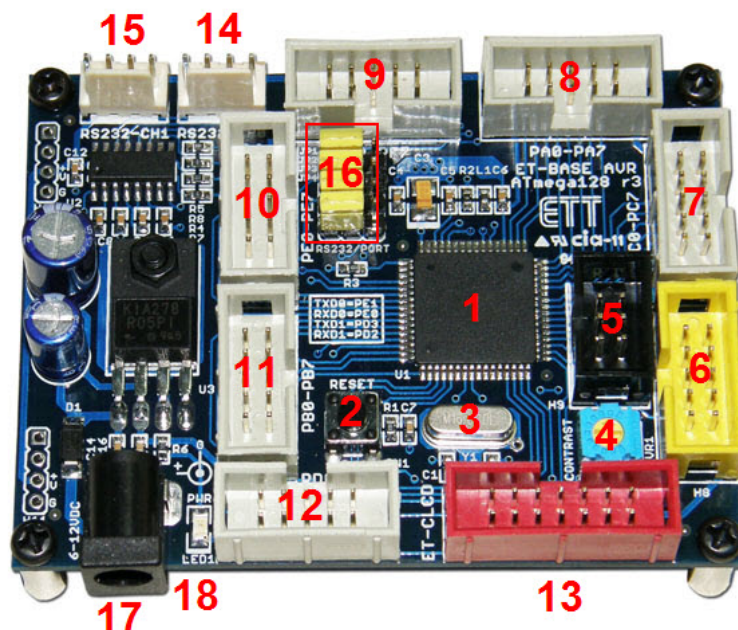


## ET-BASE AVR ATmega64/128 r3

ET-BASE AVR ATmega64/128 r3 is AVR Board Microcontroller from ATMEL that uses 64Pin MCU No.ATmega64 and No.ATmega128. It mainly emphasizes on using resources of MCU on this Board ET-BASE AVR ATmega64/128 r3. I/O Pins are externally connected to be Port PA, PB, PC, PD, PE, PF and Port ET-CLCD; so, it is more convenient to connect, and including Port for downloading program. Moreover, it adds Circuit Line Driver RS-232 to increase the capability of RS-232 Serial Port Communication easier and more convenient.

### Features of Board

- Use 8Bit AVR MCU No.ATmega64, ATmega128 from ATMEL and use Signal Clock Generator as XTAL 16MHz. Moreover, there are other outstanding features of MCU as follows;
  - Has 64Kbyte Flash Memory for ATmega64, 128Kbyte Flash Memory for ATmega128 and 4Kbytes RAM
  - Has 2Kbyte EEPROM for ATmega64 and 4Kbyte EEPROM for ATmega128; can be re-written more than 100,000 times
  - Has a maximum of 53 I/O Pins
  - Has 1-Channel SPI, 1-Channel I2C, 2-Channel Programmable Serial USARTs
  - Has 8-Channel 10-Bit ADC
  - Has 2-Channel 8-Bit Timers/Counters, 2-Channel 16-Bit Timers/Counters, 2-Channel 8-Bit PWM, Watchdog Timer, Real Time Counter
- Has 6 of 10PIN I/O PORTs: PA,PB,PC,PD,PE,PF
- Has Port ISP LOAD for programming MCU (has to use with ET-AVR ISP of other ISP Programmers that has the same pin arrangement)
- Has Circuit Line Driver 2-Channel for RS232 Serial Port Communication; one channel is connected with Signal PE0(RXD0) and PE1(TXD0); and other one is connected with PD2(RXD1) and PD3(TXD1); so, user can connect with RS232 easily.
- Has circuit to connect with Character LCD Display (ET-CLCD) and VR to adjust contrast or brightness of LCD; circuit is connected with 4Bit Interface LCD.
- Has Circuit +5V/2A Regulate to be Power Supply for Circuit LCD Display and I/O Devices that are compatible with +5V Power Supply, including red LED to display the operating status.
- Be small PCB Size: 8 x 6 cm.

Structure of Board

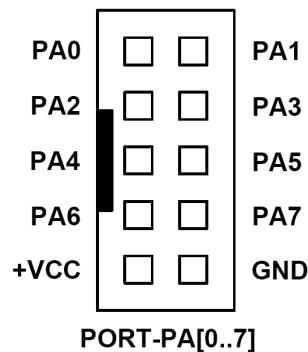
- **No.1:** It is AVR MCU No.ATmega64 or ATmega128 from ATMEL.
- **No.2:** It is Switch RESET to reset the operation of MCU.
- **No.3:** It is Crystal 16MHz.
- **No.4:** It is VR to adjust contrast or brightness of LCD.
- **No.5:** It is Port AVR ISP(6PIN) to download HEX File into MCU.
- **No.6:** It is Port AVR ISP(10PIN) to download HEX File into MCU.
- **No.7:** It is 8Bit PORTC that is PC0-PC7.
- **No.8:** It is 8Bit PORTA that is PA0-PA7.
- **No.9:** It is 8Bit PORTF that is PF0-PF7.
- **No.10:** It is 8Bit PORTE that is PE0-PE7.
- **No.11:** It is 8Bit PORTB that is PB0-PB7.
- **No.12:** It is 8Bit PORTD that is PD0-PD7.
- **No.13:** It is Port ET-CLCD to connect with Character LCD that is 4Bit Interface.
- **No.14 and 15:** It is Connector RS232 for general use.
- **No.16:** It is Jumper to choose between RS232 or Port I/O.
- **No.17:** It is Connector Power Supply for board.
- **No.18:** It is LED Power to display the operating result of +5VDC Power Supply.

## Connectors

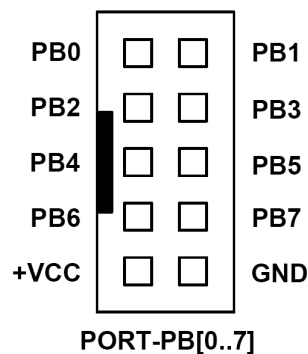
It has designed and arranged Connectors of Port I/O from MCU through 6 of Connectors IDC-Header 10PIN (2x5) that are PA, PB, PC, PD, PE, and PF respectively. Connectors in each set consist of Signal I/Os that are directly connected from Pin of MCU. The connecting points that are used to interface with external signal are listed below;

- Connector Power Supply for board
- Connector 8Bit PORTA that is PA0-PA7
- Connector 8Bit PORTB that is PB0-PB7
- Connector 8Bit PORTC that is PC0-PC7
- Connector 8Bit PORTD that is PD0-PD7
- Connector 8Bit PORTE that is PE0-PE7
- Connector 8Bit PORTF that is PF0-PF7
- Connector ET-CLCD to connect with Character LCD
- 2-Channel Connector RS232; one channel is connected with PE0(RXD0) and PE1(TXD0); and other one is connected with PD2(RXD1) and PD3(TXD1); so, user can connect with RS232 easily.
- Connector AVR ISP to download HEX File into MCU

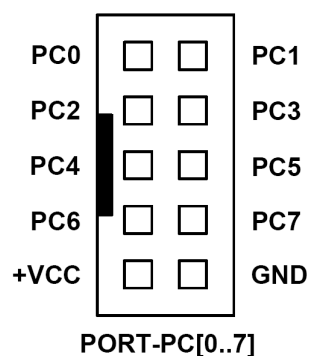
### 8BIT PORT PA



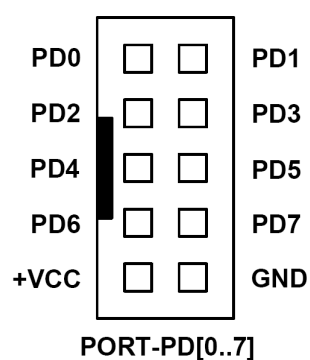
### 8BIT PORT PB



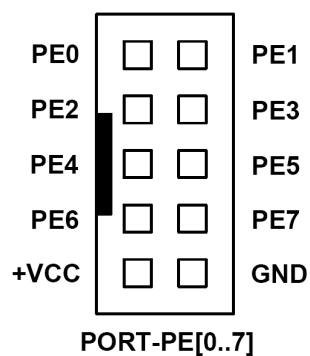
## 8BIT PORT PC



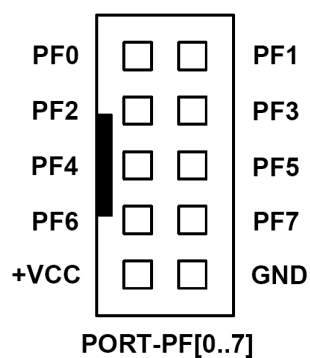
## 8BIT PORT PD



## 8BIT PORT PE

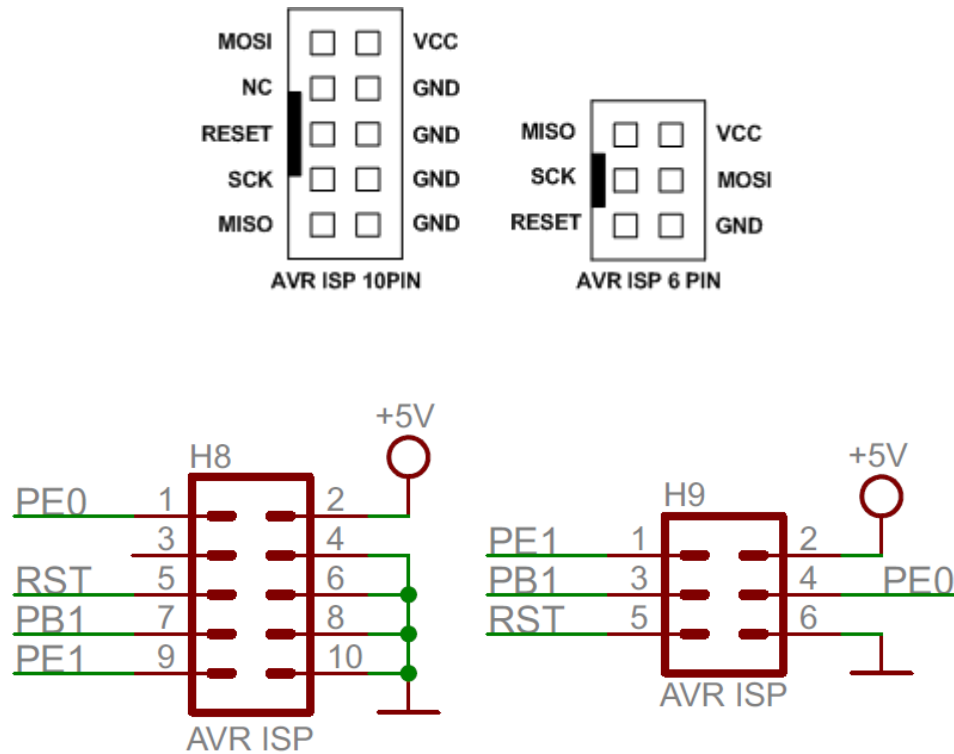


## 8BIT PORT PF



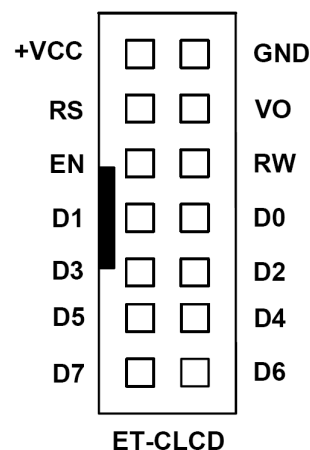


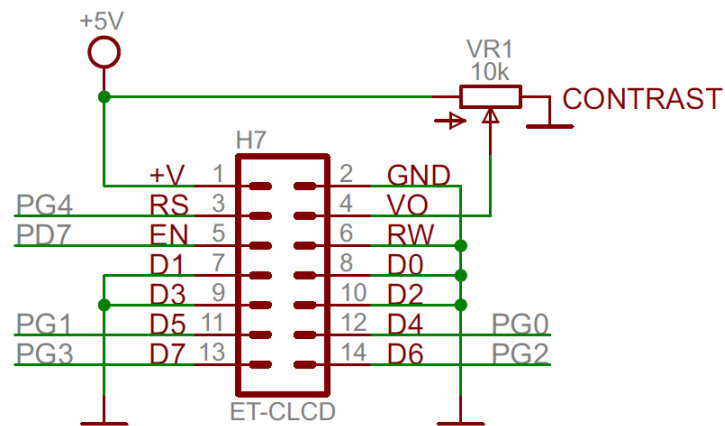
## Port AVR ISP



Figures display circuit to connect with AVR ISP.

**Port ET-CLCD:** It is used with Character LED as 4Bit Interface and it uses signals from Port PG and PD(PD7) to connect with LCD. It connects Cable from Connector of Port LCD to LCD Display; in this case, it uses the pin names to be the reference point; so, all 14 Cables must be agreed with all 14 pin names.

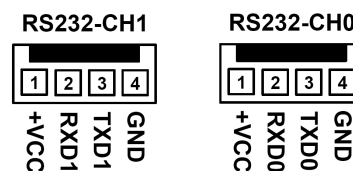




1	2	3	4	5	6	7	8	9	10	11	12	13	14
GND	+VCC	VO	RS	RW	EN	D0	D1	D2	D3	D4	D5	D6	D7

Figure displays how to arrange pin of Character LCD as standard type.

**2-Channel Port RS232:** There are 2 channels; one channel is connected with Signal PE0(RXD0) and PE1(TXD0); and other one is connected with Signal PD2(RXD1) and PD3(TXD1).



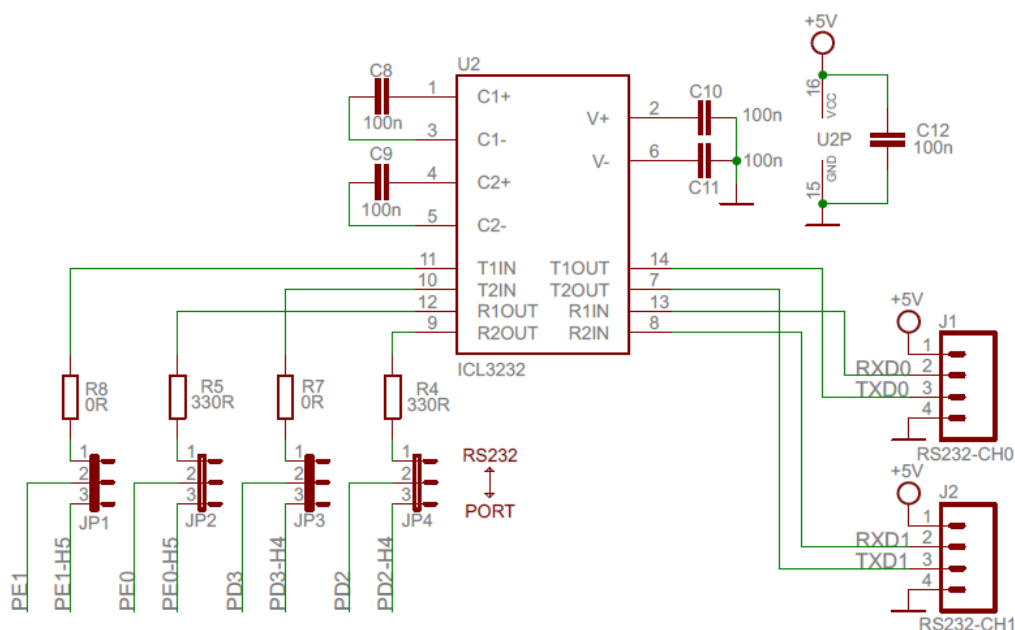
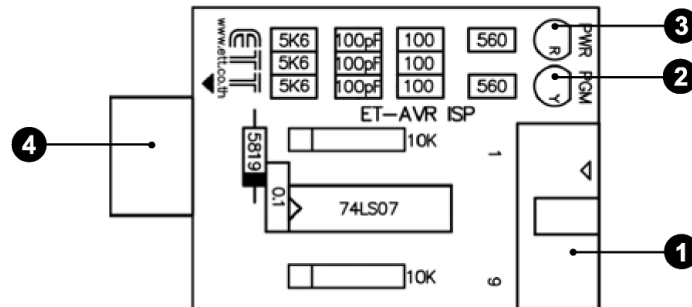


Figure displays the part of circuit that is connected with RS232.

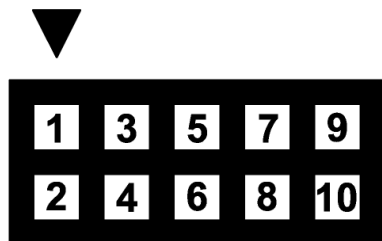
## How to download HEX File into MCU

If user requires download HEX File into AVR MCU of ATMEL through Serial Programming, it has to use ET-AVR ISP or other ISP Programmers such as AVRISP of AMTEL. If using ET-AVR ISP, it downloads HEX File through Parallel Port of computer; moreover, it has to use with ET-CAP10P of ETT and Software PonyProg200. This PonyProg2000 is the program to download HEX File data into AVR CPU through Serial Programming and is compatible with AVR Board from ETT well. User can read more information about general use of program from Help of Program by self. In this case, we will mention about how to setup Program PonyProg2000 to use with AVR Board from ETT; moreover, it is compatible with all AVR Boards from ETT.

## Structure of Board ET-AVR ISP



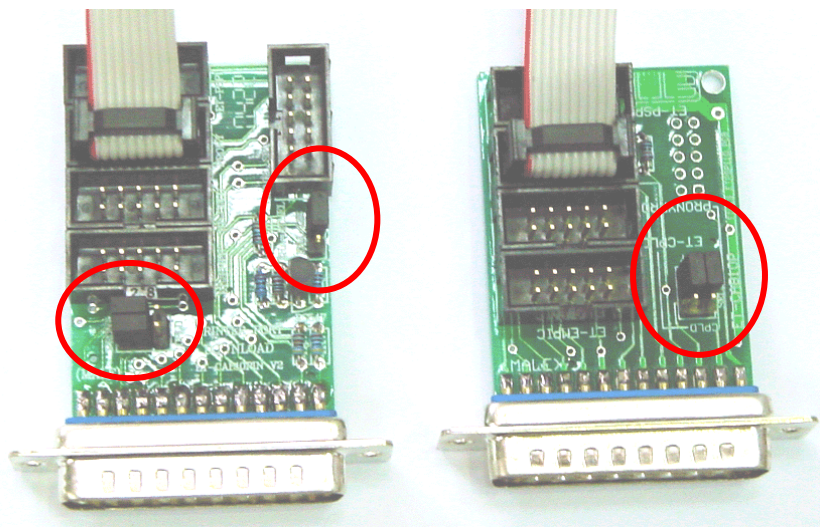
- **No.1:** It is Port for interfacing with ET-CAP10P of ETT to program HEX File into MCU.
- **No.2:** It is LED PGM (Green) to display the status of programming or downloading HEX File into MCU.
- **No.3:** It is LED PWR (Red) to display the operating status of Power Supply for board.
- **No.4:** It is Port to connect with Target Board that can program HEX File into board ET-BASE AVR ATmega64/128 r3 by plugging Board ET-AVR ISP in Port AVR ISP. It arranges pins as shown in the picture below;



Pin Position	Signal Name
1	MOSI
2	VCC
3	Unused
4,6,8,10	GND
5	RESET
7	SCK
9	MISO

## How to connect devices to program HEX File

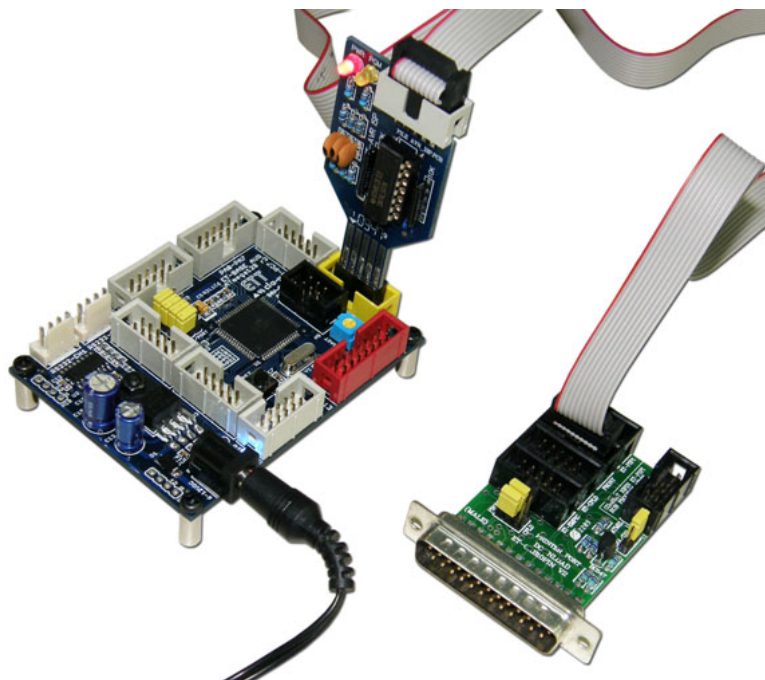
If user requires programming HEX File Code into AVR MCU, it has to use with ET-CAP10PIN and Program PonyProg2000. It connects ET-CAP10PIN with Printer Port; sets Jumper for using with Program PonyProg2000; connects Cable Download at Connector AVR ISP Download of board; and then supplies power into the board completely. If there is any external device is connected with Port PB, user has to remove it from the port first. The connection's feature is shown in the picture below;



(Left) ET-CAP10P V2.0

(Right) ET-CAP10P V1.0

Picture shows how to set Jumper and interface Cable Download of ET-CAP10P for using with AVR.



Picture shows how to connect ET-AVR ISP with ET-BASE AVR ATmega64/128 r3 by connecting both boards together. Please look at position of PIN 1, it must agree.

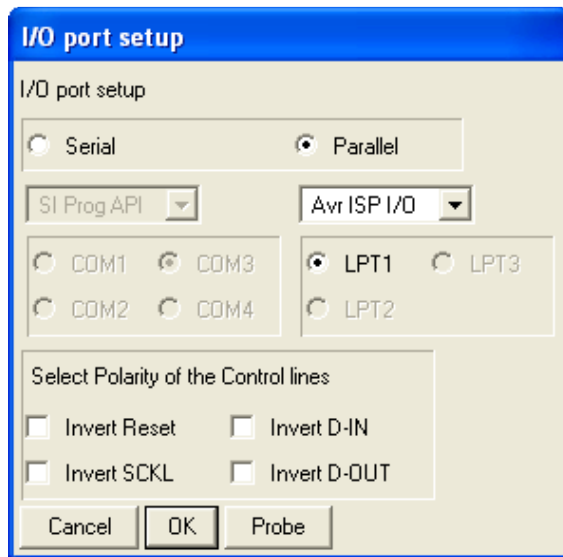
### How to program Board ET-BASE AVR ATmega64/128 r3 by Program PonyProg2000

Program PonyProg2000 is the program to download HEX File Data into AVR CPU through Serial Programming and it is compatible with AVR Board from ETT well. User can read more information about general use of program from Help of Program by self. In this case, we will mention about how to setup Program PonyProg2000 to use with AVR Board from ETT; moreover, it is compatible with all AVR Boards from ETT.

Please be careful in using AVR CPU No.ATmega64/128, there are many Fuse Bits internal structure of ATmega64/128 that are used to set conditional operations of CPU. Some Fuse Bits have effect on Serial Programming Download; if user sets these Fuse Bits wrongly, it is unable to program any CPU by Serial Programming any more. In this case, user needs to edit these Fuse Bits by Parallel Programmer correctly first.

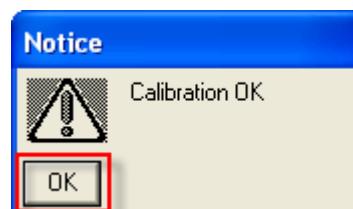
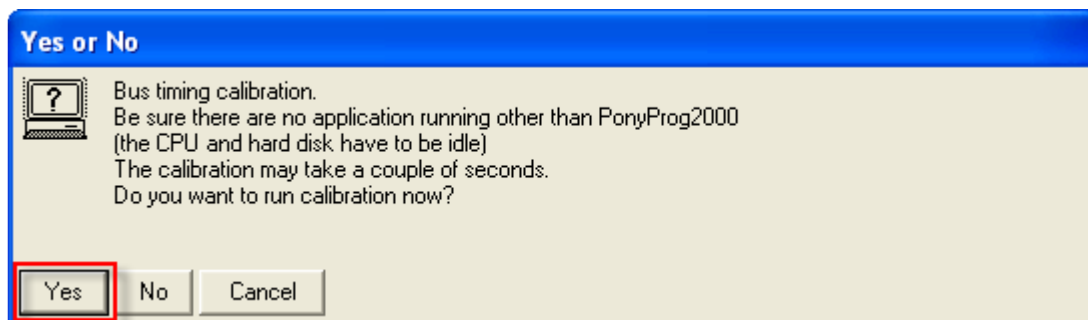
If using Program PonyProg2000 to program AVR CPU that is used with ETT Board, it needs to set Option of program as follows;

1. Click **Setup** → **Interface Setup...** as follows;



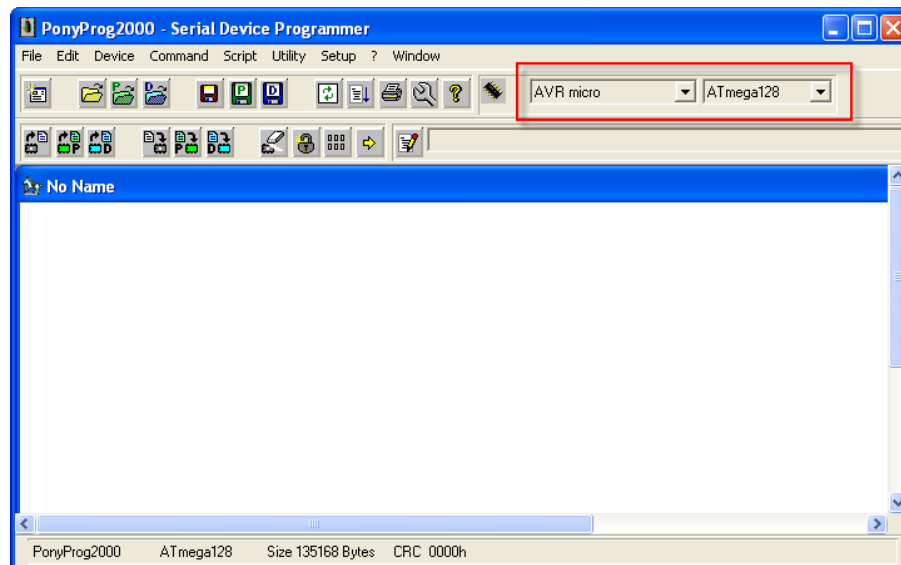
- Set **I/O Port** as **Parallel** and the Programming type as **Avr ISP I/O**
- Set Printer Port according to actual connection; for example, LPT1 if using with Printer Port LPT1
- Do not set any value in the part of **Polarity Control Line**
- Setup only one time when using the program in the first time

2. Set Program PonyProg2000 to calculate the proper speed for transmitting signal to program CPU, click **Setup** → **Calibration**.

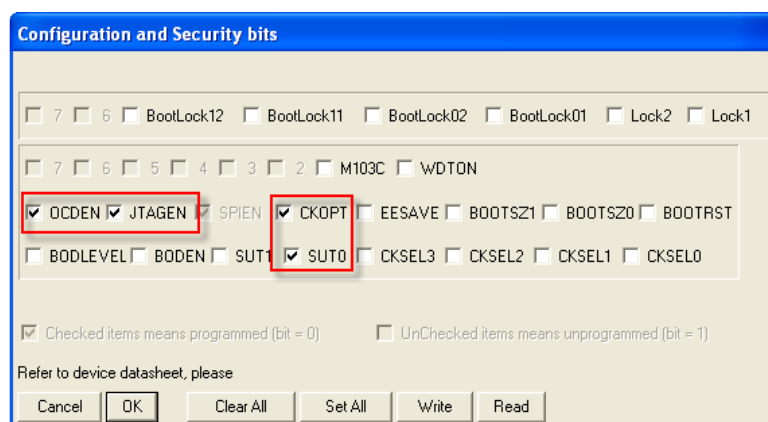


- Calibration only one time when using the program in the first time

3. Set CPU number, click **Device** → **AVR Micro** → **Atmega64** or **ATmega128**



4. Click **Command** → **Security and Configuration Bits**. If it is other AVR numbers, user can set the operation of Fuse Bit as required and user can read more information of the Fuse Bits from Data Sheet of CPU by self. If using with ATmega64/128, please be careful in setting Fuse Bits correctly, otherwise user cannot program any CPU through Serial Programming any more. After set values as shown in the picture completely, click Button Write. (This process has been done only one time and user can skip this process when entering later. If using Board ET-BASE AVR ATmega64/128 r3, it is unnecessary to do this process again because ETT has already set the values, except user requires changing the value.)



Picture displays how to set Fuse Bit to use with CPU No.ATmega64/128.



## Meaning of Fuse Bits of ATmega64/128

- ❑ If ticked [✓] in front of any Fuse Bit, it means that it sets the Fuse Bit to be "0" or it programs the Fuse Bit.
- ❑ If did not tick [✓] in front of any Fuse Bit, it means that it sets the Fuse Bit to be "1" or it does not program the Fuse Bit.

## Meaning of Fuse Bits of ATmega64/128 that has effect on Serial Programming

- **SPIEN:** It is Serial Programming Enable Bit and user always sets this Fuse Bit because it is able to download program into CPU through In-System Serial Programming. Normally, this Fuse Bit has been set from the factory and it cannot be erased or edited by any Serial Programming Mode. If CPU has been programmed by Parallel Programming, user does not forget to set and program this Fuse Bit.
- **OCDEN and JTAGEN:** Both bits are used in case of debugging the operation of MCU, programming data through JTAG Interface, and it has to use with AVR JTAG Debugger. If user does not use this operation, it is unnecessary to choose both bits.
- **CKOPT:** It is Oscillator Option Bit. If this Fuse Bit has been programmed, it sets CPU to run at Frequency 16MHz; on the other hand, if this Fuse Bit has not been programmed, it sets CPU to run at Frequency not higher than 8MHz. If using with the standard Board from ETT that uses XTAL to be the Frequency Generator, user should program this Fuse Bit because it makes CPU run at XTAL frequency in the range of 1.0MHz-16.0MHz.
- **CKSEL3...0:** It is Select Clock Source Bit that is used together to choose Generator and frequency range for using with CPU. If using with standard Board of ETT, it has to set the value as External Crystal 1.0MHz-16.0MHz, otherwise it makes the operation of program error. If user sets the Frequency Generator wrongly; for example, if user sets the value as External Clock or External RC Oscillator, it makes CPU cannot work because it has not been connected with external Signal Clock. So, it is unable to program CPU through Serial Programming any more. User has to edit Fuse Bit of the CPU correctly and it needs to choose the Frequency Generator as External Crystal.

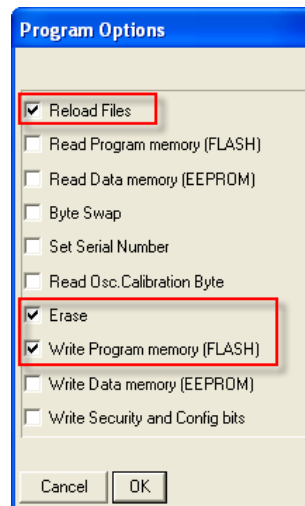
Clock Generator of AVR ATmega128	Setting Fuse Bit of CKSEL[3...0] (0=Program,1=Un-Program)
External Crystal/Ceramic Resonator	1111-1010
External Low Frequency Crystal	1001
External RC Oscillator	1000-0101
Calibrated Internal RC Oscillator	0100-0001
External Clock	0000

Table shows how to choose Frequency Generator from Fuse Bit CKSEL [3...0].

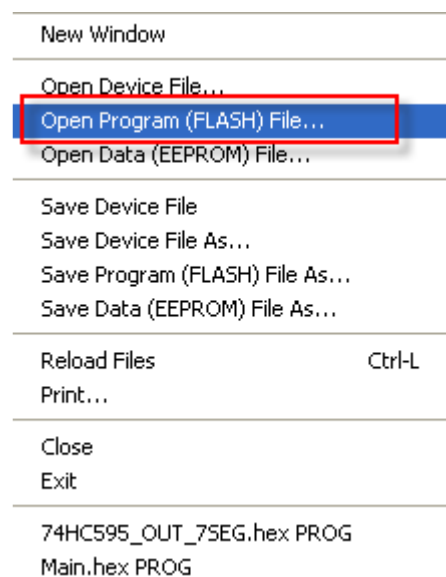
**NOTE:**

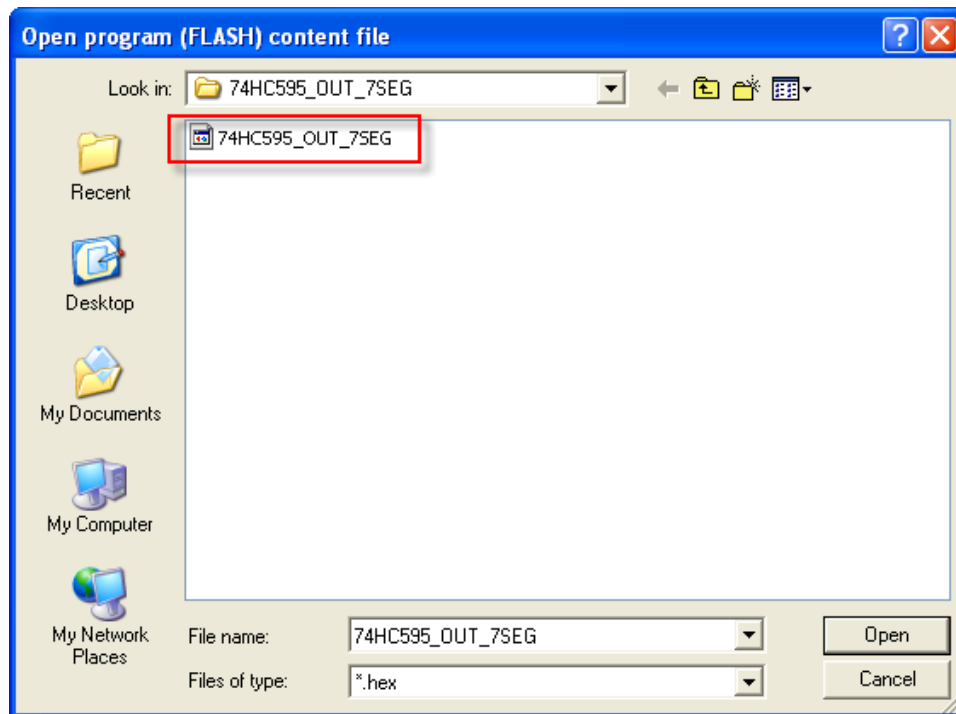
- **Value "1"**: It means that it does not program the Fuse Bit and there is no any tick sign [√] in front of the Fuse Bit.
- **Value "0"**: It means that it programs the Fuse Bit and
- Should program the Fuse Bit of CKOPT to run at the Frequency 1.0MHz-16.00MHz
- Not allowed to program Fuse Bit of CKSEL[3..0] because it makes the operation error. For example, if user programs all Fuse Bits of CKSEL[3..0] as 0; after Program PnonyProg2000 wrote this Fuse Bit into CPU completely, it is unable to use CPU with ETT Board anymore; moreover, it is unable to program and edit any new Fuse Bit into CPU through Serial Programming because CPU is unable to work. Remember, if user has programmed all Fuse Bit of CKSEL[3..0] as "0", it commands CPU to run with frequency of External Clock, so it makes the Frequency Generator of External Crystal stop running; in this case, CPU waits to receive the External Clock only. However, ETT Board only uses Signal Clock from Circuit Crystal (External Crystal); so, if CPU cannot start running, it is unable to program or edit any Fuse Bit for CPU correctly through Serial Programming. It has to edit Fuse Bit of the CPU by Parallel Programmer first and user can use CPU with Serial Programming again.
- Set the Fuse Bit position of Lock[2...1] as required.

5. Click **Command → Program Option** as follows;

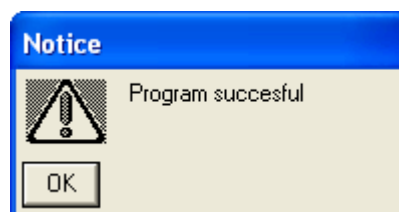
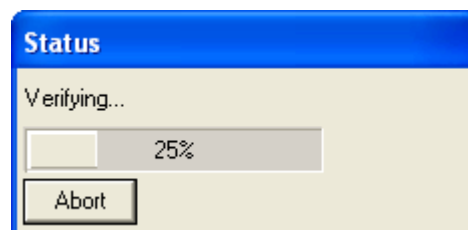


6. Open file to program into CPU, click **File** → **Open Program (FLASH) File...** → specify name and location of HEX File to program completely.





7. Start programming data into CPU, click **Command** → **Program**; and program start running as chosen in the step 5 that are **Load File** → **Erase** → **Write Program memory (FLASH)** respectively. User needs to wait until the operation of program runs completely.



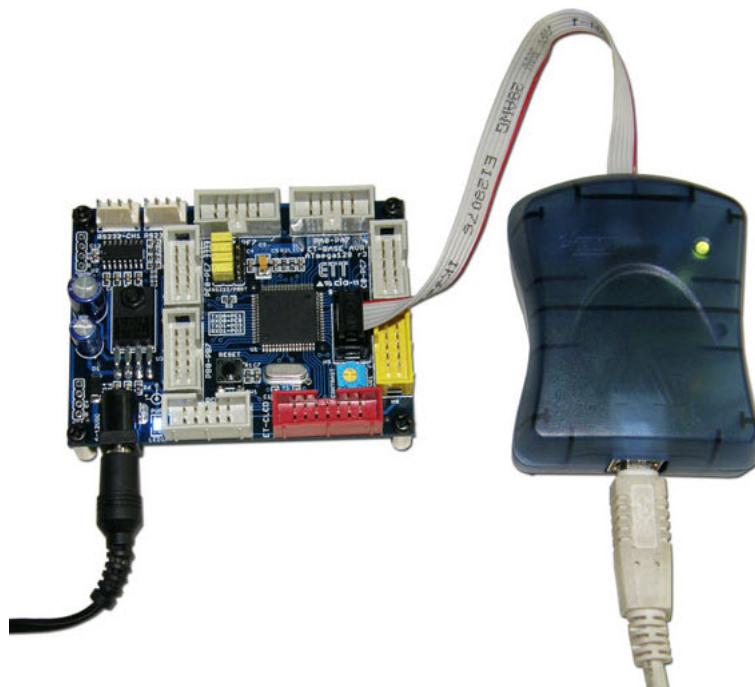
After programmed successfully, CPU starts running according to the downloaded data instantly.

## How to initially check if be unable to download program

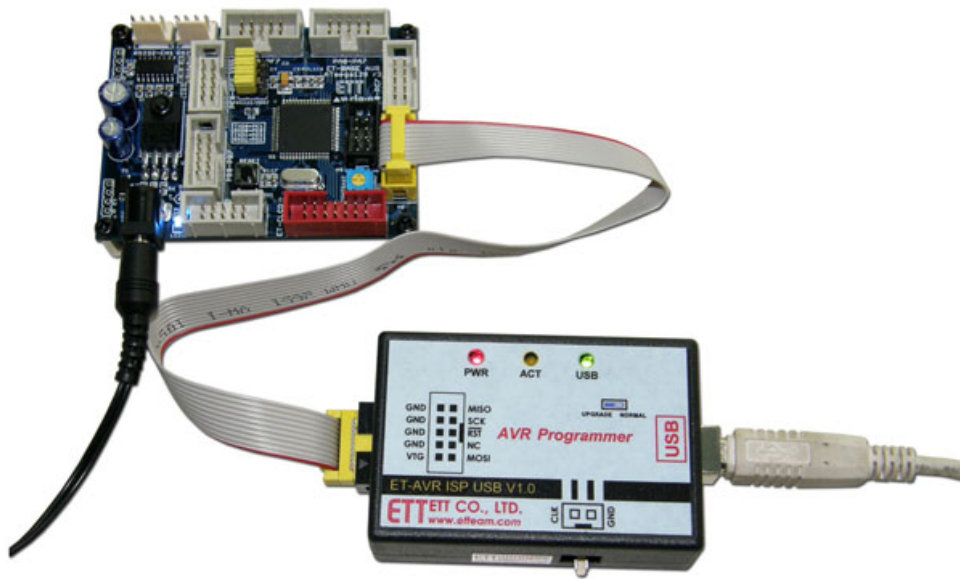
If any error occurred while programming, please check the problem as follows;

- Read manual of board and learn how to download Program AVR by PonyProg2000 throughout
- Check the Cables' connection; if downloading program through PonyProg2000, it has to use with Cable Download "ET-CAP10PIN" of ETT and then sets Jumper to be "PonyProg" correctly (see more detailed in Page 9).
- Check Power Supply of board
- Check setting values of program to download PonyProg2000
- Check whether signal from PORT-PB of CPU is connected with external devices while downloading or not. For example, Signal from Port PB is connected with LED or other circuits while downloading data.

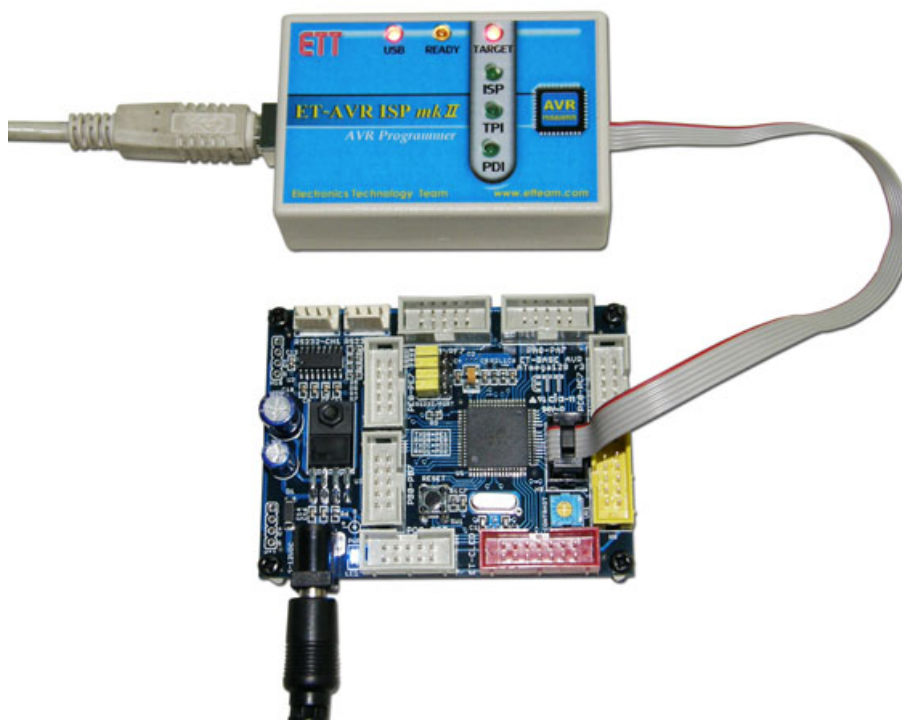
## How to use with other Programmers



Example of using with AVRISP mkII from ATMEL



Example of using with ET-AVR ISP USB V1.0 from ATMEL



Example of using with ET-AVR ISP mkII from ETT