

ET-BASE AVR ATmega64/128

ET-BASE AVR ATmega64/128 which is a Board Microcontroller AVR family from ATMEL uses MCU No.ATmega64 and ATmega128 64PIN. Board ET-BASE AVR ATmega64/128 uses MCU's resources on board mainly and arranges Pin I/O as Port PA, PB, PC, PD, PE, and PF, Port ET-CLCD, and including Port for downloading Program, it makes us more convenient to apply for various project works. Moreover, there's addition circuit Driver RS-232, so we can use RS-232 Serial Port Communication easily and conveniently.

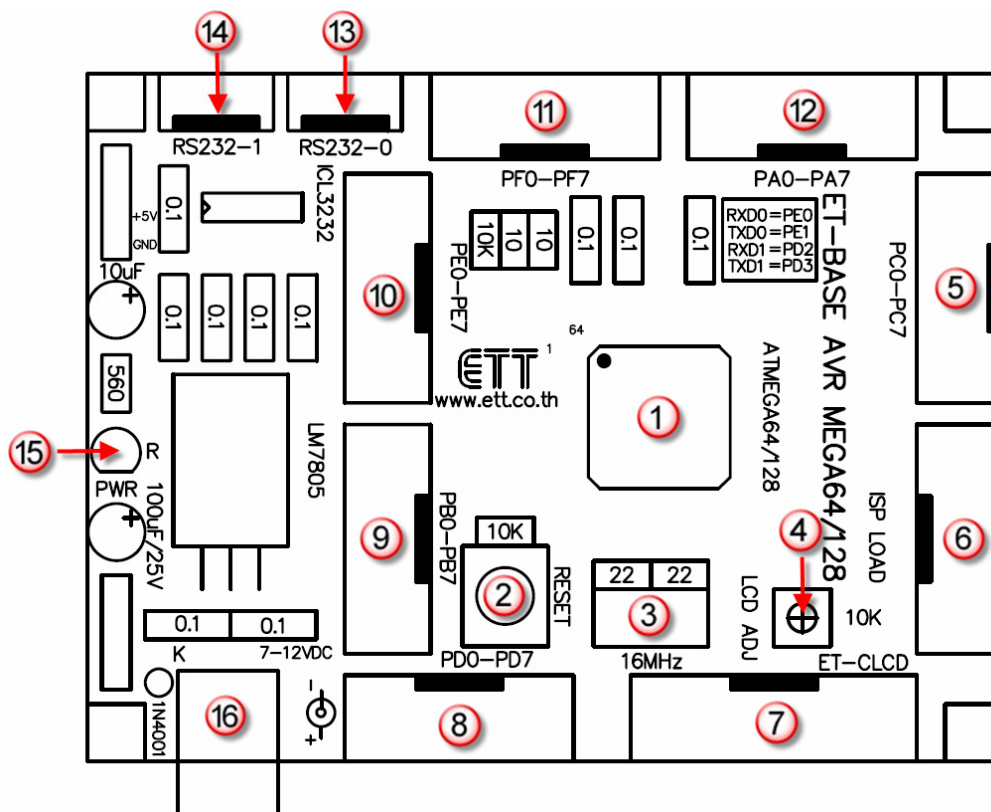
Specifications

- Use MCU AVR family No.ATmega64, ATmega128 from ATMEL 8BIT to be permanent MCU on board, and XTAL 16 MHz. Moreover, this MCU has other interesting specifications as following;
 - Flash Memory 64 K Bytes for ATmega64 and 128 K Bytes for ATmega128, and RAM 4 K Bytes
 - EEPROM 2K Bytes for ATmega64 and 4 K Bytes for ATmega128, in this case, we can erase and re-write data more than 100,000 times.
 - Maximum 53 I/O Pins
 - Circuit SPI Communication 1 Channel, I2C 1 Channel, Programmable Serial USARTs 2 Channel
 - ADC 10-Bit 8 channel
 - Timers/Counter 8-Bit 2 Channel, Timers/Counter 16-Bit 2 Channel, 8-Bit PWM 2 Channel, Watchdog Timer, Real Time Counter
- 6 I/O Port 10PIN; PA,PB,PC,PD,PE, and PF
- Port ISP LOAD for program MCU (must use with ET-AVR ISP or ISP Programmer that has the same type of Pin arrangement).
- Line Driver for RS232 Serial Port Communication 2 Channel; one Port for interfacing with Signal PE0(RXD0) and PE1(TXD0) and other one Port for interfacing with Signal PD2(RXD1) and PD3(TXD1), in this case, we can test connection of circuit RS232 Serial Port Communication easier.
- Circuit for connecting with Character LCD Display Monitor (ET-CLCD) and VR for contrasting brightness of LCD, in this case, it is

interfaced circuit with LCD as 4 Bit Interface type.

- Circuit Regulate +5V/1A for supplying power into circuit of LCD Display Monitor and I/O components for using with Power Supply that is not much than +5V and red LED.
- PCB Size 8 x 6 cm.

Structure of Board



- **No.1** is MCU No.ATmega64 or ATmega128 AVR family from ATMEL.
- **No.2** is Switch RESRT to reset operation of MCU.
- **No.3** is Crystal 16 MHz.
- **No.4** is Resistor to contrast brightness of LCD.
- **No.5** is PORTC 8 Bit; PC0-PC7.
- **No.6** is Port ISP LOAD to download Hex File into MCU.
- **No.7** is Port ET-CLCD to connect with Character LCD as 4 Bit type.

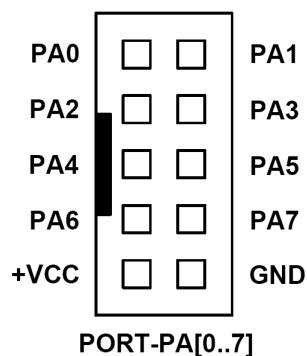
- **No.8 is** PORTD 8 Bit; PD0-PD7.
- **No.9 is** PORTB 8 Bit; PB0-PB7.
- **No.10 is** PORTE 8 Bit; PE0-PE7.
- **No.11 is** PORTF 8 Bit; PF0-PF7.
- **No.12 is** PORTA 8 Bit; PA0-PA7.
- **No.13 and 14** are Connector RS232 for general usage.
- **No.15** is LED Power to display status of Power Supply +5VDC.
- **No.16** is Connector Power Supply.

Connectors

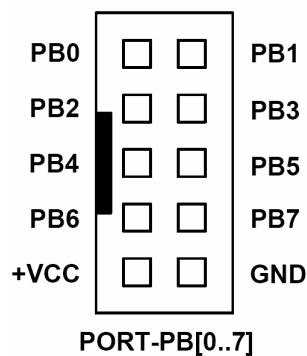
For Connector Port I/O from MCU, it is designed and is provided through 6 Connectors IDC-Header 10PIN (2X5); PA, PB, PC, PD, PE, and PF respectively. Each Connector consists of Signal I/O that is connected directly from Pin MCU. The external connecting point of board is shown as following;

- Connector Power Supply of Board
- Connector PORTA 8 Bit; PA0-PA7
- Connector PORTB 8 Bit; PB0-PB7
- Connector PORTC 8 Bit; PC0-PC7
- Connector PORTD 8 Bit; PD0-PD7
- Connector PORTE 8 Bit; PE0-PE7
- Connector PORTF 8 Bit; PF0-PF7
- Connector ET-CLCD for interfacing with LCD Character type
- Connector RS232 2 Channel; one channel for interfacing with Signal PE0(RXD0) and PE1(TXD0) and other one channel for interfacing with Signal PD2(RXD1)and PD3(TXD1), in this case, we can test connection of circuit RS232 Serial Port Communication easily.
- Connector ISP LOAD for downloading HEX File into MCU

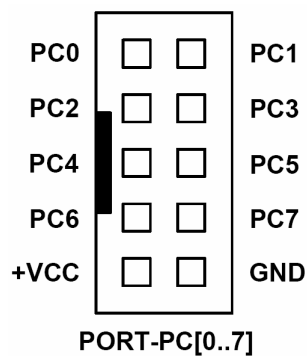
Port PA 8 Bit



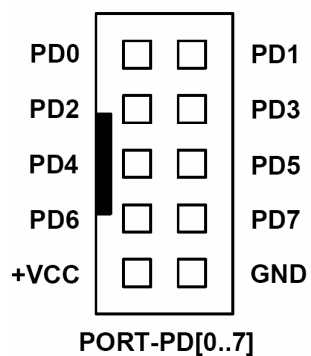
Port PB 8 Bit



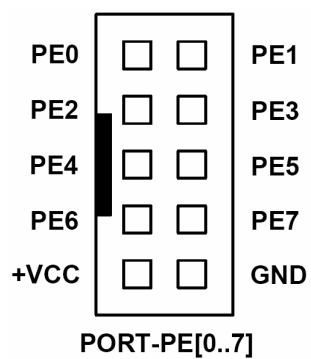
Port PC 8 Bit



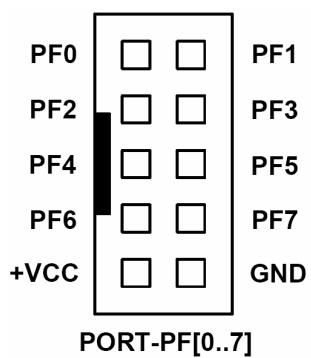
Port PD 8 Bit



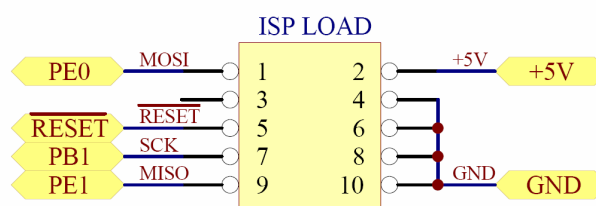
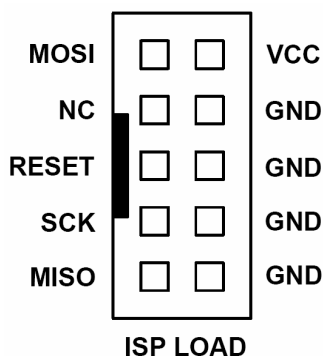
Port PE 8 Bit



Port PF 8 Bit

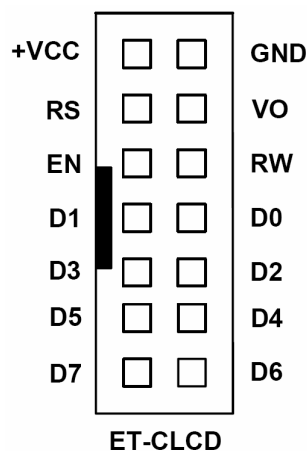


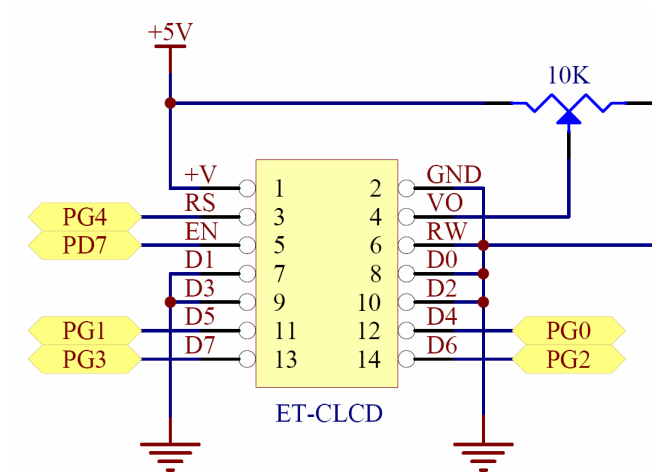
Port ISP LOAD



Picture displays **circuit for connecting with ISP LOAD.**

Port ET-CLCD: It is used with Character Type LCD for connecting as 4 Bit type. It uses signal from Port PG and PD (PD7) to connect with LCD. In this case, we must use the signal name to be reference and must interface all 14 cables corresponding with the truly name on the Connector.

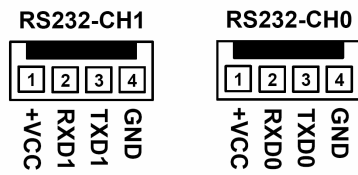


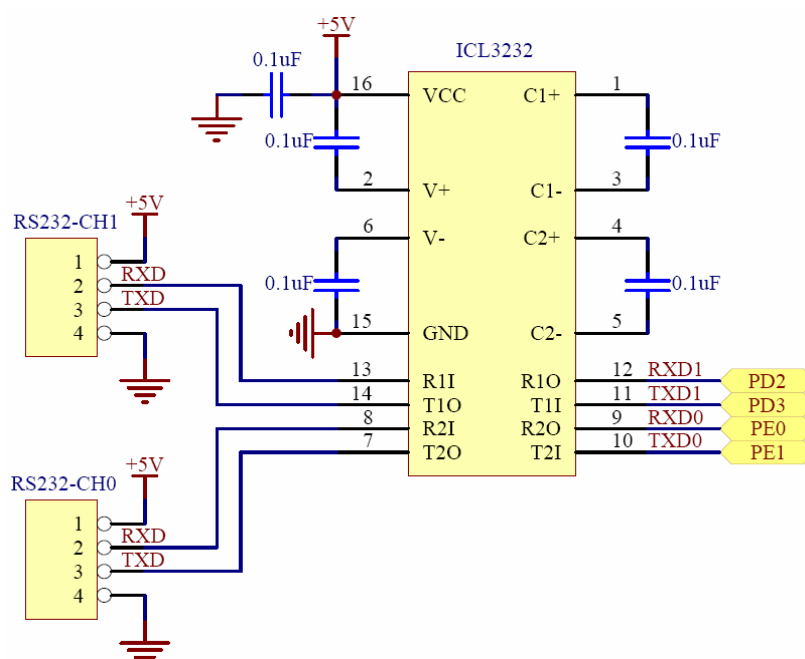


1	2	3	4	5	6	7	8	9	10	11	12	13	14
GND	+VCC	VO	RS	RW	EN	D0	D1	D2	D3	D4	D5	D6	D7

Picture displays **Pin Arrangement of Character LCD Standard.**

Port RS232 2 Channel: It is used 1 Channel to connect with Signal PE0(RXD0) and PE1(TXD0) and other one Channel for interfacing with Signal PD2(RXD1) and PD3(TXD1).





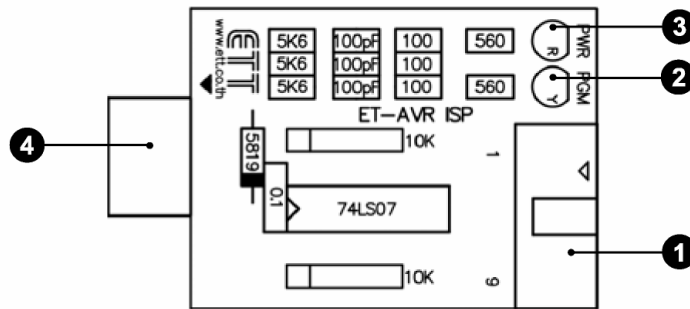
Picture displays **circuit for connecting with RS232.**

To Download Hex file into MCU

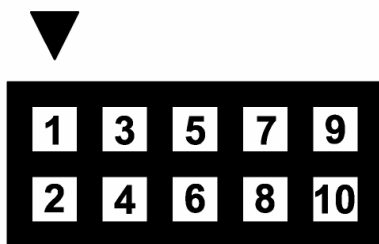
It is necessary to use ET-AVR ISP or other types of ISP Programmer such as AVRISP from ATMEL to download Hex File into MCU AVR family from ATMEL, especially procedure of Serial Programming Type.

If we use ET-AVR ISP, we must download Hex File through Parallel Port of computer and must use it with ET-CAP10P from ETT and Software PonyProg2000 program that can use with ETT Board well. User can learn general usage from menu HELP by self, in this case, we will only mention procedure to setup program PonyProg2000 for using with all versions Board AVR family from ETT.

Structure of Board ET-AVR ISP



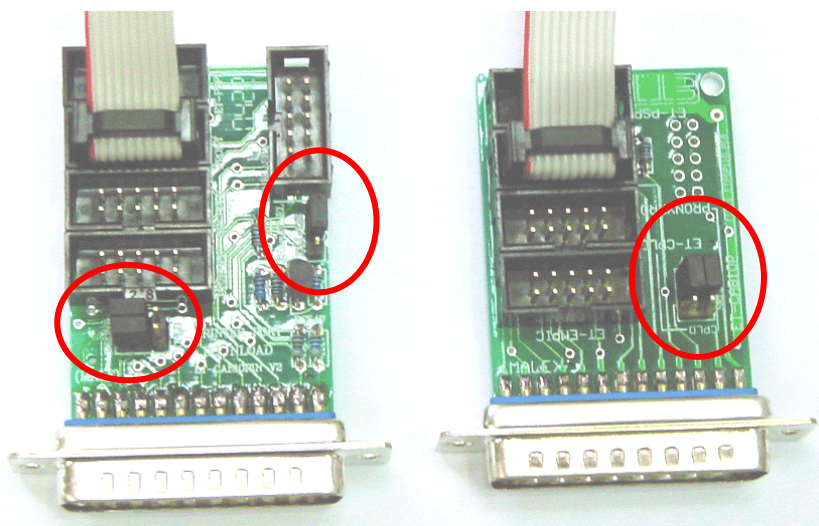
- **No.1** is Port for connecting with ET-CAP10P from ETT to program Hex File into MCU.
- **No.2** is LED PGM (GREEN) to display status of programming or to download Hex File into MCU.
- **No.3** is LED PWR (RED) to display status of supplying power on board.
- **No.4** is Port for connecting with Board Target that can program Hex File into Board ET-AVR STAMP ATmega64 by inserting Board ET-AVR ISP at the position of Port ET-PSPI. Pin arrangement is shown as in the picture below;



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

To Connect Component for Programming Hex File

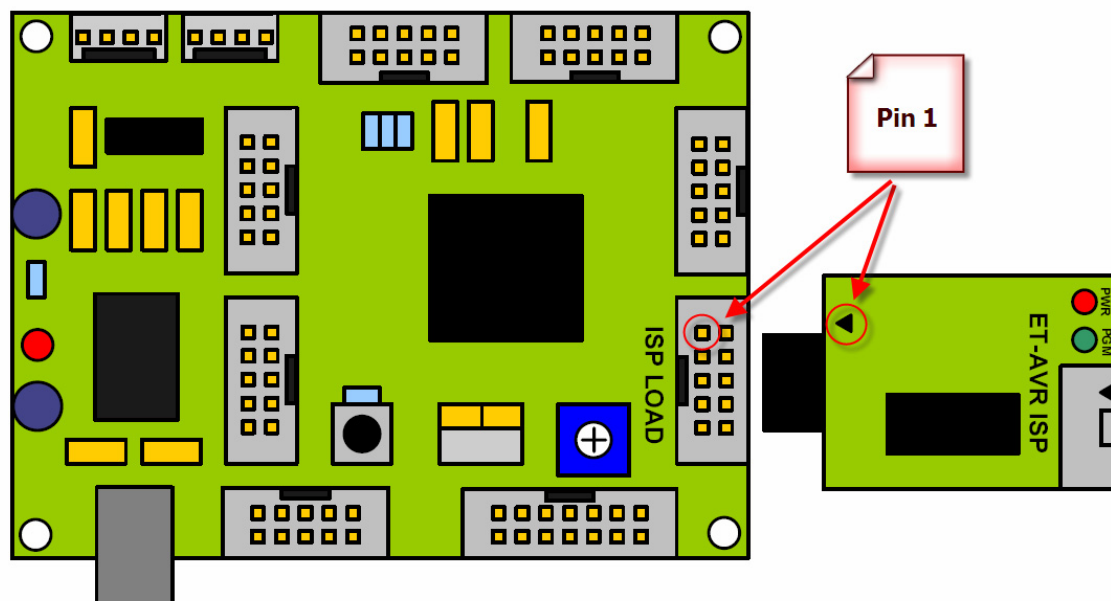
We must use ET-CAB10PIN and Program PonyProg2000 to program Code (Hex File) into AVR MCU. We must interface ET-CAB10PIN with Port Printer and then set Jumper for using Program PonyProg2000. Next, interface cable Download at the Connector ET-PSPI Download of Board, then supply power into board. We should take off all external components that interface with Port PB before programming code Hex File. The feature of connection is shown as in the picture below.



(LEFT) ET-CAP10P V2.0

(RIGHT) ET-CAP10P V1.0

Picture displays **Set Jumper and interface cable Download of ET-CAP10P for using with AVR.**



Picture displays the connection ET-AVR ISP with ET-BASE AVR ATmega64/128, in this case, Pin position of one board must be corresponding with other one board.

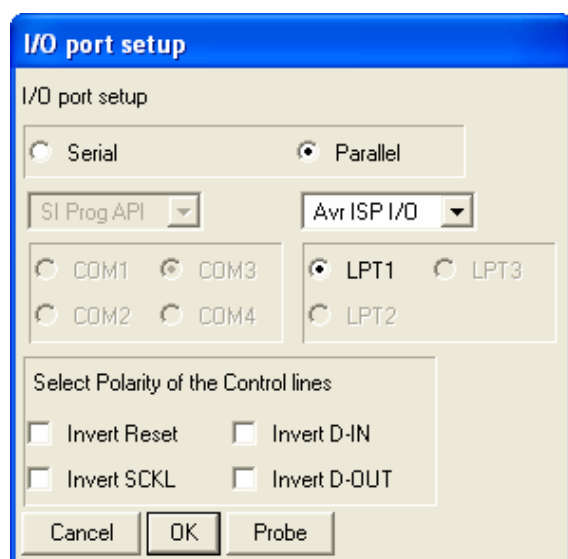
To program Board ET-BASE AVR ATmega64/128 by using Program PonyProg2000

Program Ponyprog2000 is a program for downloading data Hex File into CPU AVR family through Serial Programming Type. This method can be used with all versions of board AVR family from ETT. User can learn general usage from menu HELP by self, in this case, we will only mention procedure to setup program PonyProg2000 for using with all versions Board AVR family from ETT.

Be carefully, in case of using CPU AVR family No.ATmega64 because internal structure has many Fuse Bits to configure condition of CPU operation. These Fuse Bits are effect on downloading as Serial Programming Type. In case of selecting Fuse Bit incorrectly, we can not program CPU by Serial Programming Type. To solve this problem is modifying Fuse Bit of CPU by using Parallel Programmer first and then we can use it as usual.

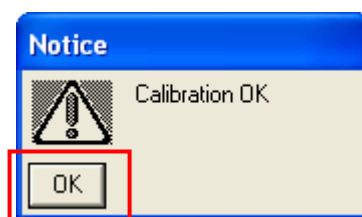
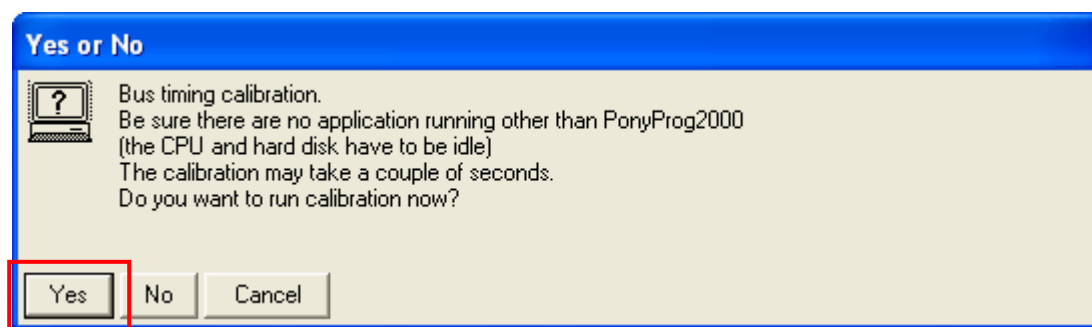
If we program CPU AVR family with ETT Board and use Program PonyProg2000, we must configure Option of program as following;

1. Configure **Setup → Interface Setup...** as following.



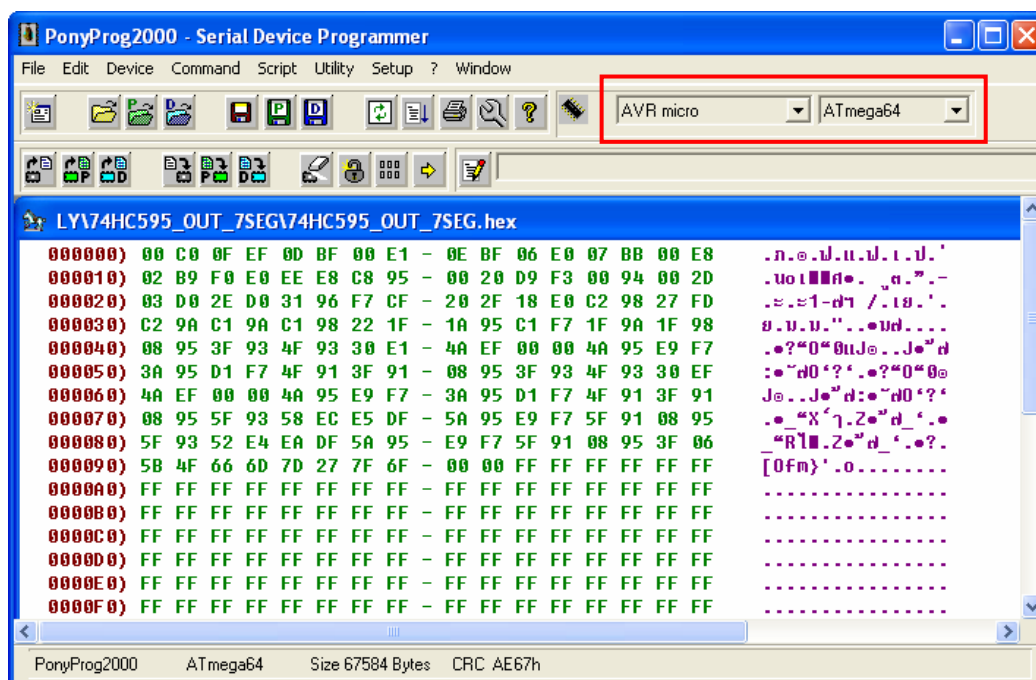
- Select I/O Port to be Parallel and then select format of programming to be Avr ISP I/O.
- Select Printer Port as truly usage such as LPT1 in case of using Printer Port LPT1.
- No select in the part of Polarity Control Line.
- Only setup program for the first time usage.

2. Command Program PonyProg200 to calculate the suitable speed for sending signal into CPU, click **Setup → Calibration.**

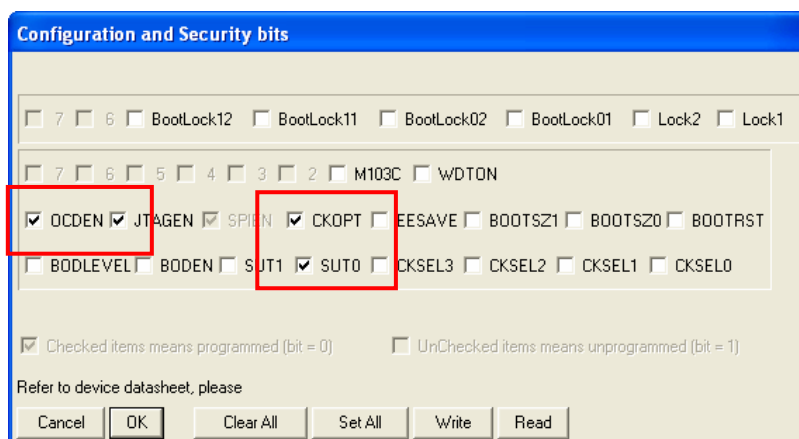


- We command to **Calibration** only one time when we use program in the first time.

3. Select CPU number; click **Device** → **AVR Micro** → such as **Atmega64**.



4. Select **Command** → **Security and Configuration Bits**, if it is other AVR number, can configure Fuse Bit as required. User can learn more detailed of Fuse Bit from Data Sheet of CPU by self. **Be carefully, if using with ATmega64, we must configure Fuse Bit correctly otherwise we can not program CPU through Serial Programming Type**, we can configure Fuse Bit as following;



Picture displays **configuration of Fuse Bit for using with CPU No. ATmega64.**

The Fuse Bit's Meaning of ATmega64

- If select or tick [√] sign in front of which Fuse Bit, it means that we configure the Fuse Bit to be "0" or Command to program Fuse Bit.
- If not select or tick [√] sign in front of which Fuse Bit, it means that we configure Fuse Bit to be "1" or Command to not program Fuse Bit.

The Fuse Bit's meaning of ATmega64 that is effect on Serial programming

- **SPIEN:** It is Serial Programming Enable Bit. We must always program this Fuse Bit because we can program CPU through In-System Serial Programming. Generally, it is configured by manufacturer and can not be erased or modified by Mode Serial Programming. If CPU is programmed by Parallel Programming, do not forget to command this Fuse Bit.
- **OCDEN** and **JTAGEN:** We use both Fuse Bit to debug operation of MCU and program through JTAG Interface and must use with AVR JTAG Debugger. If we do not use this Fuse Bit for operation, it is necessary to select both Fuse Bits.
- **CKOPT:** It is Oscillator Option Bit. If we select this Fuse Bit, it means that we configure CPU to

run with frequency 16MHz. If we do not select this Fuse Bit, we configure CPU to run with frequency not much than 8 MHz. If we use the standard ETT Board, we use XTAL to be frequency generator. So, we should command this Fuse Bit because CPU can run with frequency XTAL range 1.0MHz-16.0MHz.

- **CKSEL3..0**: It is Select Clock Source Bit. We use it to select generator frequency and frequency range for CPU. If using with standard ETT Board, must select External Crystal 1.0MHz-16.0MHz but if we select other value, it makes operation of program error. **Remember, if we select generator frequency incorrectly such as "External Clock" or "External RC Oscillator", it makes CPU can not run because we do not interface external Clock and we can not program CPU through Serial Programming Type.** We can solve this problem by modification Fuse Bit of CPU to be "External Crystal".

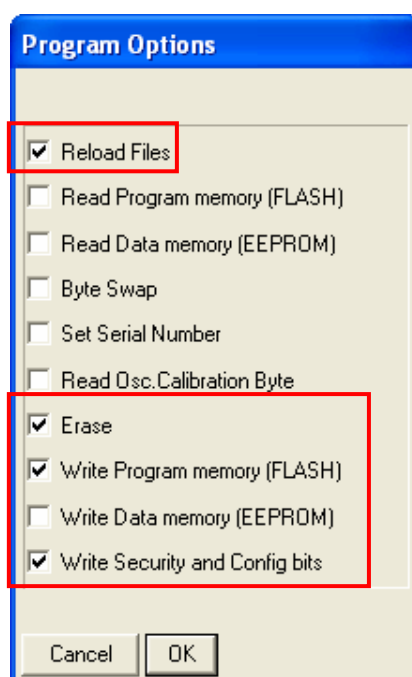
Generator Frequency AVR Atmega64	To Configure Fuse Bit of CKSEL[3..0] (0=Program,1=Un-Program)
External Crystal/Ceramic Resonator	1111-1010
External Low Frequency Crystal	1001
External RC Oscillator	1000-0101
Calibrated Internal RC Oscillator	0100-0001
External Clock	0000

Picture displays **the selection generator frequency from Fuse Bit CKSEL [3..0]**.

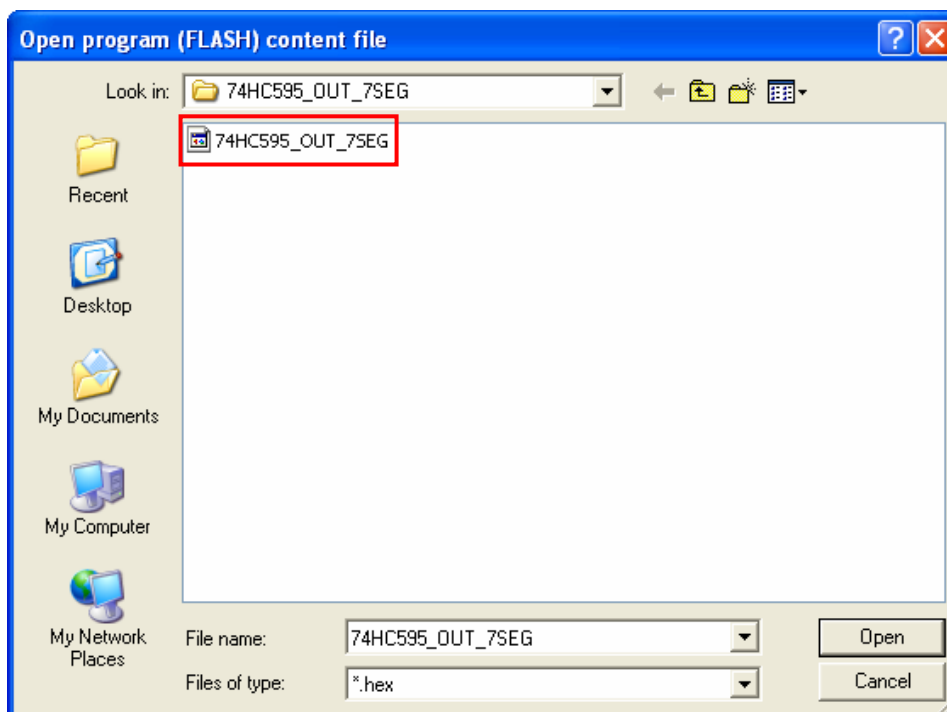
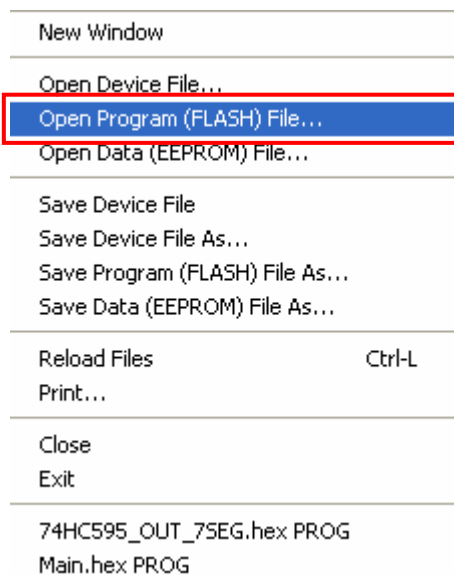
NOTE :

- Default value = 1. It means that not program Fuse Bit and not tick any mark [√] in front of Fuse Bit' name.
- Default value = 0. It means that command to program fuse Bit and must tick mark [√] in front of Fuse Bit's name.
- **Should program Fuse Bit of CKOPT to run with frequency range 1.0MHz-16.0MHz.**
- **Do not command to program Fuse Bit of CKSEL[3..0] because the operation will be error. For example, if we program all Fuse Bit of CKSEL[3..0] to be "0", after program PonyProg200 has already written default value of Fuse Bit into CPU completely, we can not use CPU with ETT Board and can not program through Serial Programming Type. If we program all Fuse Bit of CKSEL[3..0] to be "0", it configures CPU to run with External Clock, so External Crystal stops running and CPU wait for receiving External Clock only but ETT Board use only External crystal. So, we must modify Fuse Bit of CPU by using Parallel first, and then we can use CPU as usual.**
- Fuse Bit's Position of Lock[2..1] can be configured as required.

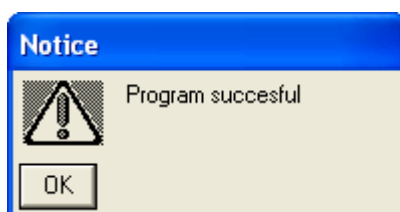
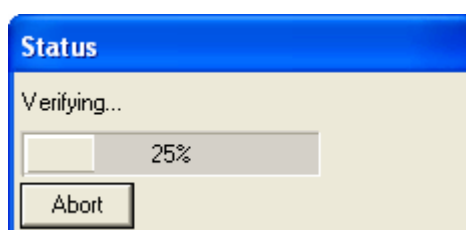
5. Select **Command** → **Program Option** as following;



6. Open Hex File that we want to program into CPU, click **File** → **Open Program (FLASH) File...** → and then assign name and address of Hex File.



7. Start to program data into CPU, click **Command** → **Program**. Then program will run as downloaded 5 Commands as following; **Load File** → **Erase** → **Write Program memory (FLASH)** and **Write Security and Config Bits** respectively. In this case, we must wait for program finished completely.



After programming completely, CPU will start operation as downloaded data in program instantly.

The basic procedure to check if can not download program

If there's error in step of programming, we can check the problem as following;

- Read User's Manual of Board and User's Manual of Download program AVR by using Program Ponyprog2000 well before using board.
- Check connection of cables and for downloading program PonyProg2000 must use with Cable Download version "ET-CAP10PIN" from ETT. We must set Jumper to be PonyProg correctly (can see more detailed from page 9).
- Check Power Supply of Board.
- Check default values of program for downloading Program PonyProg2000.
- Check signal from PORT-PB of CPU is interfaced with external component while downloading or not. For example, it is interfaced with LED or interfaced with other circuit while downloading data.
- Should not interface Cable RS232 with Port RS-232-CH0 because SPI Programming of ATmega64 uses Pin with RS232 (RXD0, TXD0).