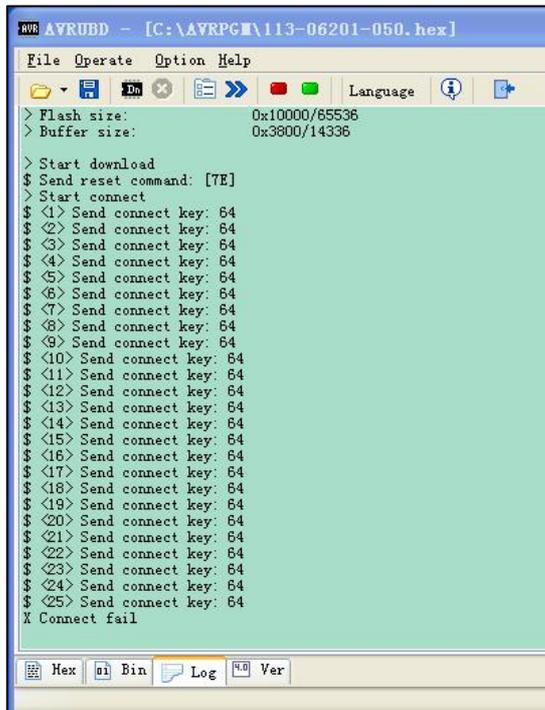


Fig. 8



Fig. 9

**Notes**

Fuse configuration of the ATmega64 is as follows. Usually these settings should not be changed. In case these bytes are modified, a programmer is needed to restore them. They can not be modified with the bootloader.

- Fuse Extended Byte:** 0bxxxxxx11 (0xFF)
- Fuse Higher Byte:** 0b11x00000 (0xC0)
- Fuse Lower Byte:** 0b10101110 (0xAE)

**Revision History**

| Version | Date       | Descriptions                                                           |
|---------|------------|------------------------------------------------------------------------|
| 01      | 2008.09.28 | First created                                                          |
| 02      | 2010.03.27 | Corrected language errors by Mr. Brent Burns. Some minor changes made. |
|         |            |                                                                        |
|         |            |                                                                        |
|         |            |                                                                        |
|         |            |                                                                        |
|         |            |                                                                        |
|         |            |                                                                        |