UNIT II: MICROCONTROLLER

Introduction to microcontrollers and microprocessors, Different microcontrollers, Arduino: Types, UNO Architecture, ADC, DAC, Data acquisition

---

2.3 Arduino: Types

What are the Types of Arduino Boards?

Arduino board is an open-source platform used to make electronics projects. It consists of both a microcontroller and a part of the software or Integrated Development Environment (IDE) that runs on your PC, used to write & upload computer code to the physical board. The platform of an Arduino has become very famous with designers or students just starting out with electronics, and for an excellent cause.

Fig: Types of Arduino Boards

Unlike most earlier programmable circuit boards, the Arduino does not require a separate part of hardware in order to program a new code onto the board you can just use a USB cable. As well, the Arduino IDE uses a basic version of C++, making it simpler to learn the program. At last, the Arduino board offers a typical form factor that breaks out the functions of the microcontroller into a more available package.

Why Arduino Boards?

Arduino board has been used for making different engineering projects and different applications. The Arduino software is very simple to use for beginners, yet flexible adequate for advanced users. It runs Windows, Linux, and Mac. Teachers and students in the schools utilize it to design low-cost scientific instruments to verify the principles of physics and chemistry. There are numerous other microcontroller platforms obtainable for physical computing. The Netmedia's BX-24, Parallax Basic Stamp, MIT's Handyboard, Phidget, and many others present related functionality. Arduino also makes simpler the working process of microcontroller, but it gives some advantages over other systems for teachers, students, and beginners.

- Inexpensive
- Cross-platform
- The simple, clear programming environment
- Open source and extensible software

- Open source and extensible hardware

The Function of the Arduino Board

The flexibility of the Arduino board is enormous so that one can do anything they imagine. This board can be connected very easily to different modules such as obstacle sensors, presence detectors, fire sensors, GSM Modules GPS modules, etc. The main function of the Arduino board is to control electronics through reading inputs & changing it into outputs because this board works like a tool. This board is also used to make different electronics projects in the field of electronics, electrical, robotics, etc.

Features of Different Types of Arduino Boards

The features of different types of Arduino boards are listed in the tabular form.

Arduino Board	Processor	Memory	Digital I/O	Analogue I/O
Arduino Uno	16Mhz ATmega328	2KB SRAM, 32KB flash	14	6 input, 0 output
Arduino Due	84MHz AT91SAM3X8E	96KB SRAM, 512KB flash	54	12 input, 2 output
Arduino Mega	16MHz ATmega2560	8KB SRAM, 256KB flash	54	16 input, 0 output
Arduino Leonardo	16MHz ATmega32u4	2.5KB SRAM, 32KB flash	20	12 input, 0 output

Different Types Of Arduino Boards

The list of Arduino boards includes the following such as

- Arduino Uno (R3)
- Arduino Nano
- Arduino Micro
- Arduino Due
- LilyPad Arduino Board
- Arduino Bluetooth
- Arduino Diecimila
- RedBoard Arduino Board
- Arduino Mega (R3) Board
- Arduino Leonardo Board
- Arduino Robot
- Arduino Esplora
- Arduino Pro Mic
- Arduino Ethernet
- Arduino Zero
- Fastest Arduino Board

Arduino Uno (R3)

The Uno is a huge option for your initial Arduino. This Arduino board depends on an ATmega328P based microcontroller. As compared with other types of arduino boards, it is very simple to use like the Arduino Mega type board. .It consists of 14-digital I/O pins, where 6-pins can be used as PWM(pulse width modulation outputs), 6-analog inputs, a reset button, a power jack, a USB connection, an In-Circuit Serial Programming header (ICSP), etc. It includes everything required to hold up the microcontroller; simply attach it to a PC with the help of a USB cable and give the supply to get started with an AC-to-DC adapter or battery.

Arduino Uno (R3)

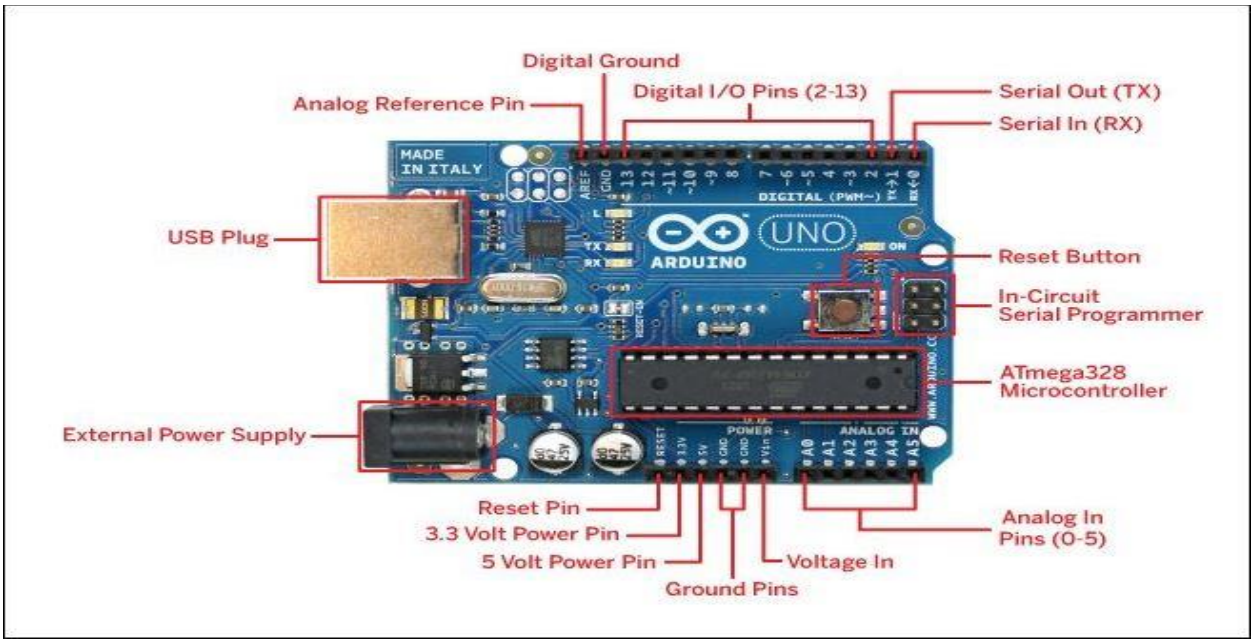
Arduino Uno is the most frequently used board and it is the standard form apart from all the existing Arduino Boards. This board is very useful for beginners.

2.4 UNO Architecture

Arduino Uno (R3)

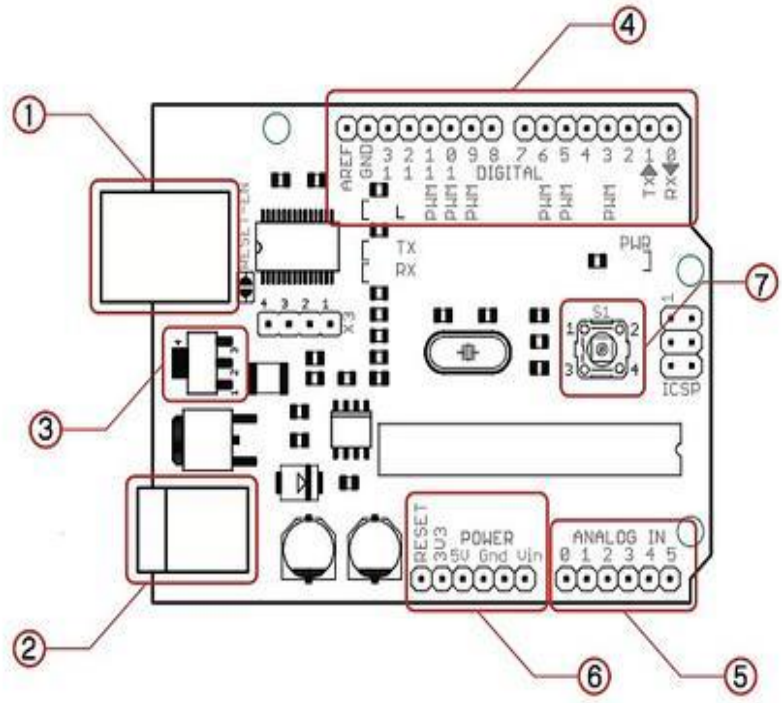
- The Uno is a huge option for your initial Arduino.
- This Arduino board depends on an ATmega328P based microcontroller.
- As compared with other types of Arduino boards, it is very simple to use like the Arduino Mega type board.
- It consists of 14-digital I/O pins, where 6-pins can be used as PWM (pulse width modulation outputs), 6-analog inputs, a reset button, a power jack, a USB connection, an In-Circuit Serial Programming header (ICSP), etc.
- It includes everything required to hold up the microcontroller; simply attach it to a PC with the help of a USB cable and give the supply to get started with an AC-to-DC adapter or battery.

Fig: Pin diagram



UNO Architecture:

- Arduino's processor basically uses the Harvard architecture where the program code and program data have separate memory.
- It consists of two memories- Program memory and the data memory.
- The code is stored in the flash program memory, whereas the data is stored in the data memory.
- The Atmega328 has 32 KB of flash memory for storing code (of which 0.5 KB is used for the bootloader), 2 KB of SRAM and 1 KB of EEPROM and operates with a clock speed of 16MHz.

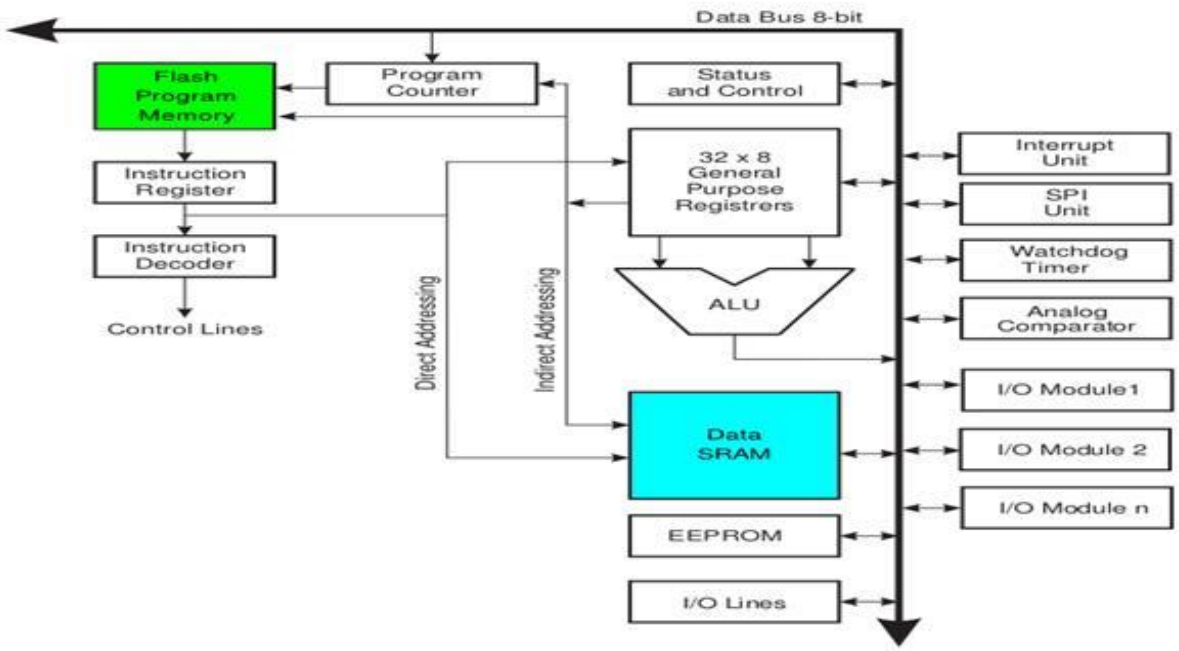


The most important parts on the Arduino board high lighted in red:

1: USB connector
2: Power connector
3: Automatic power switch
4: Digital pins
5: Analog pins
6: Power pins
7: Reset switch

UNO Pin details:

Pin Category	Pin Name	Details
Power	Vin, 3.3V, 5V, GND	Vin: Input voltage to Arduino when using an external power source. 5V: Regulated power supply used to power microcontroller and other components on the board. 3.3V: 3.3V supply generated by on-board voltage regulator. Maximum current draw is 50mA. GND: ground pins.
Reset	Reset	Resets the microcontroller.
Analog Pins	A0 – A5	Used to provide analog input in the range of 0-5V
Input/Output Pins	Digital Pins 0 - 13	Can be used as input or output pins.
Serial	0(Rx), 1(Tx)	Used to receive and transmit TTL serial data.
External Interrupts	2, 3	To trigger an interrupt.
PWM	3, 5, 6, 9, 11	Provides 8-bit PWM output.
SPI	10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK)	Used for SPI communication.
Inbuilt LED	13	To turn on the inbuilt LED.
TWI	A4 (SDA), A5 (SCA)	Used for TWI communication.
AREF	AREF	To provide reference voltage for input voltage.



- The 14 digital input/output pins can be used as input or output pins by using pinMode(), digitalRead() and digitalWrite() functions in arduino programming. Each pin operate at 5V and can provide or receive a maximum of 40mA current, and has an internal pull-up resistor of 20-50 KOhms which are disconnected by default.
- Out of these 14 pins, some pins have specific functions as listed below:
- **Serial Pins 0 (Rx) and 1 (Tx):** Rx and Tx pins are used to receive and transmit TTL serial data.
- They are connected with the corresponding ATmega328P USB to TTL serial chip.
- **External Interrupt Pins 2 and 3:** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
- **PWM Pins 3, 5, 6, 9 and 11:** These pins provide an 8-bit PWM output by using analogWrite() function.
- **SPI Pins 10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK):** These pins are used for SPI communication.
- **In-built LED Pin 13:** This pin is connected with an built-in LED, when pin 13 is HIGH – LED is on and when pin 13 is LOW, its off.
- Along with 14 Digital pins, there are 6 analog input pins, each of which provide 10 bits of resolution, i.e. 1024 different values. They measure from 0 to 5 volts but this limit can be increased by using AREF pin with analog Reference() function.
- Analog pin 4 (SDA) and pin 5 (SCA) also used for TWI communication using Wire library.
- Arduino Uno has a couple of other pins as explained below:
- **AREF:** Used to provide reference voltage for analog inputs with analogReference() function.
- **Reset Pin:** Making this pin LOW, resets the microcontroller.

The basic working of CPU of ATmega328:-

- The data is uploaded in serial via the port (being uploaded from the computer's Arduino IDE). The data is decoded and then the instructions are sent to instruction register and it decodes the instructions on the same clock pulse.
- On the next clock pulse the next set of instructions are loaded in instruction register.
- In general purpose registers the registers are of 8-bit but there are 3 16-bit registers also.
- 8-bit registers are used to store data for normal calculations and results.
- 16-bit registers are used to store data of timer counter in 2 different register. E.g., X-low & X-high. They are fast and are used to store specific hardware functions.
- EEPROM stores data permanently even if the power is cut out. Programming inside a EEPROM is slow.
- Interrupt Unit checks whether there is an interrupt for the execution of instruction to be executed in ISR (Interrupt Service Routine).
- Serial Peripheral Interface (SPI) is an interface bus commonly used to send data between microcontrollers and small peripherals such as Camera, Display, SD cards, etc. It uses separate clock and data lines, along with a select line to choose the device you wish to talk to.
- Watchdog timer is used to detect and recover from MCU malfunctioning.
- Analog comparator compares the input values on the positive and negative pin, when the value of positive pin is higher the output is set.
- Status and control is used to control the flow of execution of commands by checking other blocks inside the CPU at regular intervals.
- ALU (Arithmetic and Logical unit)The high performance AVR ALU operates in direct connection with all the 32 general purpose working registers. Within a single clock cycle, arithmetic operations b/w general purpose registers are executed. The ALU operations are divided into 3 main categories – arithmetic, logical and bit-function.
- I/O pins The digital inputs and outputs (digital I/O) on the Arduino are what allow you to connect the Arduino sensors, actuators, and other ICs. Learning how to use them will allow you to use the Arduino to do some useful things, such as reading switch inputs, lighting indicators, and controlling relay outputs.