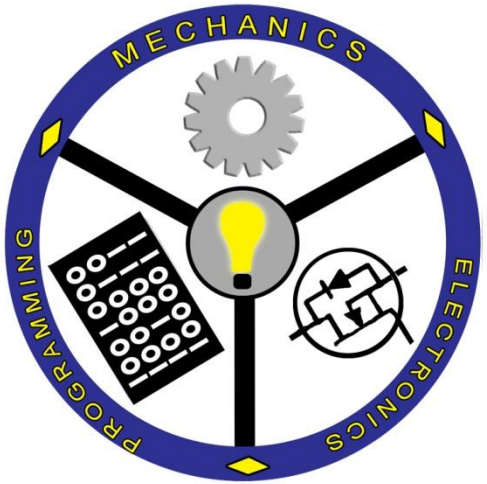


THE ROBOTICS CLUB

Integrating Knowledge...

Introduction to Arduino

The Robotics Club - SNIST

Compiled by Krishna Chaitanya

What is a microcontroller?

THE ROBOTICS CLUB
Integrating Knowledge...

- A microcontroller is a small computer on a single integrated circuit(IC) containing a processor core , memory, and programmable input/output peripherals.

- There are many types of MCs depending on their compatibility in terms of memory size and configurations.

Microcontrollers in robotics

THE ROBOTICS CLUB
Integrating Knowledge...

- In robotics ,MCs are mainly used to control robots automatically rather controlling them manually.
- A code(program) is fed into the microcontroller and this program has the instructions to control the robots.

What is arduino?

THE ROBOTICS CLUB
Integrating Knowledge...

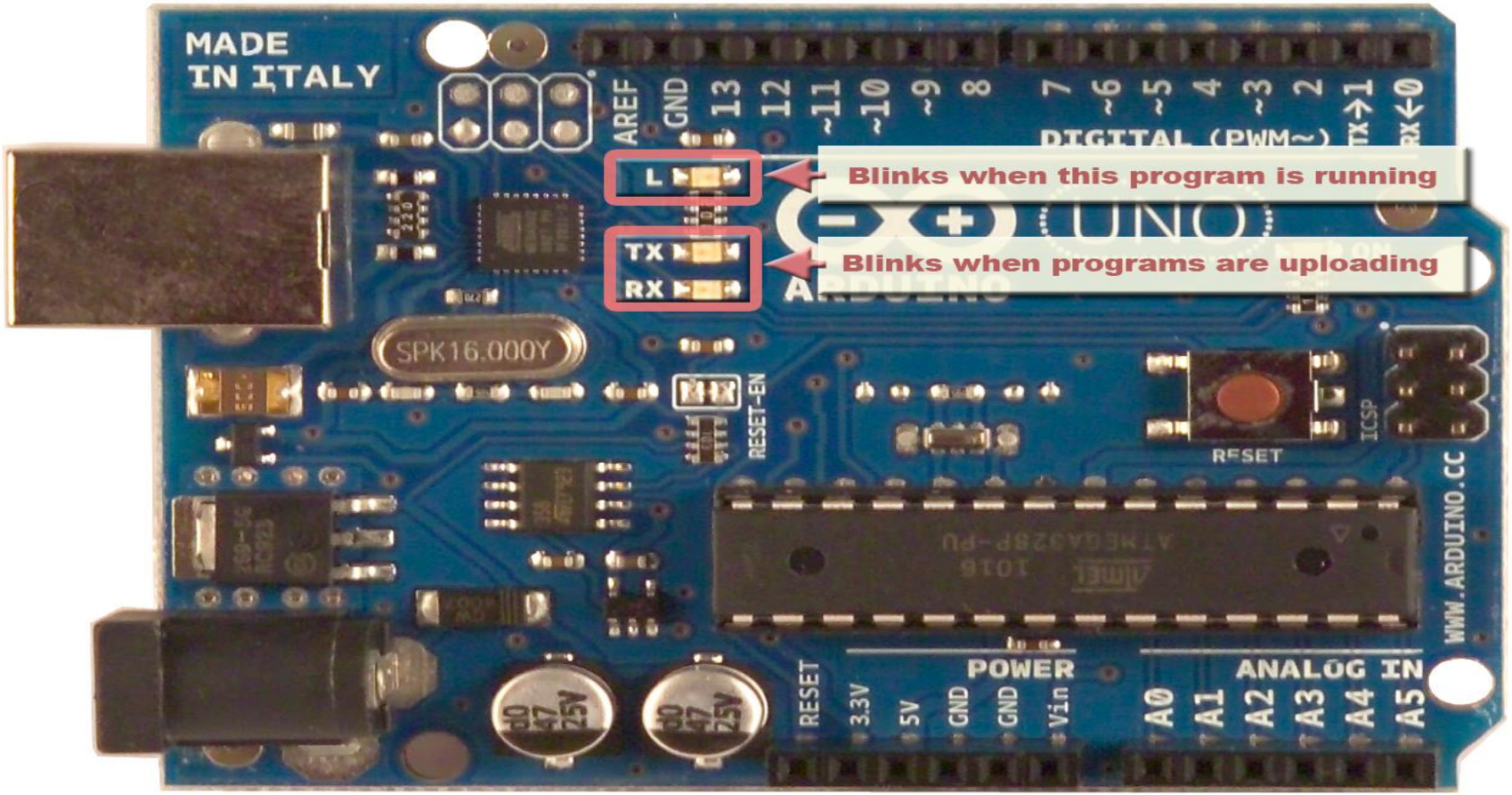
- Arduino is an open-source prototyping platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

Why should I use arduino?

THE ROBOTICS CLUB
Integrating Knowledge...

- **Cross-platform** - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
- **Simple, clear programming environment** - The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.
- **Open source and extensible software**
- **Open source and extensible hardware**

ARDUINO
@IntenseSchool


MADE IN ITALY
Blinks when this program is running
Blinks when programs are uploading
L
TX
RX
SPK16.000Y
RESET-EN
RESET
ICSP
POWER
ANALOG IN
WWW.ARDUINO.CC

Power

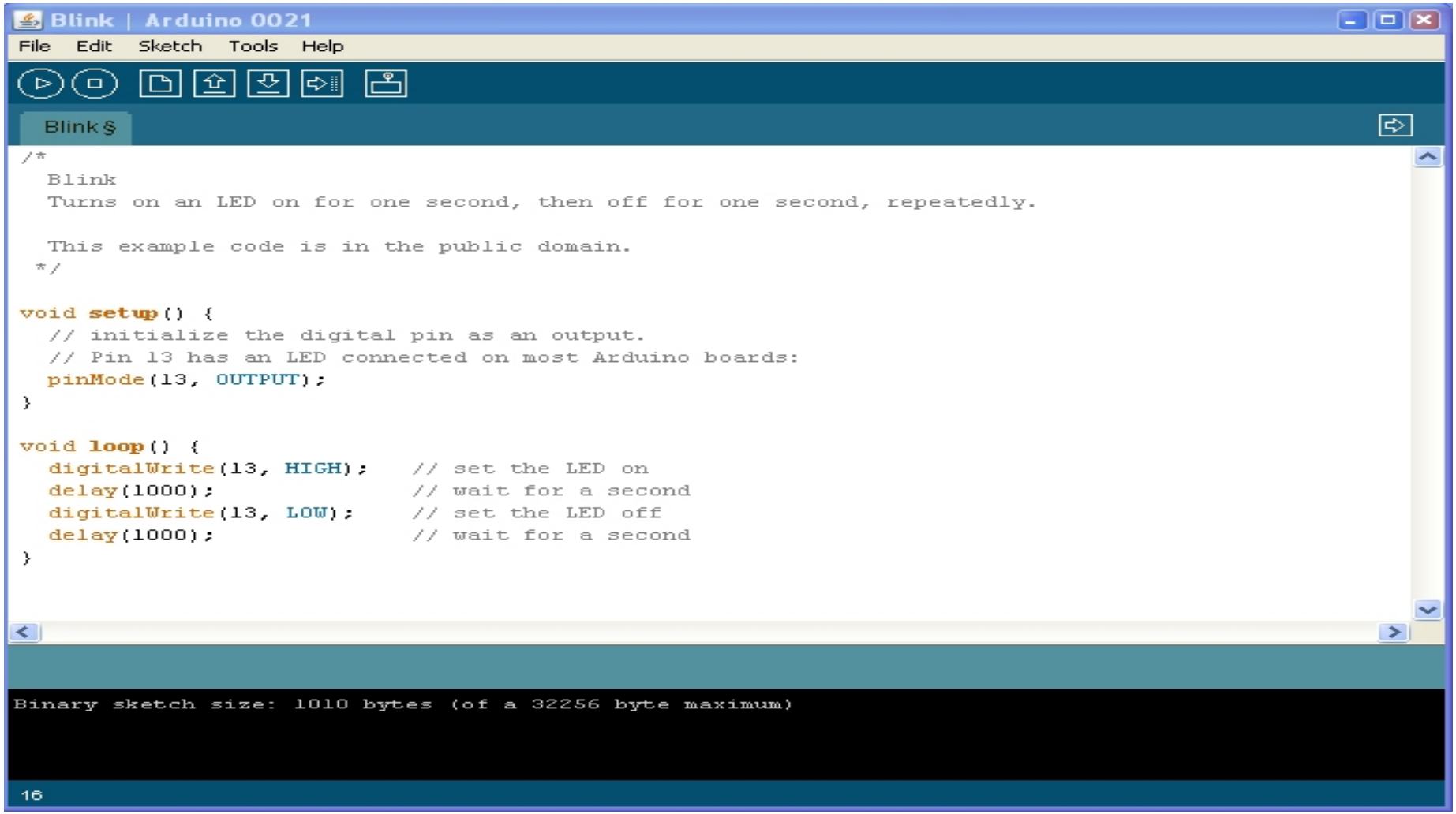
THE ROBOTICS CLUB
Integrating Knowledge...

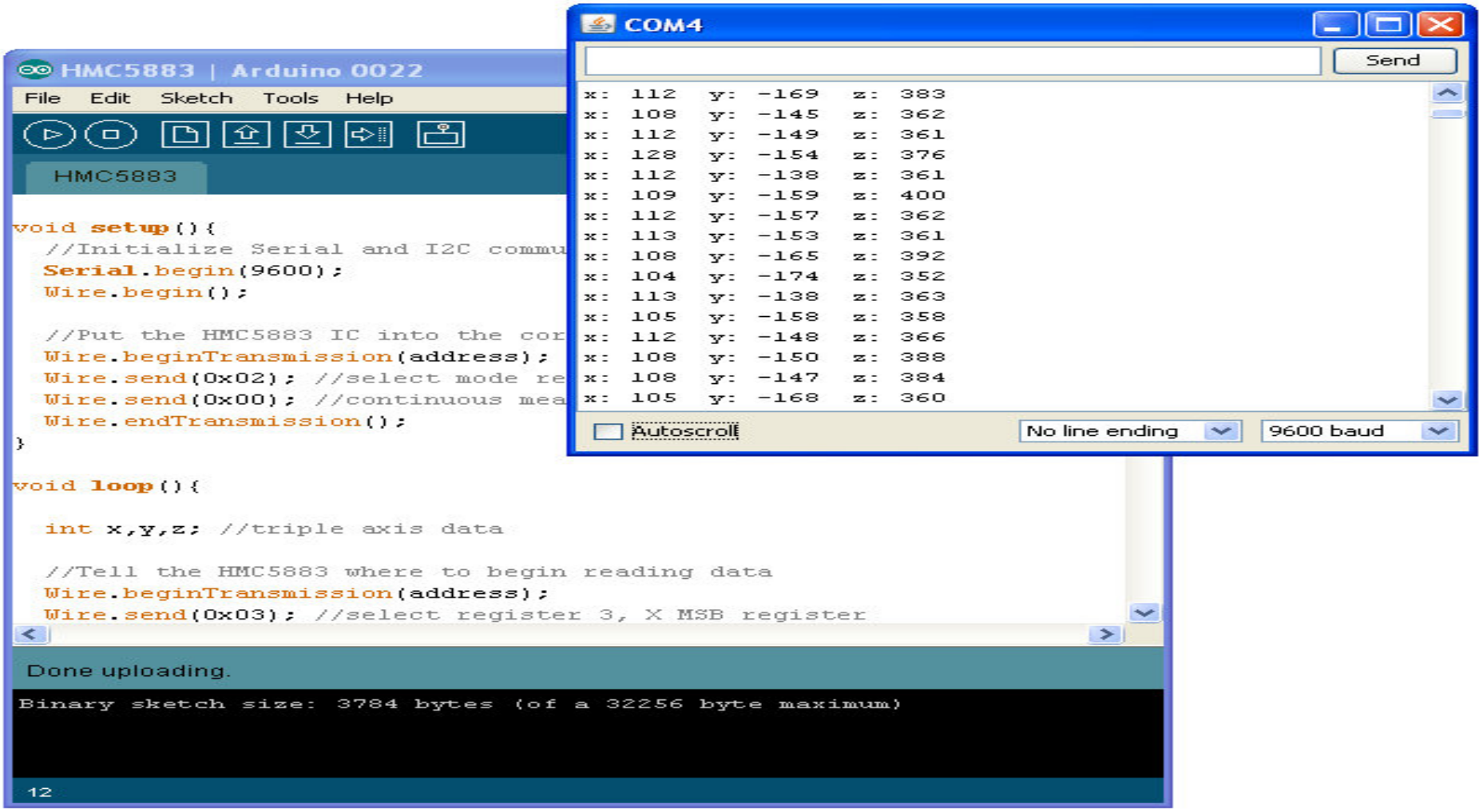
- The Uno board can be powered via the USB connection or with an external power supply. The power source is selected automatically.
- External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the GND and Vin pin headers of the POWER connector.
- The board can operate on an external supply from 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may become unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

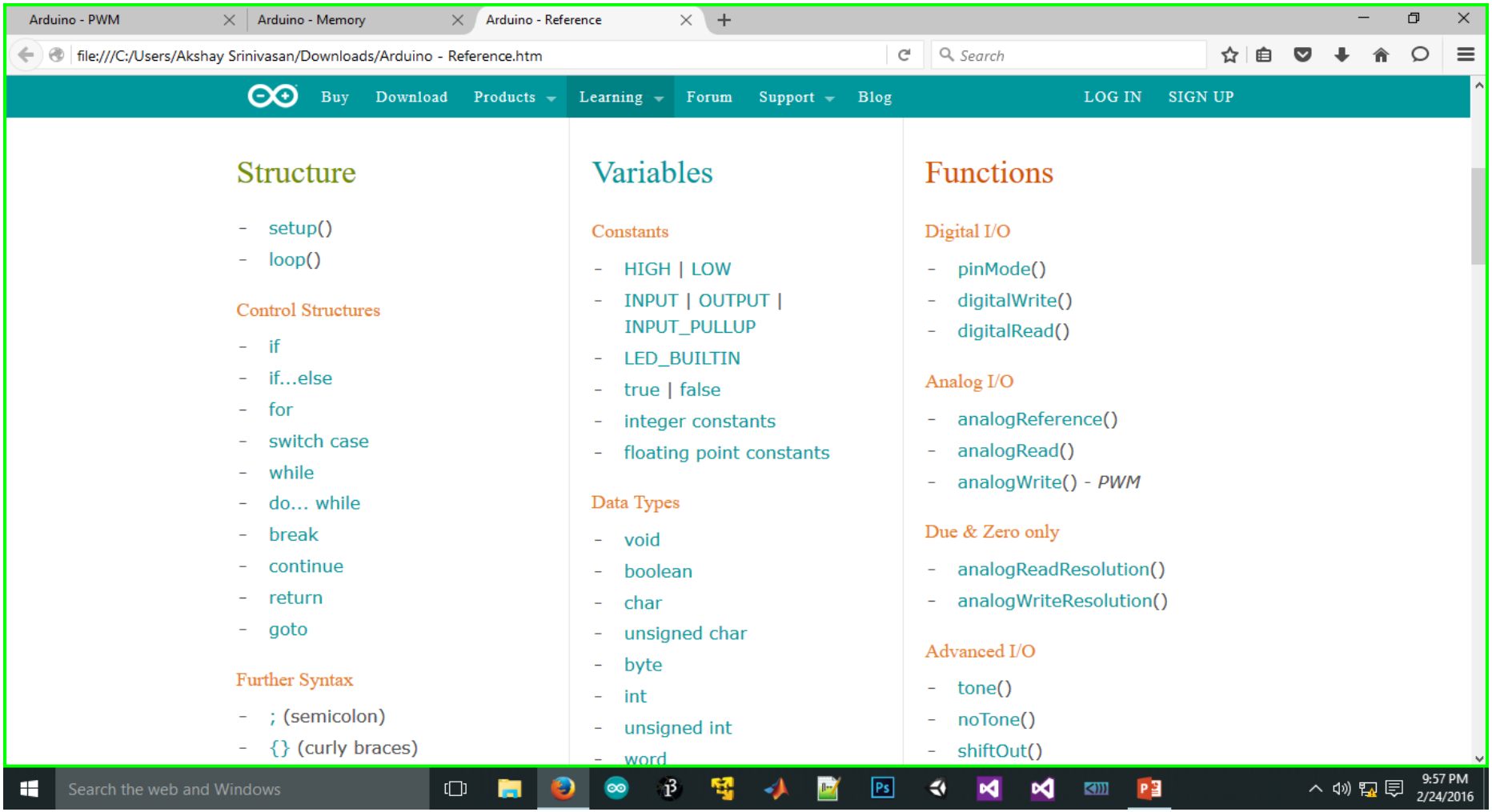
- Vin. The input voltage to the Uno board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- 5V.This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- GND. Ground pins.
- IOREF. This pin on the Uno board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5V or 3.3V.

Arduino IDE

THE ROBOTICS CLUB
Integrating Knowledge...




HMC5883 | Arduino 0022
File Edit Sketch Tools Help
HMC5883
void setup(){
//Initialize Serial and I2C commu
Serial.begin(9600);
Wire.begin();
//Put the HMC5883 IC into the cor
Wire.beginTransmission(address);
Wire.send(0x02); //select mode re
Wire.send(0x00); //continuous mea
Wire.endTransmission();
}
void loop(){
int x,y,z; //triple axis data
//Tell the HMC5883 where to begin reading data
Wire.beginTransmission(address);
Wire.send(0x03); //select register 3, X MSB register
Done uploading.
Binary sketch size: 3784 bytes (of a 32256 byte maximum)
12
COM4
Send
x: 112 y: -169 z: 383
x: 108 y: -145 z: 362
x: 112 y: -149 z: 361
x: 128 y: -154 z: 376
x: 112 y: -138 z: 361
x: 109 y: -159 z: 400
x: 112 y: -157 z: 362
x: 113 y: -153 z: 361
x: 108 y: -165 z: 392
x: 104 y: -174 z: 352
x: 113 y: -138 z: 363
x: 105 y: -158 z: 358
x: 112 y: -148 z: 366
x: 108 y: -150 z: 388
x: 108 y: -147 z: 384
x: 105 y: -168 z: 360
Autoscroll
No line ending
9600 baud


Arduino - PWM
Arduino - Memory
Arduino - Reference
file:///C:/Users/Akshay Srinivasan/Downloads/Arduino - Reference.htm
Search
Buy Download Products Learning Forum Support Blog LOG IN SIGN UP
Structure
setup()
loop()
Control Structures
if
if...else
for
switch case
while
do... while
break
continue
return
goto
Further Syntax
; (semicolon)
{} (curly braces)
Variables
Constants
HIGH | LOW
INPUT | OUTPUT | INPUT_PULLUP
LED_BUILTIN
true | false
integer constants
floating point constants
Data Types
void
boolean
char
unsigned char
byte
int
unsigned int
word
Functions
Digital I/O
pinMode()
digitalWrite()
digitalRead()
Analog I/O
analogReference()
analogRead()
analogWrite() - PWM
Due & Zero only
analogReadResolution()
analogWriteResolution()
Advanced I/O
tone()
noTone()
shiftOut()
Search the web and Windows
9:57 PM
2/24/2016

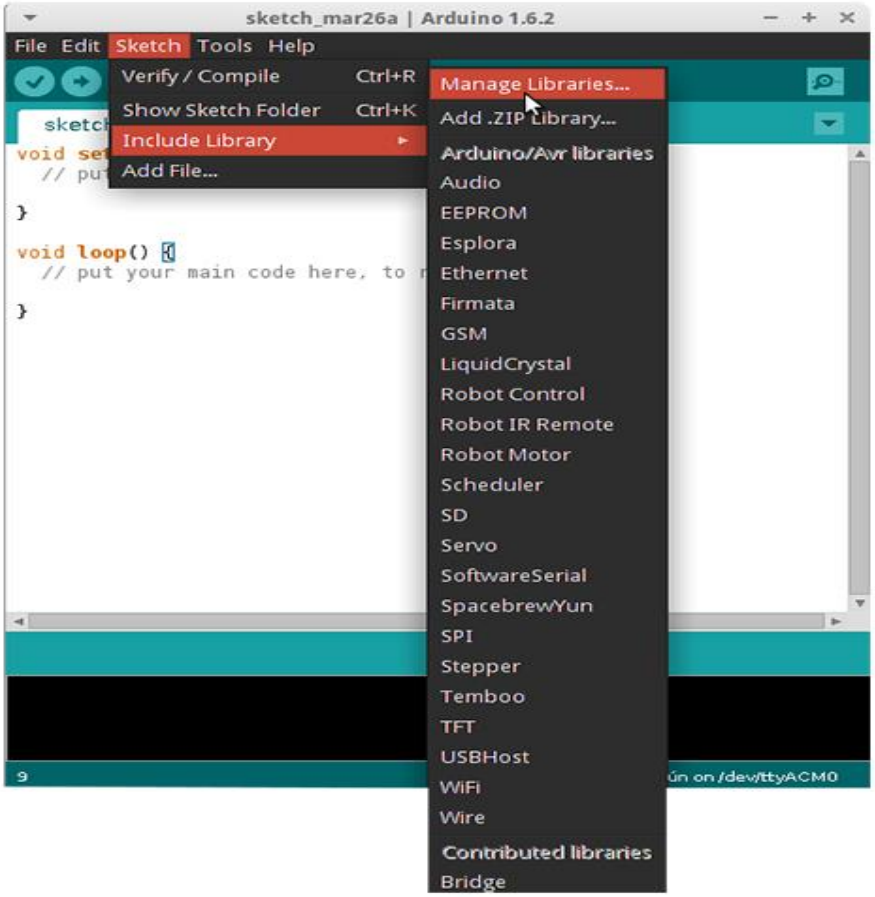
LIBRARIES

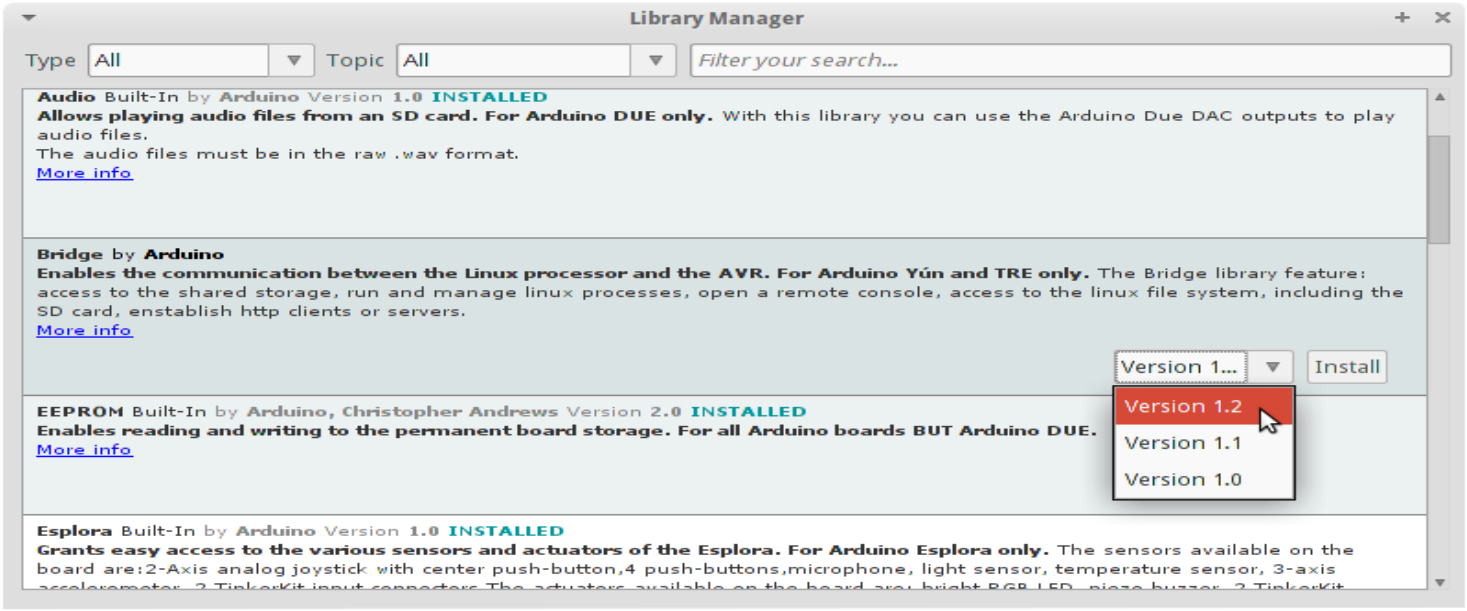
- Libraries are a collection of code that makes it easy for you to connect to a sensor, display, module, etc. For example, the built-in LiquidCrystal library makes it easy to talk to character LCD displays. There are hundreds of additional libraries available on the Internet for download. The built-in libraries and some of these additional libraries are listed in the reference. To use the additional libraries, you will need to install them.

How to Install a Library?

THE ROBOTICS CLUB
Integrating Knowledge...

- To install a new library into your Arduino IDE you can use the Library Manager (available from IDE version 1.6.2). Open the IDE and click to the "Sketch" menu and then *Include Library > Manage Libraries*


sketch_mar26a | Arduino 1.6.2
File Edit Sketch Tools Help
Verify / Compile Ctrl+R
Show Sketch Folder Ctrl+K
Include Library
Add File...
Manage Libraries...
Add .ZIP Library...
Arduino/Avr libraries
Audio
EEPROM
Esplora
Ethernet
Firmata
GSM
LiquidCrystal
Robot Control
Robot IR Remote
Robot Motor
Scheduler
SD
Servo
SoftwareSerial
SpacebrewYun
SPI
Stepper
Temboo
TFT
USBHost
WiFi
Wire
Contributed libraries
Bridge
void loop() {
// put your main code here, to
}


Library Manager
Type All
Topic All
Filter your search...
Audio Built-In by Arduino Version 1.0 INSTALLED
Allows playing audio files from an SD card. For Arduino DUE only. With this library you can use the Arduino Due DAC outputs to play audio files.
The audio files must be in the raw .wav format.
More info
Bridge by Arduino
Enables the communication between the Linux processor and the AVR. For Arduino Yún and TRE only. The Bridge library feature: access to the shared storage, run and manage linux processes, open a remote console, access to the linux file system, including the SD card, enstablish http clients or servers.
More info
Version 1...
Install
Version 1.2
Version 1.1
Version 1.0
EEPROM Built-In by Arduino, Christopher Andrews Version 2.0 INSTALLED
Enables reading and writing to the permanent board storage. For all Arduino boards BUT Arduino DUE.
More info
Esplora Built-In by Arduino Version 1.0 INSTALLED
Grants easy access to the various sensors and actuators of the Esplora. For Arduino Esplora only. The sensors available on the board are:2-Axis analog joystick with center push-button,4 push-buttons,microphone, light sensor, temperature sensor, 3-axis

Analog pins

The Atmega controllers used for the Arduino contain an onboard 6 channel analog-to-digital (A/D) converter. The converter has 10 bit resolution, returning integers from 0 to 1023. While the main function of the analog pins for most Arduino users is to read analog sensors, the analog pins also have all the functionality of general purpose input/output (GPIO) pins (the same as digital pins 0 - 13).

MEMORY

THE ROBOTICS CLUB
Integrating Knowledge...

There are three pools of memory in the microcontroller used on avr-based Arduino boards :

- Flash memory (program space), is where the Arduino sketch is stored.
- SRAM (static random access memory) is where the sketch creates and manipulates variables when it runs.
- EEPROM is memory space that programmers can use to store long-term information.

Flash memory and EEPROM memory are non-volatile (the information persists after the power is turned off). SRAM is volatile and will be lost when the power is cycled.

- The ATmega328 chip found on the Uno has the following amounts of memory:

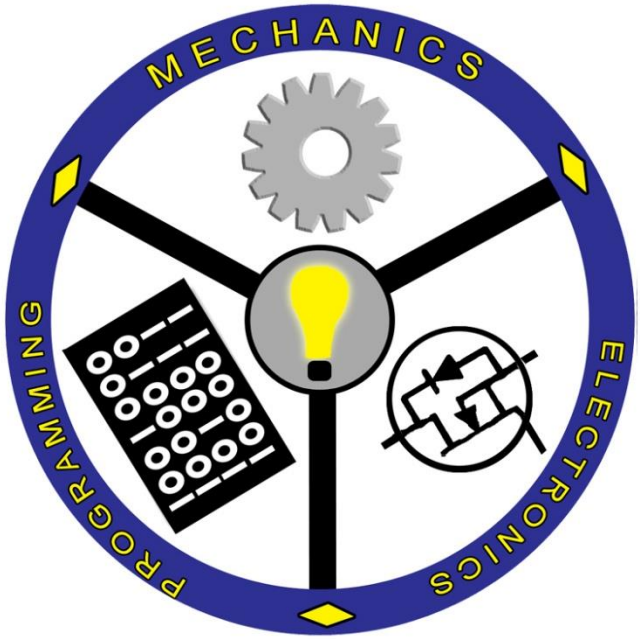
Flash 32k bytes (of which .5k is used for the bootloader) SRAM 2k bytes EEPROM 1k byte

- The ATmega2560 in the Mega2560 has larger memory space :

Flash 256k bytes (of which 8k is used for the bootloader) SRAM 8k bytes EEPROM 4k byte

Basic functions for programming

- void setup(),void loop()
- pinMode(pin number,OUTPUT/INPUT)
- digitalRead(pin)
- digitalWrite(pin,value)//HIGH or LOW
- analogRead(pin)
- analogWrite(pin,value)//0-255 or voltage value
- Serial.begin(9600);
- Serial.println(“……”);



THE ROBOTICS CLUB

Integrating Knowledge...

Compiled by Krishna Chaitanya