Prepared By Teran Subasinghe

MICROCONTROLLERS

WHAT IS A MICRO CONTROLLER?

- A micro-controller is a simple computer that can (generally) run one program at a time and very simple to use.
- A micro-controller is a single chip that contains a CPU, memory and programmable input/output peripherals.
- Arduino Uno is based on an 8-bit AVR micro-controller. AVR is a family of 8-bit micro-controllers developed by Atmel (now owned by Microchip). ATmega is one family of AVR micro-controllers. For example, the Arduino Uno uses an ATmega328P micro-controller.
- Other Arduino boards use different micro-controllers, some of them a lot more powerful than the one used in the Arduino Uno.
- A Raspberry Pi is a low power and small size full computer. It is a general purpose computer.
- Raspberry Pi is neither a microprocessor or micro-controller , it is a single board computer which contains a SOC (System On Chip - Has multicore processor,GPU,ROM,I/O Peripherals inside it.),DDR RAM memory, Ethernet port, USB host, micro HDMI on it.
- A Raspberry Pi is based on a 32-bit microprocessor (Raspberry Pi 1 and original Raspberry Pi 2) or 64-bit microprocessor (Raspberry Pi 3 and more recent Raspberry Pi 2), it can run a full size operating system (such as Linux), and by doing so it can run multiple programs at a time.

MICRO CONTROLLER VS MICRO PROCESSOR

- Micro-controllers are meant to do only one specific task while microprocessor perform multiple tasks.
- In Micro-controller all hardware like RAM, ROM ,External memory with all other peripherals are embedded on a single chip but In microprocessor hardware are connected externally , It comes only with CPU.

EMBEDDED SYSTEMS

Embedded systems is the embedding (combining) the software with hardware to make a system. Hardware part in embedded systems is the micro-controller /microprocessor chip and software part is the program (code) running on this chip. **Micro-controller is the body of any embedded system and software is the mind which controls the complete behaviour of the system.** So, basically there is no difference between embedded systems and micro-controllers, instead, any electronic system made with micro-controller is known as Embedded Systems.

....

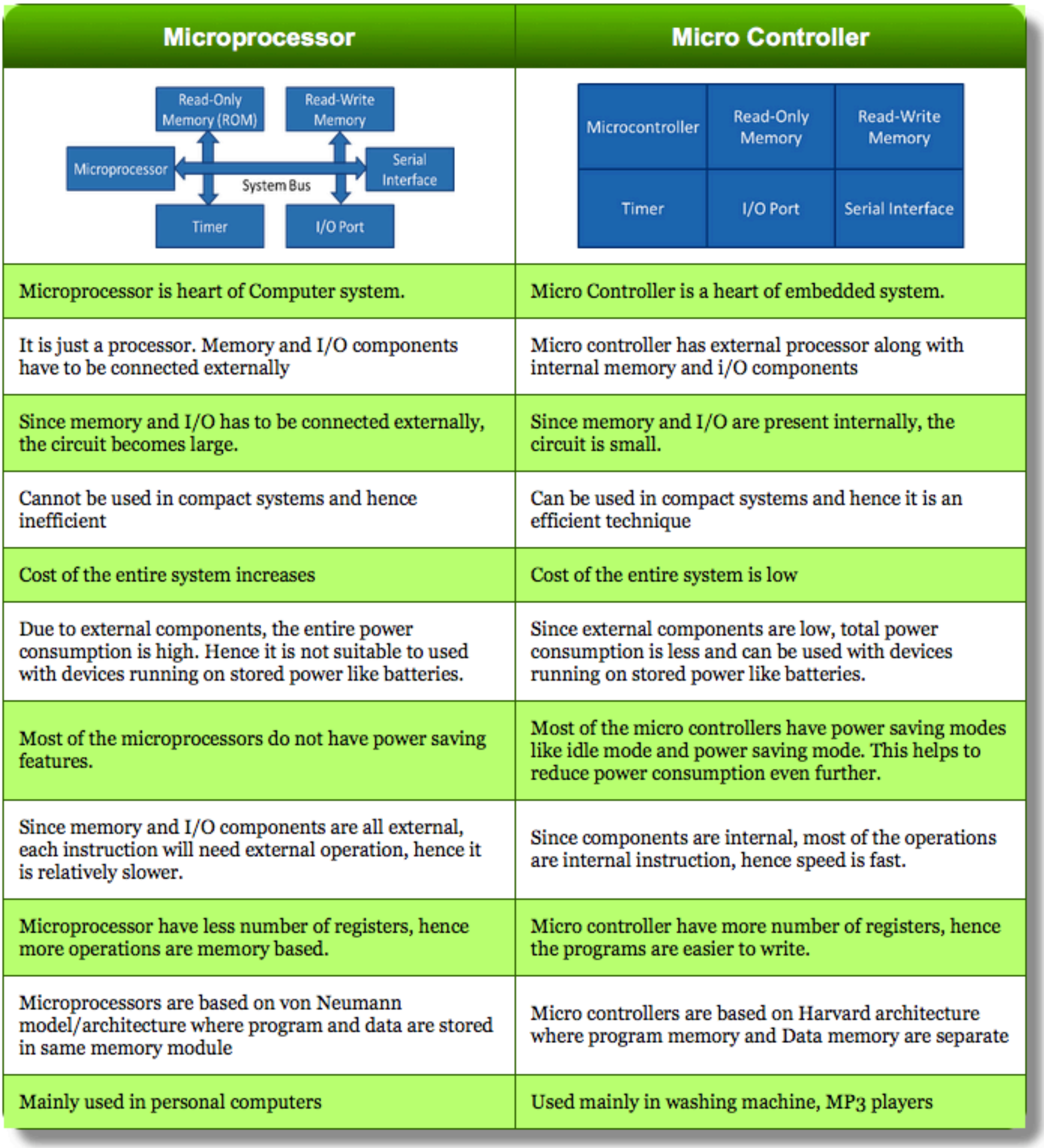
....

....

....

Microprocessor vs Micro Controller

Below is an image from the web including a significant comparison between the two



Microprocessor	Micro Controller
Read-Only Memory (ROM) Read-Write Memory Microprocessor System Bus Serial Interface Timer I/O Port	Microcontroller Read-Only Memory Read-Write Memory Timer I/O Port Serial Interface
Microprocessor is heart of Computer system.	Micro Controller is a heart of embedded system.
It is just a processor. Memory and I/O components have to be connected externally	Micro controller has external processor along with internal memory and i/O components
Since memory and I/O has to be connected externally, the circuit becomes large.	Since memory and I/O are present internally, the circuit is small.
Cannot be used in compact systems and hence inefficient	Can be used in compact systems and hence it is an efficient technique
Cost of the entire system increases	Cost of the entire system is low
Due to external components, the entire power consumption is high. Hence it is not suitable to used with devices running on stored power like batteries.	Since external components are low, total power consumption is less and can be used with devices running on stored power like batteries.
Most of the microprocessors do not have power saving features.	Most of the micro controllers have power saving modes like idle mode and power saving mode. This helps to reduce power consumption even further.
Since memory and I/O components are all external, each instruction will need external operation, hence it is relatively slower.	Since components are internal, most of the operations are internal instruction, hence speed is fast.
Microprocessor have less number of registers, hence more operations are memory based.	Micro controller have more number of registers, hence the programs are easier to write.
Microprocessors are based on von Neumann model/architecture where program and data are stored in same memory module	Micro controllers are based on Harvard architecture where program memory and Data memory are separate
Mainly used in personal computers	Used mainly in washing machine, MP3 players

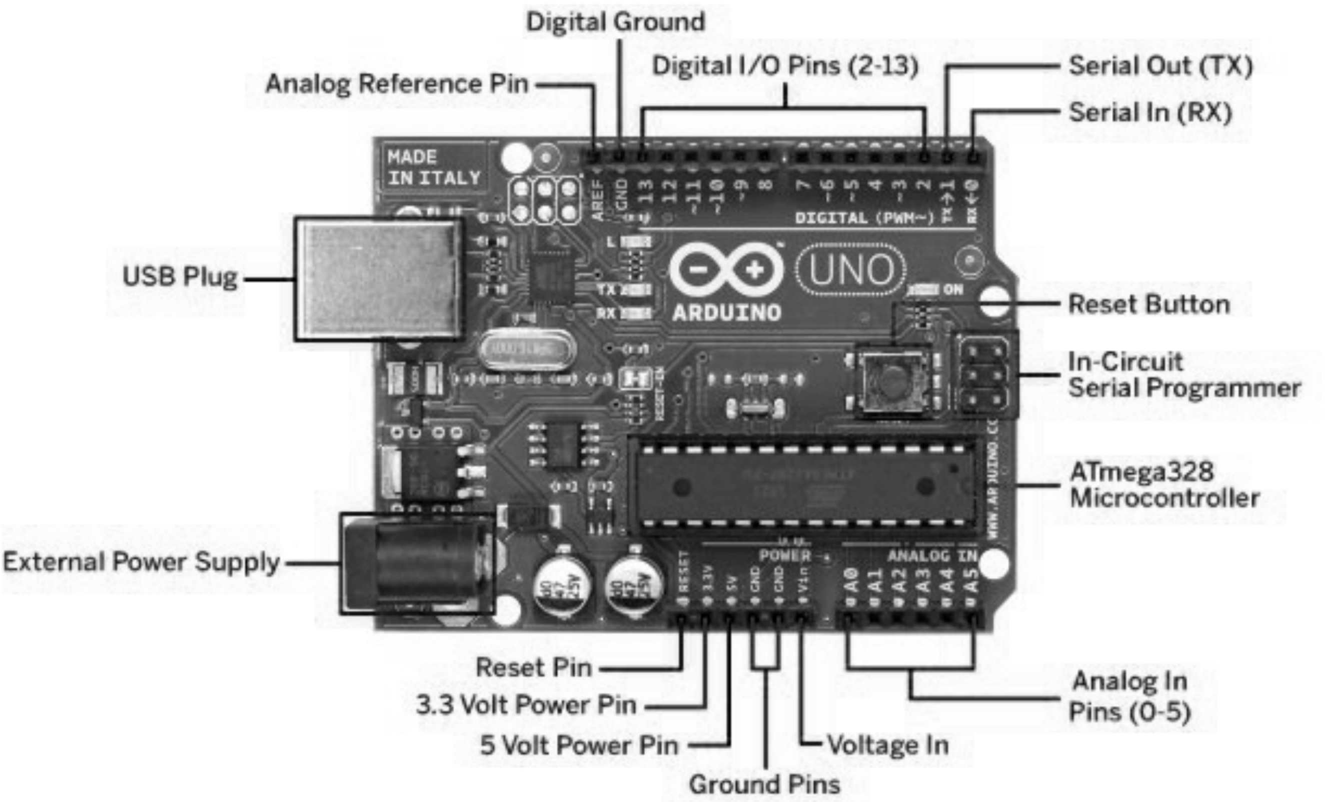
ARDUINO UNO R3

FEATURES OF THE ARDUINO UNO

- Microcontroller: ATmega328
- Operating Voltage: 5V
- Input Voltage (recommended): 7-12V
- Input Voltage (limits): 6-20V
- Digital I/O Pins: 14 (of which 6 provide PWM output)
- Analog Input Pins: 6
- DC Current per I/O Pin: 40 mA
- DC Current for 3.3V Pin: 50 mA
- Flash Memory: 32 KB of which 0.5 KB used by bootloader
- SRAM: 2 KB (ATmega328)
- EEPROM: 1 KB (ATmega328)
- Clock Speed: 16 MHz

ARDUINO IDE

"The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development boards" (wikipedia.org)

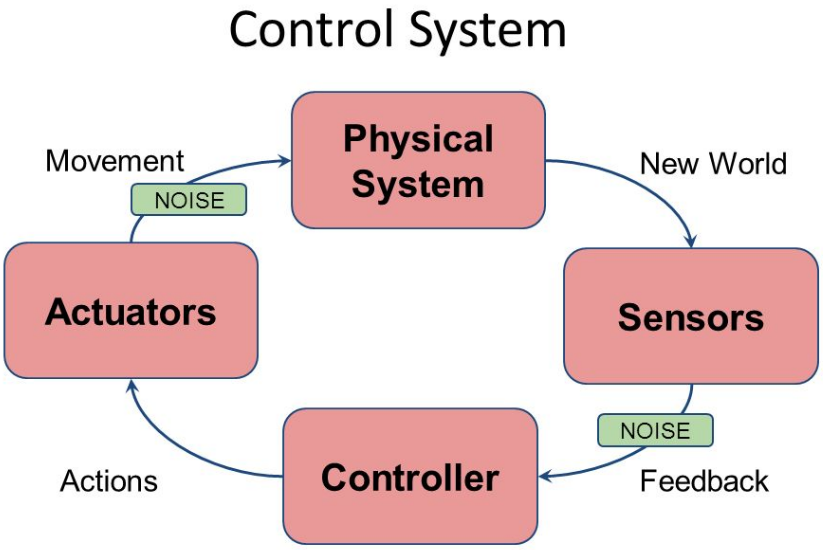


PIN DESCRIPTION

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. It can also be used for communication between two micro-controllers.
  - MISO (Master In Slave Out) - The Slave line for sending data to the master,
  - MOSI (Master Out Slave In) - The Master line for sending data to the peripherals,
  - SCK (Serial Clock) - The clock pulses which synchronize data transmission generated by the master
- **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.



Structure of the Code

Consist of mainly two functions:

setup() function is called when a sketch starts. Use it to initialise the variables, pin modes, start using libraries, etc. The setup function will only run once, after each power up or reset of the Arduino board.

loop() function does precisely what its name suggests, and loops consecutively, allowing your program to change and respond. Use it to actively control the Arduino board.

Arduino Built in Functions

pinMode(pin,mode)

mode – INPUT, OUTPUT, or INPUT_PULLUP

The pinMode() function is used to configure a specific pin to behave either as an input or an output. It is possible to enable the internal pull-up resistors with the mode INPUT_PULLUP.

digitalWrite (pin ,value)

value – HIGH, or LOW.

If the pin has been configured as an OUTPUT with pinMode(), its voltage will be set to the corresponding value: 5V (or 3.3V on 3.3V boards) for HIGH, 0V (ground) for LOW.

analogRead(pin)

Arduino is able to detect whether there is a voltage applied to one of its pins and report it through the digitalRead() function.

analogWrite (pin , value)

value – the duty cycle: between 0 (always off) and 255 (always on).

The **analogWrite()** function writes an analog value (PWM wave) to a pin. It can be used to light a LED at varying brightness or drive a motor at various speeds.

Pulse Width Modulation

Pulse Width Modulation or PWM is a common technique used to vary the width of the pulses in a pulse-train. PWM has many applications such as controlling servos and speed controllers, limiting the effective power of motors and LEDs.

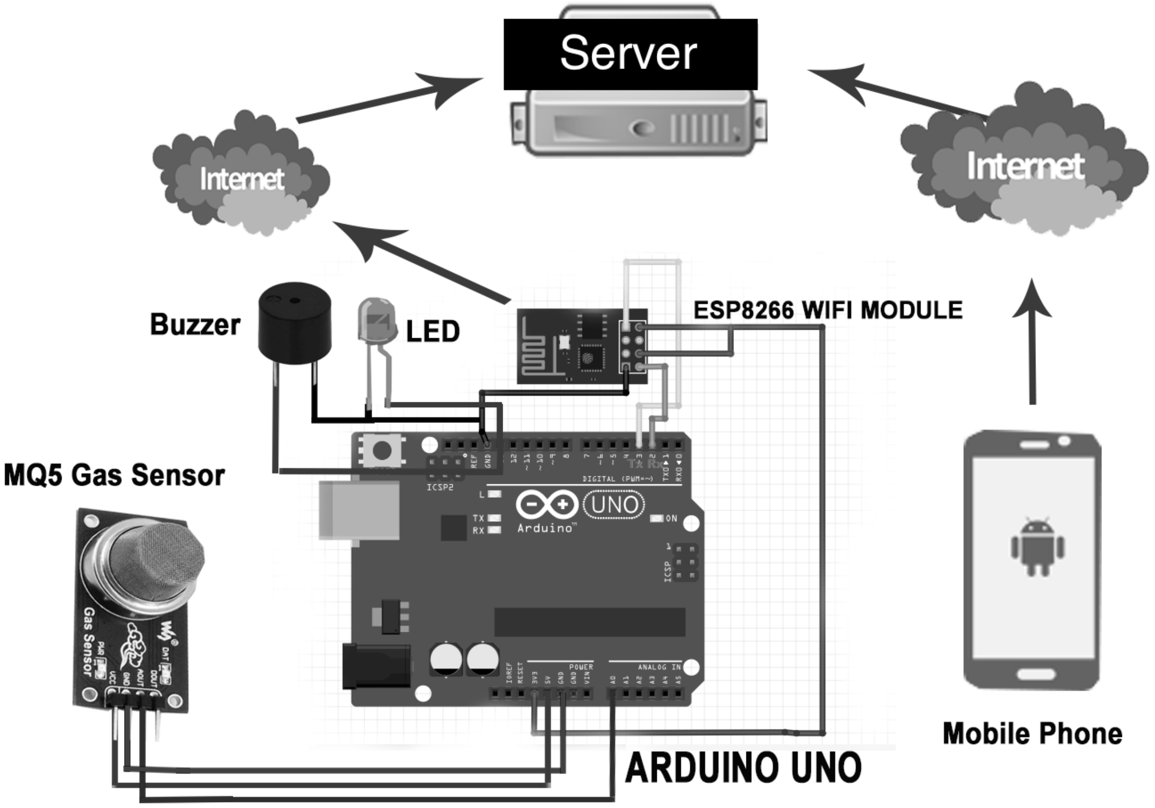
Pull Up Resistors

There are 20,000 pull-up resistors built into the Atmega chip that can be accessed from software. These built-in pull-up resistors are accessed by setting the **pinMode()** as INPUT_PULLUP. This effectively inverts the behaviour of the INPUT mode, where HIGH means the sensor is OFF and LOW means the sensor is ON.

The pull-up resistors provide enough current to light an LED dimly connected to a pin configured as an input. If LEDs in a project seem to be working, but very dimly, this is likely what is going on.

Internet of Things

IoT - a network of interoperated embedded system communicating with each other over the Internet.
The need for IoT in modern world where things communicate and interoperate with other things over the internet (Example: a smart refrigerator orders the items over the Internet from a supermarket automatically when items go below their re-order level)



Basic Components of IOT

- Sensor - component that identifies inputs (detects a state change in environment)
- Communication channel - a media that is used to establish a communication link between interoperating devices/systems
- Processing unit - component that makes decisions according to the inputs and some predetermined rules and then drives actuators
- Actuator - component that produces outputs (changes the state of environment)

For Sketches