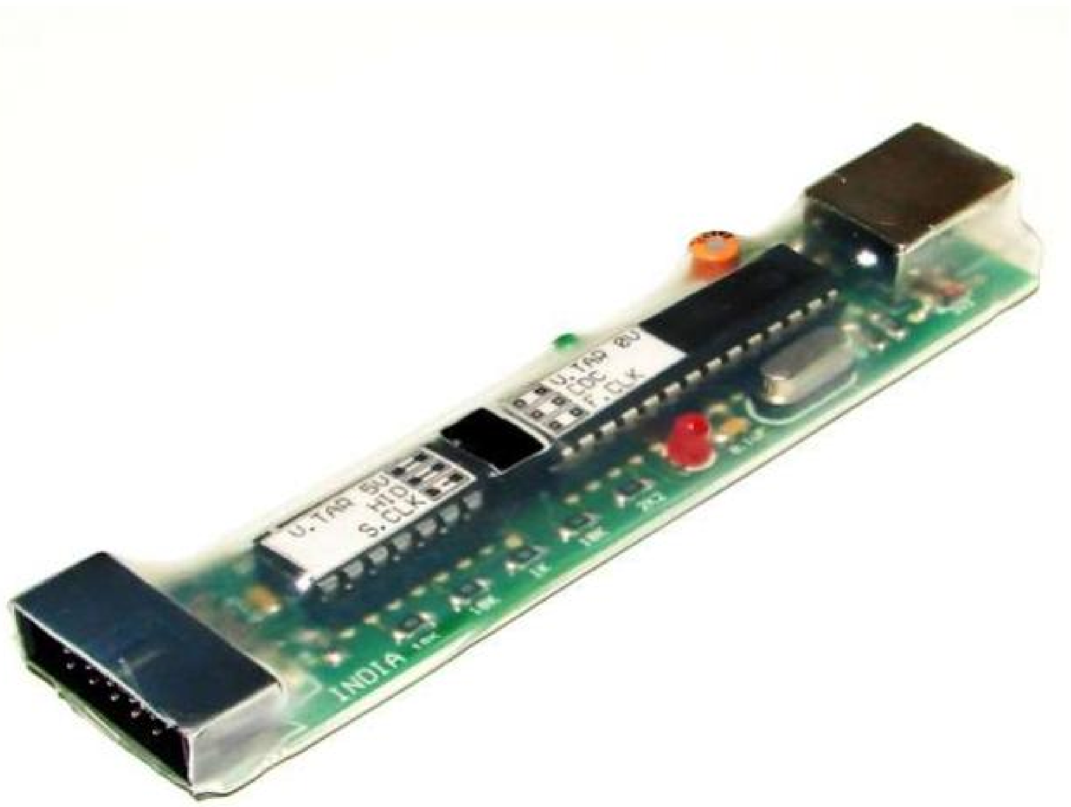
NEX AVR USB ISP STK500V2



Introduction:

NEX AVR USB ISP STK500V2 is a high-speed USB powered STK500V2 compatible In-System USB programmer for AVR family of microcontrollers. It can be used with AVR Studio on Win XP platforms. For Windows7 it can be used in HID mode with **AVRDude** command prompt as programming interface. Its adjustable clock speed allows programming of microcontrollers with lower clock speeds. The programmer is powered directly from a USB port which eliminates the need for an external power supply. The programmer can also power the target board from a USB port with a limited supply current of up to 100mA.

Note: The USB port of PC provides 5V DC. For 3.3V microcontrollers, please use appropriate voltage regulators.

NEX AVR USB ISP STK500V2 Overview

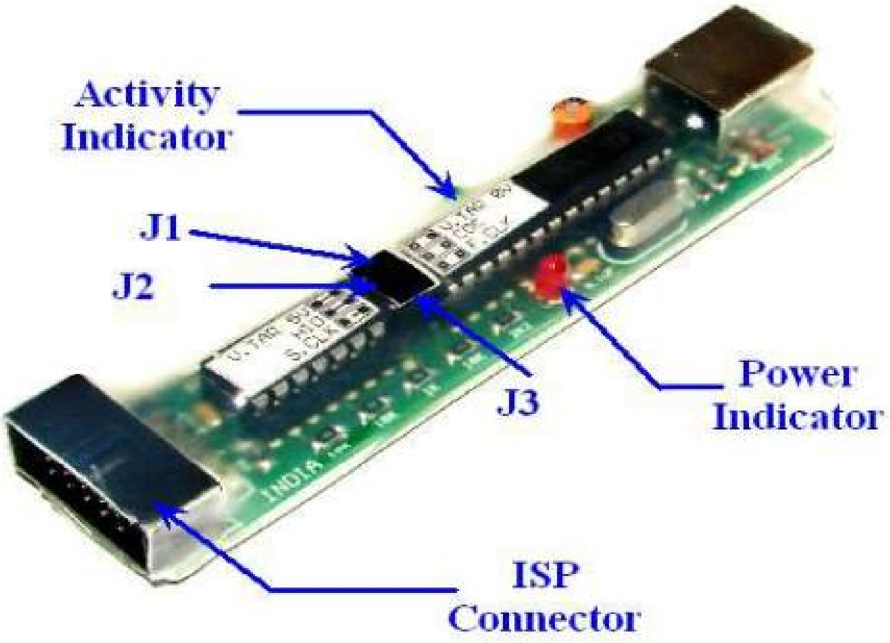


Figure 1: STK500V2 Overview

Jumper Description:

J1: If inserted, provides 5V at VTG (pin no.2) of ISP connector. If removed 0V at VTG (pin no.2) of ISP connector. **In default mode, this jumper is not inserted.**

J2: If inserted, enables UBS HID mode. If removed enables USB CDC mode. **In default mode, this jumper is not inserted.**

J3: If inserted, enables slow clock speed (for 32 KHz to 1MHz speed microcontrollers). If removed enables normal clock speed. **In default mode, this jumper is not inserted.**

No Jumper: This is the default mode of STK500v2; programming through Linux will be done in default mode.

Connections between STK500v2 and ATmega 2560:

Please follow the steps to connect STK500v2 and ATmega 2560

- Locate the ISP connector in the ATmega 2560 microcontroller as shown in the figure:

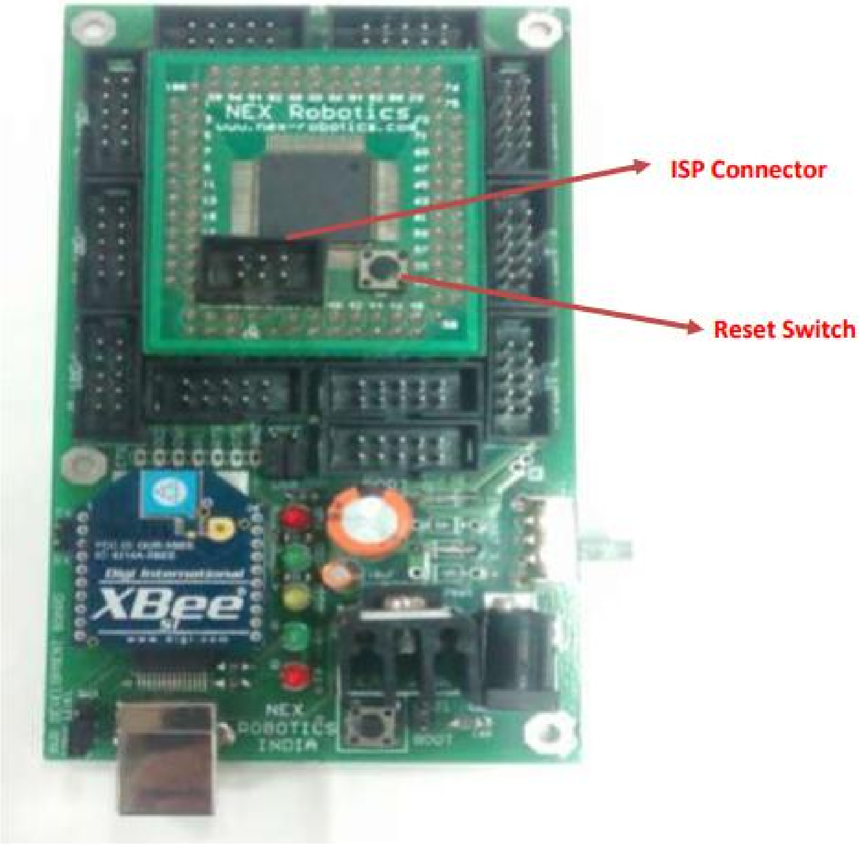


Figure 2: ISP Connector on ATMEGA2560

- Ensure that you have the following components in your kit as shown in figure:
  - STK500V2
  - 8 pin connector wire
  - 6 pin connector wire
  - 8-pin to 6-pin converter

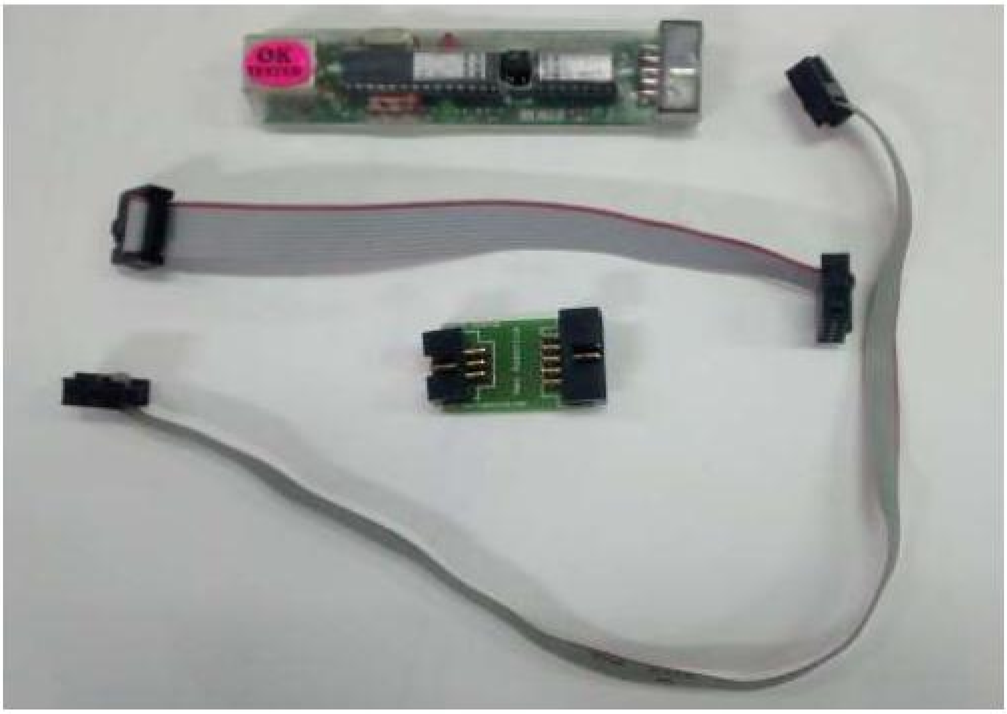


Figure 3: STK Programmer + Components

- Connect STK500V2 to the converter using 8 pin connector wire and connect the 6-pin connector wire to the other end of the converter as shown below:

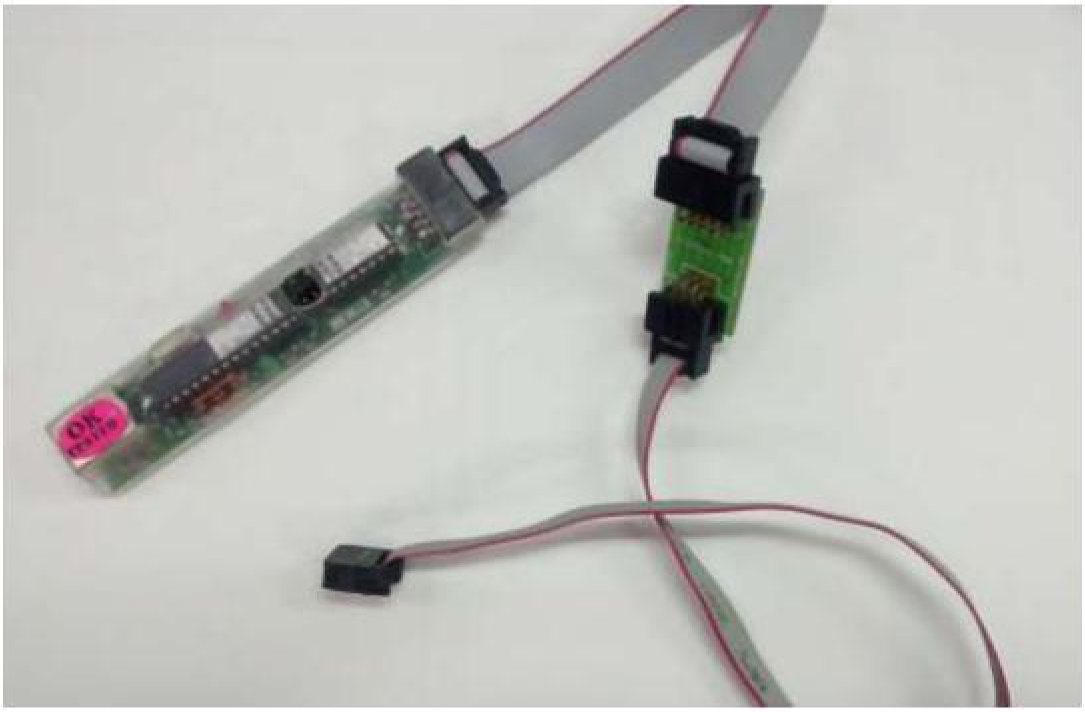


Figure 4: STK Programmer + Components

- Connect the other end of the 6-pin connector wire to the ISP connector of the ATmega 2560 as shown below:

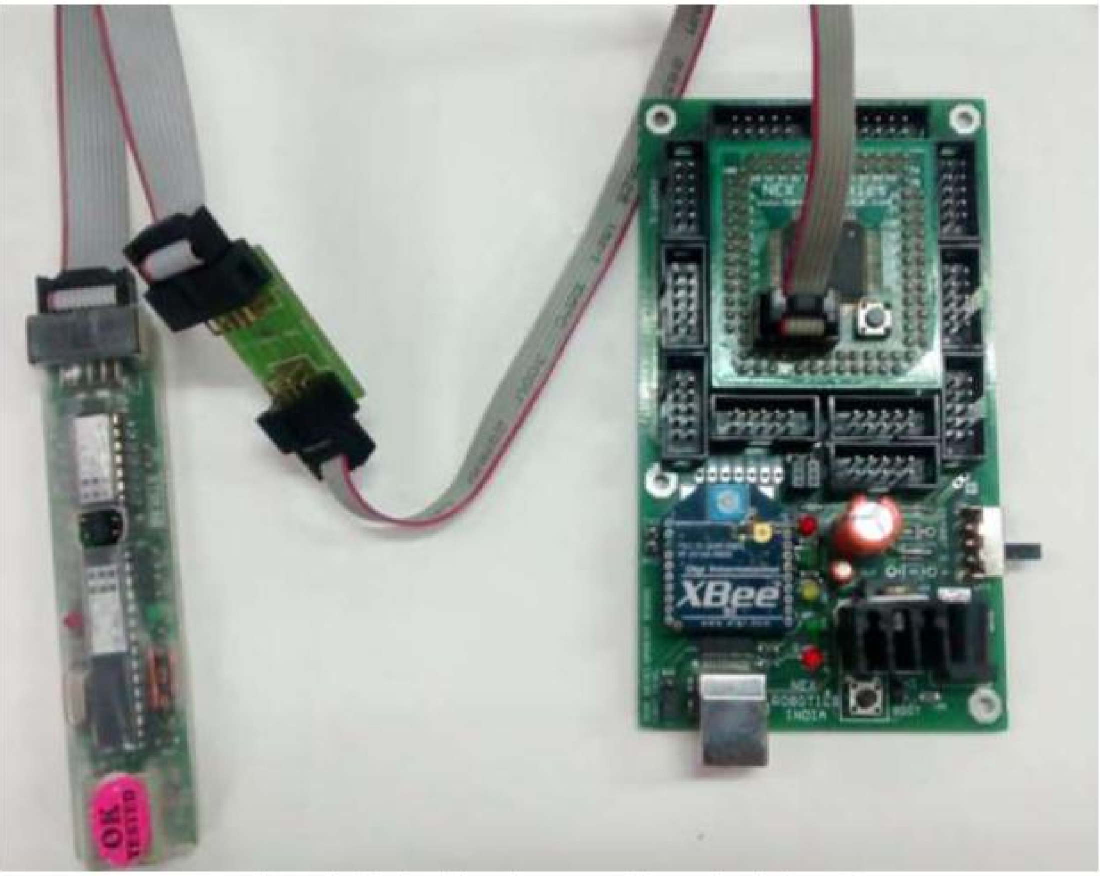
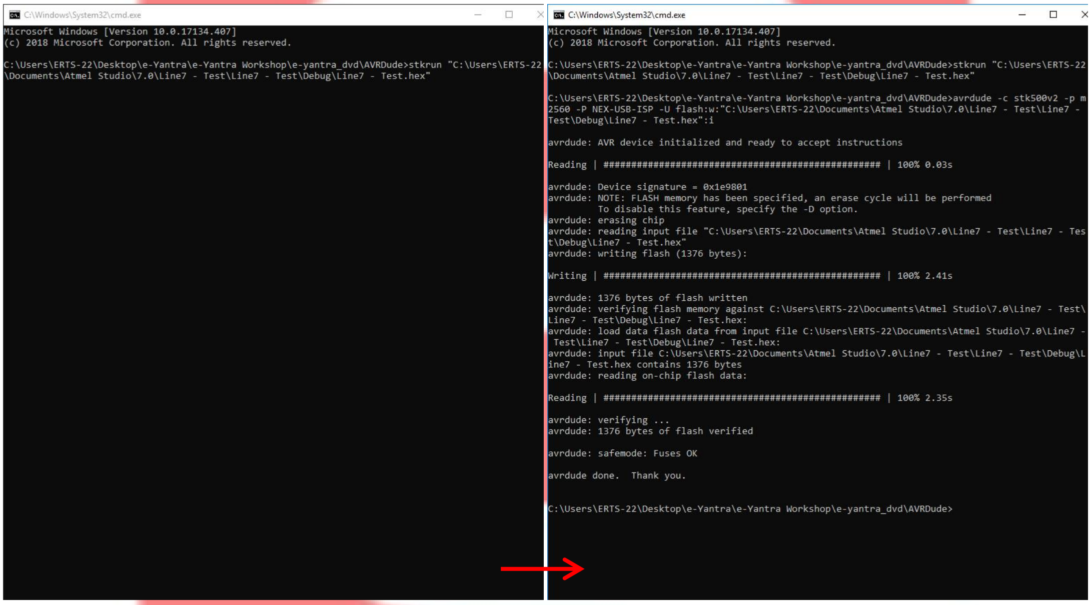


Figure 5: Connection b/w STK500 and ATMEGA2560

- Connect power supply to the jack of the microcontroller and switch on the microcontroller.
- Connect your USB cable to the STK500V2 and follow the instructions given below to burn hex file to ATmega 2560 using STK500V2.

Installing drivers for AVR programming:

- Download the AVRDude folder from this link.
- Use Atmel Studio to write the C program and build the solution to generate the .hex file. The .hex file is to be loaded on the microcontroller.
- Open the command prompt and change current directory to the AVRDude folder
- Type stkrun<space><address of filename.hex>. You can drag and drop the .hex file from the debug folder of your Atmel project into the command prompt window instead of typing the address of filename.hex.
- Execute the command (Make sure the board is connected to the computer and its power is on).



```
Microsoft Windows [Version 10.0.17134.407]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\ERTS-22\Desktop\e-Yantra\e-Yantra Workshop\e-yantra_dvd\AVRDude>stkrun "C:\Users\ERTS-22
\Documents\Atmel Studio\7.0\Line7 - Test\Line7 - Test\Debug\Line7 - Test.hex"
```

```
Microsoft Windows [Version 10.0.17134.407]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\ERTS-22\Desktop\e-Yantra\e-Yantra Workshop\e-yantra_dvd\AVRDude>stkrun "C:\Users\ERTS-22
\Documents\Atmel Studio\7.0\Line7 - Test\Line7 - Test\Debug\Line7 - Test.hex"

C:\Users\ERTS-22\Desktop\e-Yantra\e-Yantra Workshop\e-yantra_dvd\AVRDude>avrdude -c stk500v2 -p m
2560 -P NEX-USB-ISP -U flash:w:"C:\Users\ERTS-22\Documents\Atmel Studio\7.0\Line7 - Test\Line7 -
Test\Debug\Line7 - Test.hex":i

avrdude: AVR device initialized and ready to accept instructions

Reading | ################################################## | 100% 0.03s

avrdude: Device signature = 0x1e9801
avrdude: NOTE: FLASH memory has been specified, an erase cycle will be performed
         To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "C:\Users\ERTS-22\Documents\Atmel Studio\7.0\Line7 - Test\Line7 - Tes
t\Debug\Line7 - Test.hex"
avrdude: writing flash (1376 bytes):

Writing | ################################################## | 100% 2.41s

avrdude: 1376 bytes of flash written
avrdude: verifying flash memory against C:\Users\ERTS-22\Documents\Atmel Studio\7.0\Line7 - Test\
Line7 - Test\Debug\Line7 - Test.hex:
avrdude: load data flash data from input file C:\Users\ERTS-22\Documents\Atmel Studio\7.0\Line7 -
 Test\Line7 - Test\Debug\Line7 - Test.hex:
avrdude: input file C:\Users\ERTS-22\Documents\Atmel Studio\7.0\Line7 - Test\Line7 - Test\Debug\L
ine7 - Test.hex contains 1376 bytes
avrdude: reading on-chip flash data:

Reading | ################################################## | 100% 2.35s

avrdude: verifying ...
avrdude: 1376 bytes of flash verified

avrdude: safemode: Fuses OK

avrdude done.  Thank you.


C:\Users\ERTS-22\Desktop\e-Yantra\e-Yantra Workshop\e-yantra_dvd\AVRDude>
```

Figure 6: Burning Code