Swayam Arduino

Components of Arduino.

~~(1) GPIO pin(s) (General purpose Input/output pins) are~~

(1) Microcontroller (ATMEGA-328P)
↳ RAM/ROM
↳ Bootloader :- 1st program executes when the board is supplied with power

(2) GPIO pin(s) or Digital Pins.
(General perpose Input/output pins) are programmable pins which can be programmed for digital input or output.

(3) Analog pins. (Input only)
connected with ADC. (Analog to digital Converter) it takes analog signals (0 to 5V approx) and converts it to values (0 to 1023) [Arduino uno has 10-bit ADC (i.e → 2^{10}=1024 0 to 1023)].

(4) On-Board LED(s)
① Power LED
② TX-RX LED(s)
↳ They glows when the board is communicating with its standard serial port.
③ LED (Connected to D13(Arduino uno))

(5) Reset Button.
when pressed, resturts the Board.

(6) Usb interface.
to upload the code on Board or to communicate it with another device (with its standerd serial port).

⑦ Power Jack

for Powering the Board.

Arduino Programming

```
void setup() { }
  [code, to run once]

void loop() { }
  [code, to run the code repetedly]
```

Some Standard functions.

1) pinMode (<pin>, <mode>);

→ Configures the ~~specificated~~ specified pin to behave either as an input or an output.

Parameters:-

<Pin>: the Arduino pin to be configured.

<mode>: INPUT, OUTPUT or INPUT_PULLUP

For activating Internal Pullup.

2) digitalWrite (<Pin>, <value>);

→ Write a [HIGH] or a [LOW] value to a digital pin.

Note If the pin Configured as INPUT, then this will controll its internal pullup.

Recommended:- use this only for digital pins.

A1 A6 A

3) digitalRead (<Pin>);
Returns the digital value of the specified pin.

Parameter :-
<Pin> the arduino pin whose digital value is to be readed

4) analogRead (<Pin>);

Reads value (voltage, 0 to operating voltage of the board) then converts it to integer values and returns it.

Arduino boards contain multichannel ADCs.
These ADCs can be 8bit, 10bit, 12bit or 14bit or more.
Therefore values that ADC maps for input voltages are
8-bit → [0, 255] ($\because 2^8 = 256$)
10-bit → [0, 1023] ($\because 2^{10} = 1024$), etc.

if the ADC is 10-bit then its "Resolution" is 10-bit.

5) analogWrite (<Pin>, <value>);
Writes an analog value (PWM wave) to a pin. Can be used to light a LED at varying brightness or drive a motor @ various speeds.
Note: only be used with a PWM supported Pin.

for more info., please refer PWM section.

6) pulseIn(<pin>, <Value>);
used to record digital pulse widths.

Reads a pulse (either 1 or 0) on a pin.
For example, if value is 1, pulseIn() waits for pin to go LOW, then starts recording time (in ms) till the pin is HIGH, then stops recording and returns the length of pulse in ms. or gives up and returns 0 if no complete pulse was received within the timeout.

Works on pulses from 10ms to 3ms.

Syntax:-

unsigned long <variable> = pulseIn(<pin>,<value>,<timeout>);

<timeout> is optional.

Serial functions (UART)
(this functions is used to communicate with the board through its standerd serial port).

Serial.begin(<speed>);
used to start communication b/w the board through its std. serial port (UART)

<speed> is bound rate or data rate (bits/sec).

Serial.available();
checks if bytes (characters) are available to read. returns 1 if true & 0 if false.

Serial.print();
used to print anything (char, int, float, etc) on serial port (std.).

Serial.println();
used to print (newline) anything on serial port.

Serial.write();
used to write anything to serial port.

Serial.read();
used to read anything from serial port.

Serial.readString();
used to read string from serial port.

Serial.readStringUntil ("char");
used to read string untill the "char" passed to it as arg.

LCD with Arduino.

→ for ~~ind~~ including :-

```
#include <LiquidCrystal.h>
```

→ LiquidCrystal (Pins definition)

Syntex:-

```
LiquidCrystal lcd (rs, enable, d4, d5, d6, d7);
```

→ .begin

```
lcd.begin (cols, rows, charsize)
```

Example.

```
lcd.begin (16, 2);
```

→ .setCursor

Positions the LCD cursor.

Syntex:-

```
lcd.setCursor (col, row);
```

→ .print

Prints the data on LCD

Syntex:-

```
lcd.print (data);
```

↓

data to print.

→ scrollDisplayLeft or scrollDisplayRight :-

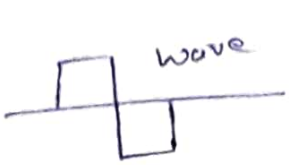
Scrolls the contents of display (text and cursor) one space left or right

Syntex :-
lcd.scrollDisplayLeft();

→ autoscroll()
Turns on automatic scrolling of the LCD. This causes each character output to the display to ~~pu~~ push previous characters over by one space (left to right).
*
Syntex :-
lcd.autoscroll();

to turn it off :-

~~lcd.autoscroll~~
lcd.noautoscroll();

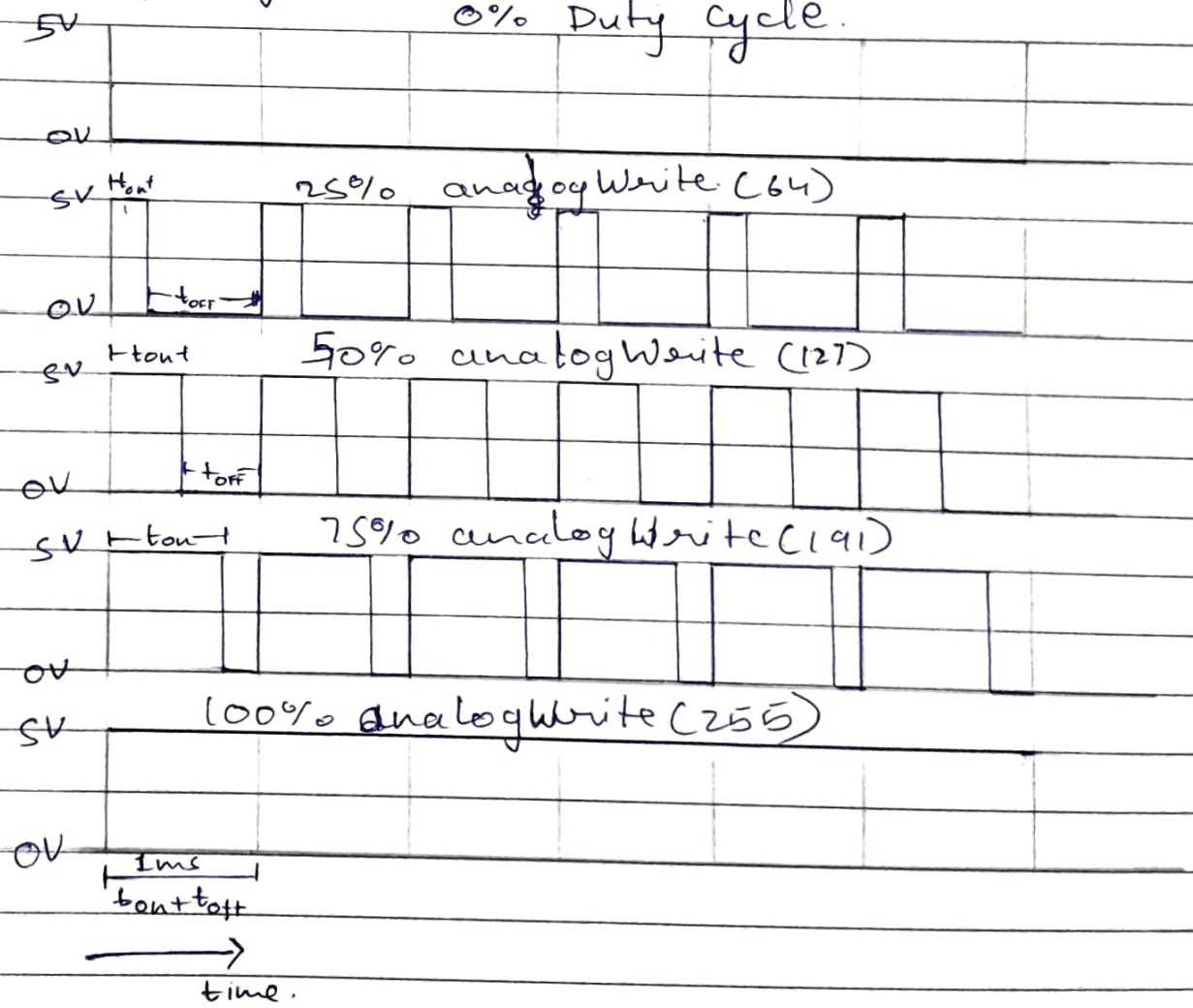


PWM :-

Pulse Width Modulation :-

- PWM signal is a high frequency square wave signal i.e. (1KHz)
- PWM is a technique by which the width of the pulse is varied while keeping the frequency constant.

- PWM signal consists of two main properties that defines its behavior.
  1) duty cycle
  2) frequency.



$$\text{duty cycle} = \frac{t_{on}}{t_{on} + t_{OFF}}$$

t_{on} = on time. (in ms)
t_{off} = off time. (in ms)
$t_{on} + t_{off}$ = Time period. (in ms)

* Frequency determines how fast the PWM completes a cycle. (i.e how fast it switches from HIGH to LOW and vice versa)

$$f = \frac{1}{\text{Time period}} = \frac{1}{t_{on} + t_{off}}$$

$t_{on} + t_{OFF} = 1ms$

$f = 1KHz$ (Standard) $\therefore$ time period $= \frac{1}{1000}$ Sec.

$= 1ms$

denotion :-
pins marked with ~ are PWM supported pins.

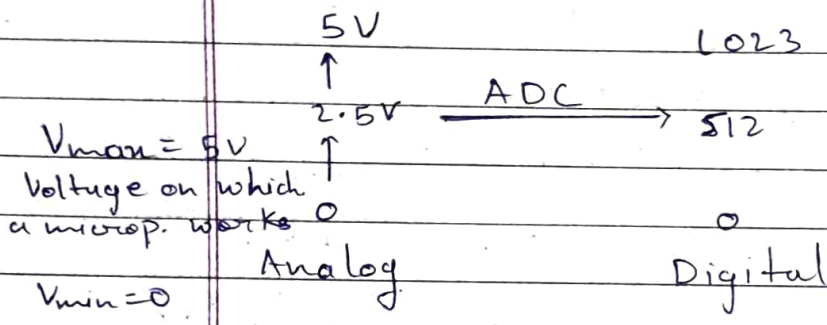
ADC

Analog to Digital Converter.

topics to be covered.

* ADC pins
* ADC Resolution.
* ~~DAC Resolution (Example)~~.

Arduino uno has 10bit Resolution it means it can convert a certain voltage range (0 to 5V) to digital range (0 to 2^{10} i.e 1024).



V_{max} = 5V
Voltage on which a microp. works

V_{min} = 0

$$\text{Resolution (min. Input voltage)} = \frac{V_{max}}{1024} = 4.9\,mV$$

Can read 10,000 times / sec.

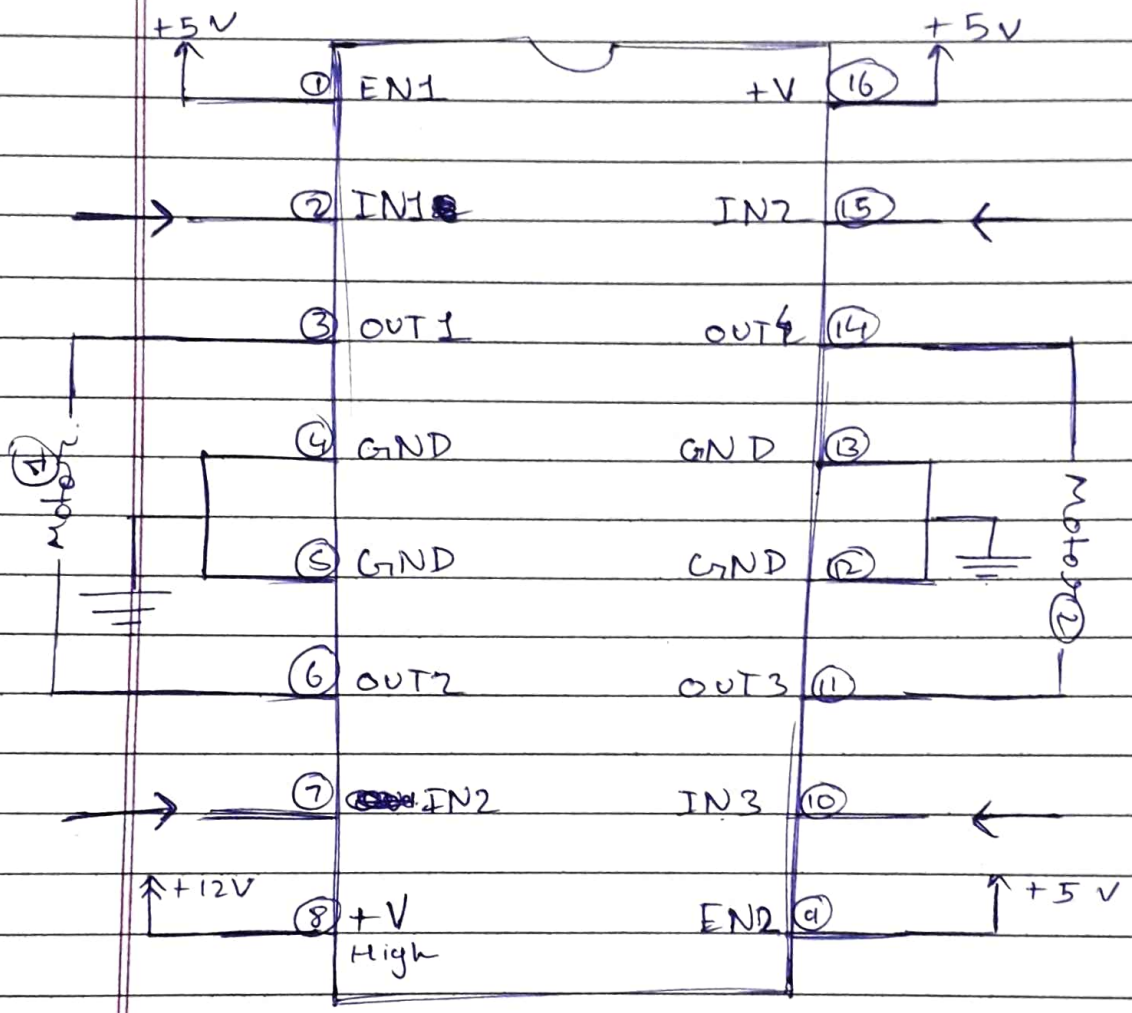
H-Bridge :-

* Consists of 4 MOSFETs in 'H' shape
* They four collectively controls the functionality of the device.

Some Other devices with arduino :-

L293D. (Motor Driver IC) (Dual H-Bridge)

Controlls motors on higher voltages (upto 12V) receiving low volt signals from a microC. ie (0 to 5V).



L298N Module. (Dual ~~H~~ Bi H-Bridge)

Module with IC L298N which is an upgrade model of ~~L293D~~ L293D works similarly but with less heat generation and improved efficiency with onboard 7805

Date
Page

Communication Protocalls.

* UART (Universal Asynchronous Rx Tx)
* I^2C (Inter - Integrated Circuit).
* SPI (Serial - Peripheral Interface)
* GPIO (General Perpose Input/output)
  ↳ (covered b4)