PATRIOT lab

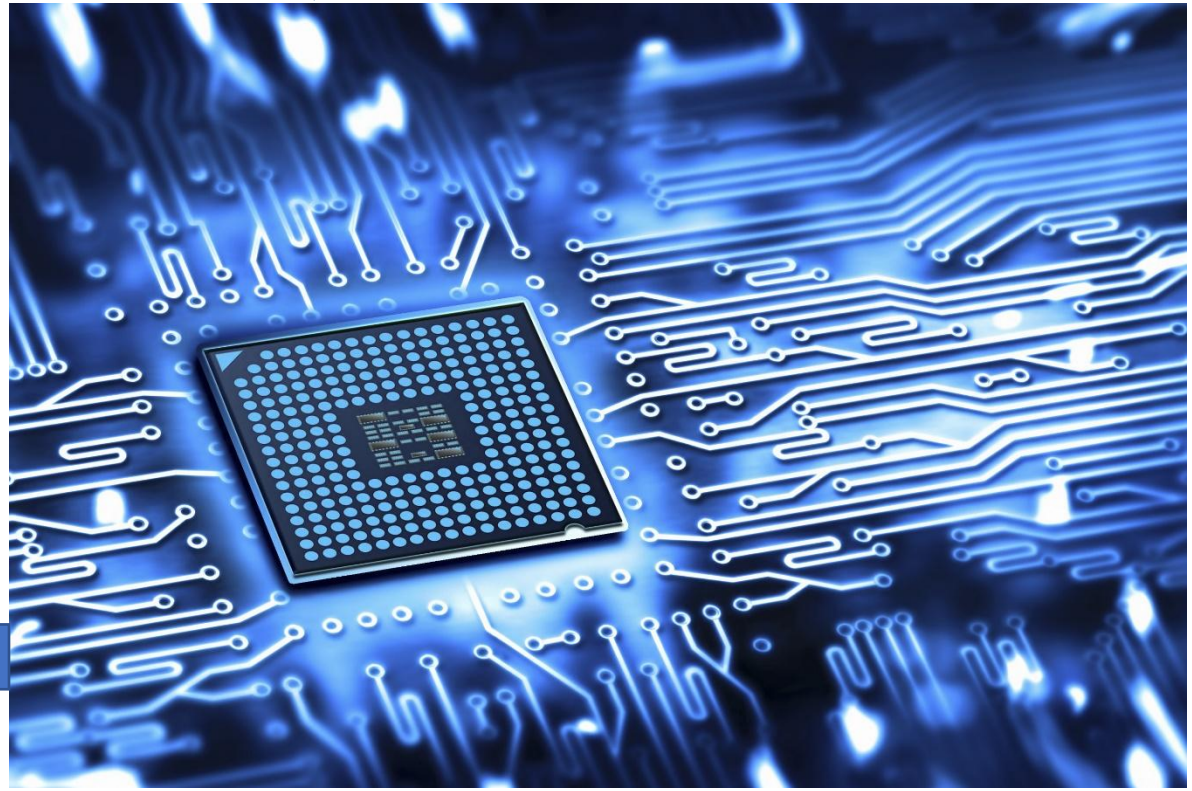
Lecture AMH2 – Board specifications

Dr. Aftab M. Hussain,

Associate Professor and Head, CVEST

Sensors and actuators

Sensor



Actuator

Need for boards

- We discussed that sensors and actuators are need to interact with the real world
- These are controlled using a microcontroller – which is a semiconductor IC that can store, process, and communicate information in digital form
- These controllers need to be programmed so that they perform a specific set of tasks
- The problem is: the hardware is a separate entity, whereas the program is typically written on a PC
- This is resolved by housing the processor on a “board” that can be connected to a PC
- This solves the hardware part, but on the PC, a compiler needs to convert the program into a binary file and write it into the controller

Need for boards

- There are many controller boards out there, but none as famous or widespread as Arduino uno
- Again, it is a shell around a microcontroller – ATMEGA328, that enables a PC to write code for it (USB communication, power etc.)
- On the software front, the Arduino IDE takes care of the C++ code compilation and writing to the board
- Other boards follow the same strategy – board + software


Arduino Nano 33


Arduino Nano Every


Arduino Nano 33 Ble

Arduino UNO


Teensy 4.0


Tessel 2

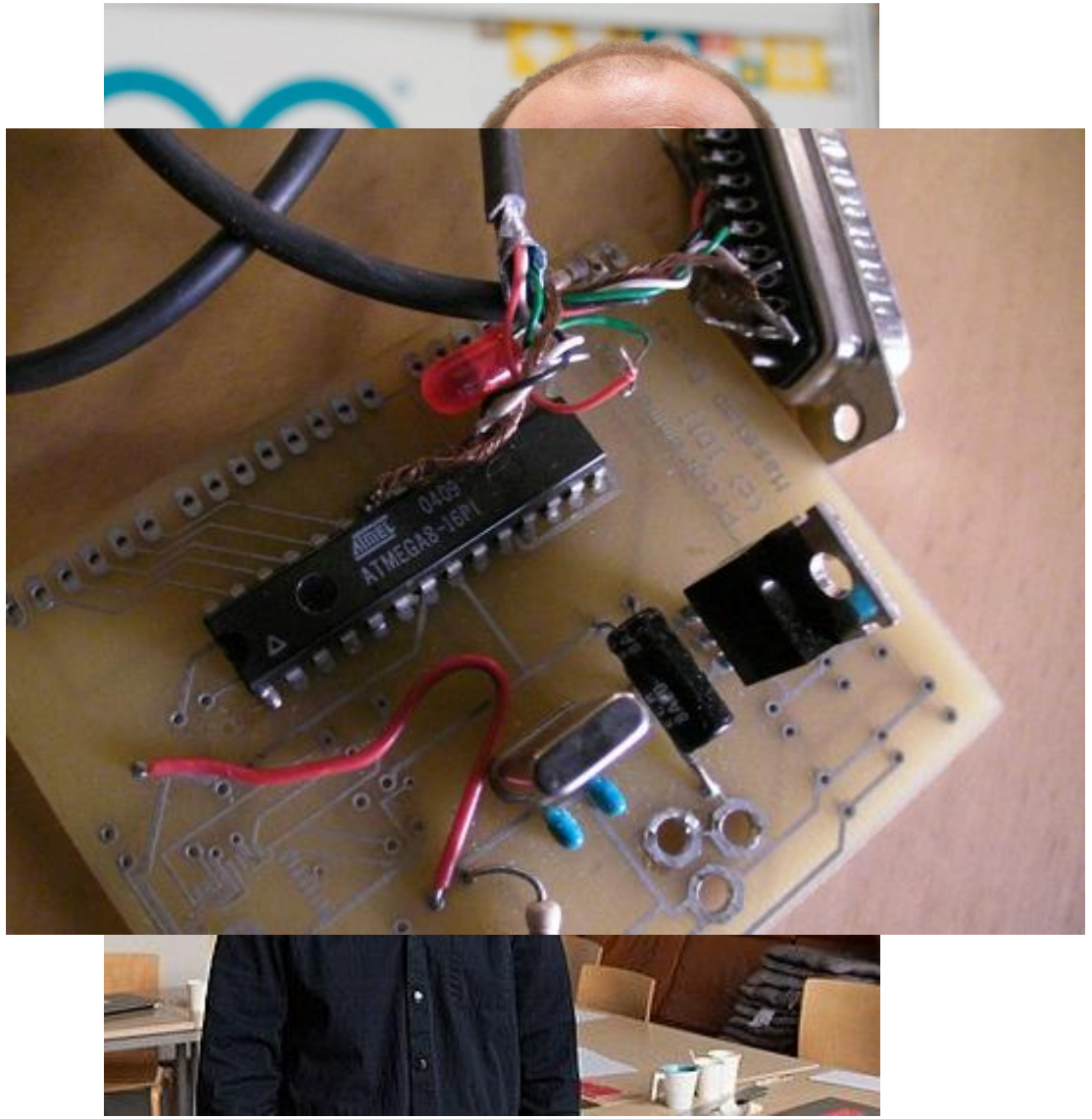
Onion Omega2s+


ESP32

Particle Argon

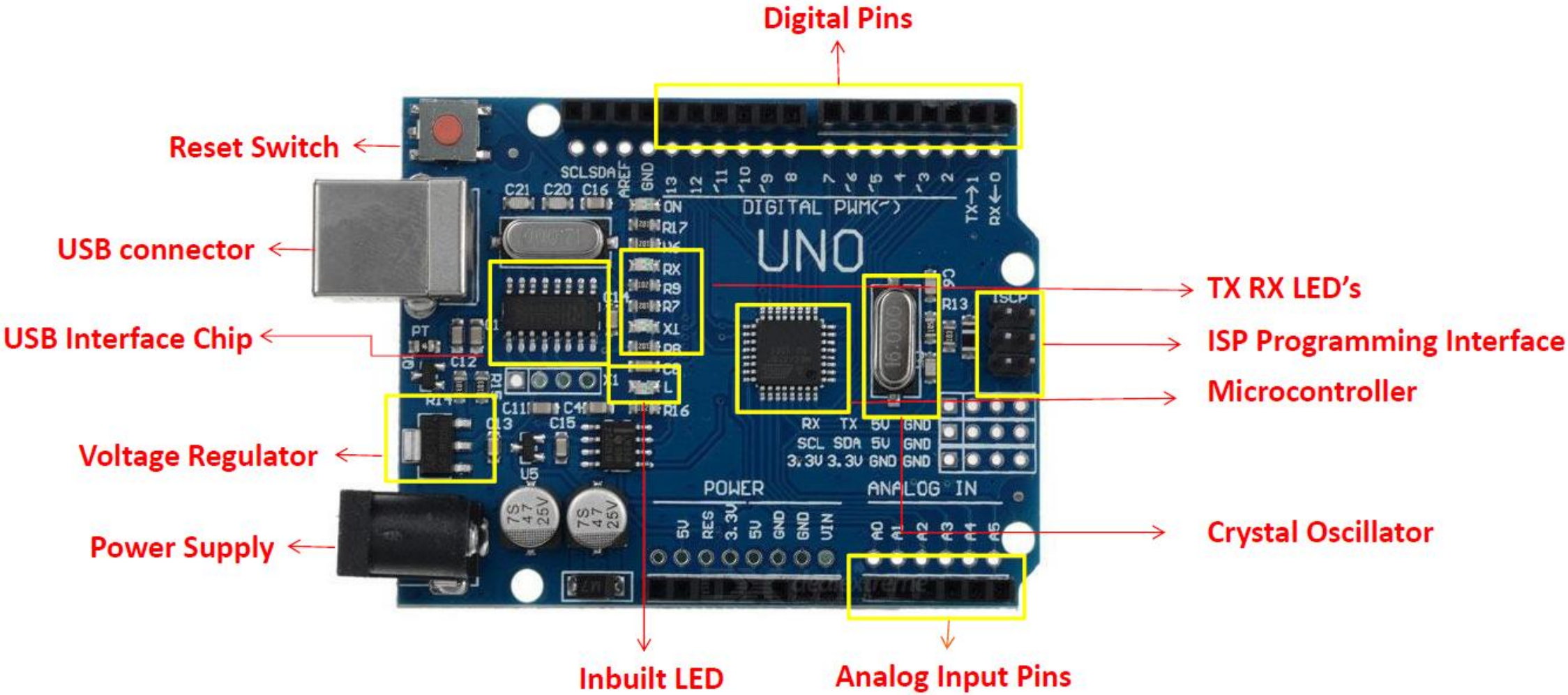
Some history

- In 2003 Hernando Barragán created the development platform Wiring as a Master's thesis project at IDII, under the supervision of Massimo Banzi and Casey Reas
- In 2005, Massimo Banzi, with David Mellis, another IDII student, and David Cuartielles, extended Wiring by adding support for the cheaper ATmega8 microcontroller
- The new project, forked from Wiring, was called *Arduino*.



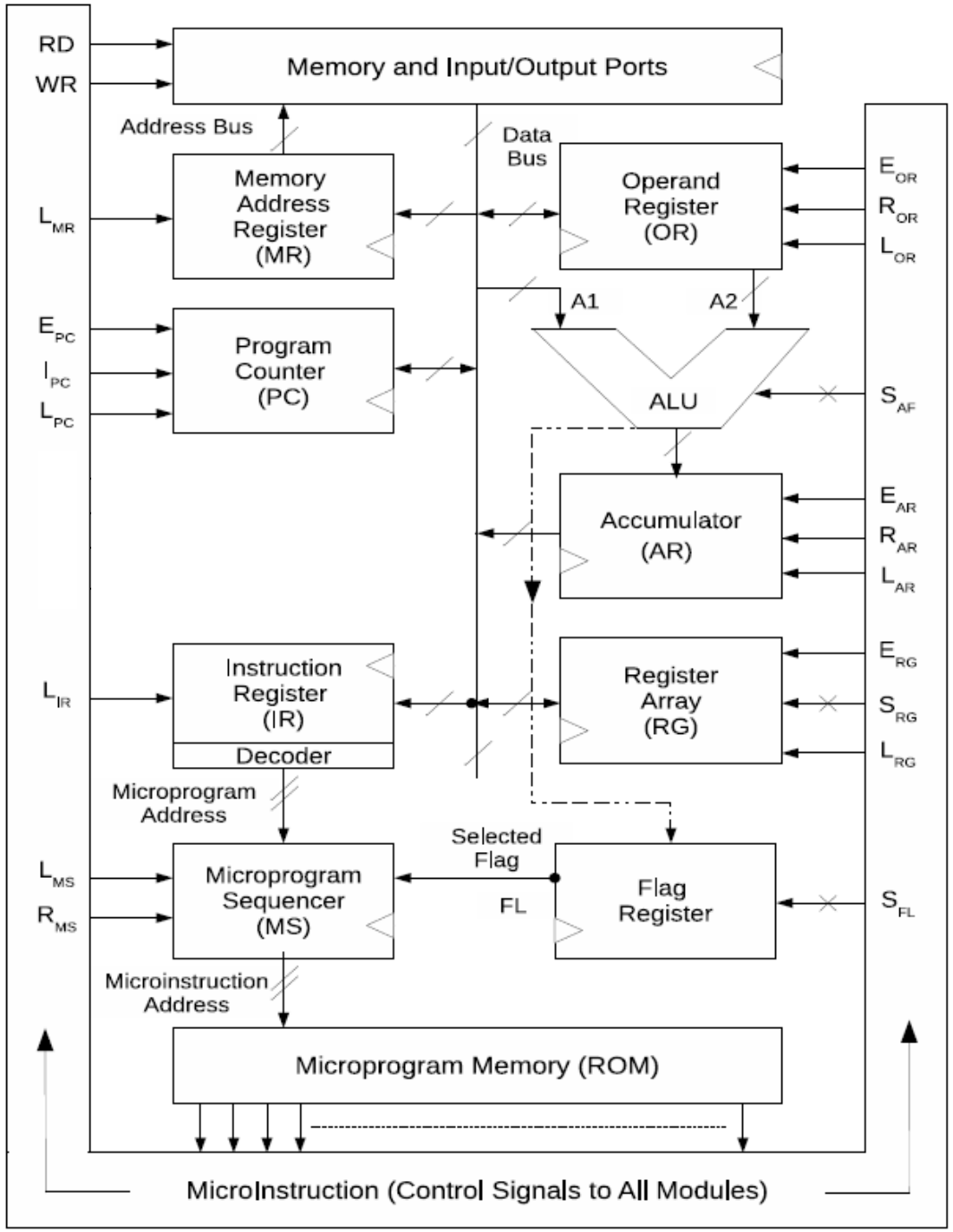
Casey Reas

Arduino Uno board



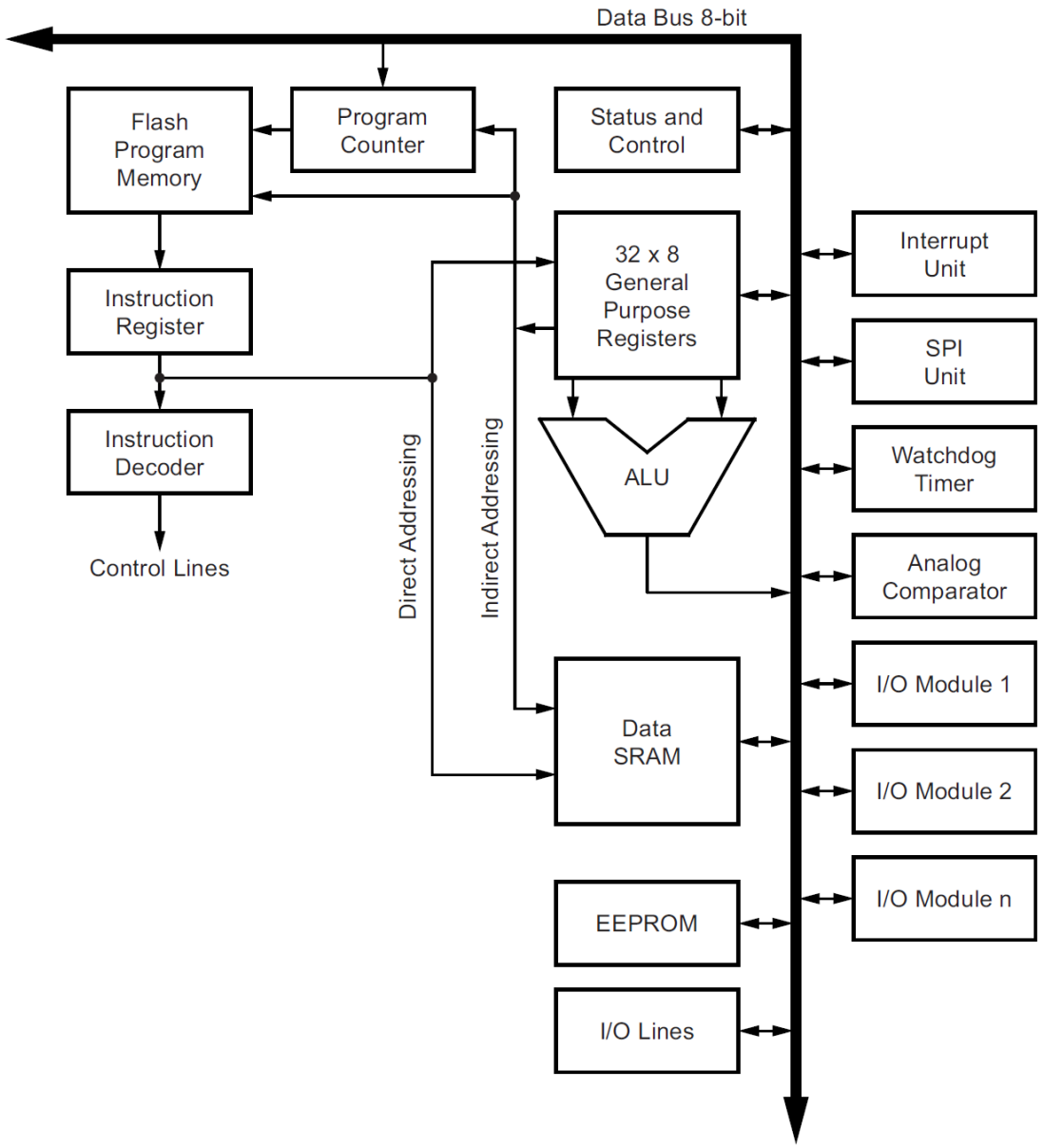
The final processor design

- This 8-bit, single bus processor can take instructions and data stored in the memory and perform tasks
- Each instruction is first fetched from memory into the IR, decoded and stored in the MS
- This is done using the address being pointed at by the PC (through the MR)
- Once, the microprogram word is obtained, the subsequent processing is done by the hardware
- Because the MS updated at the (delayed) negative edge of the clock, the control signals are available just after the negative edge , i.e., just before the positive edge



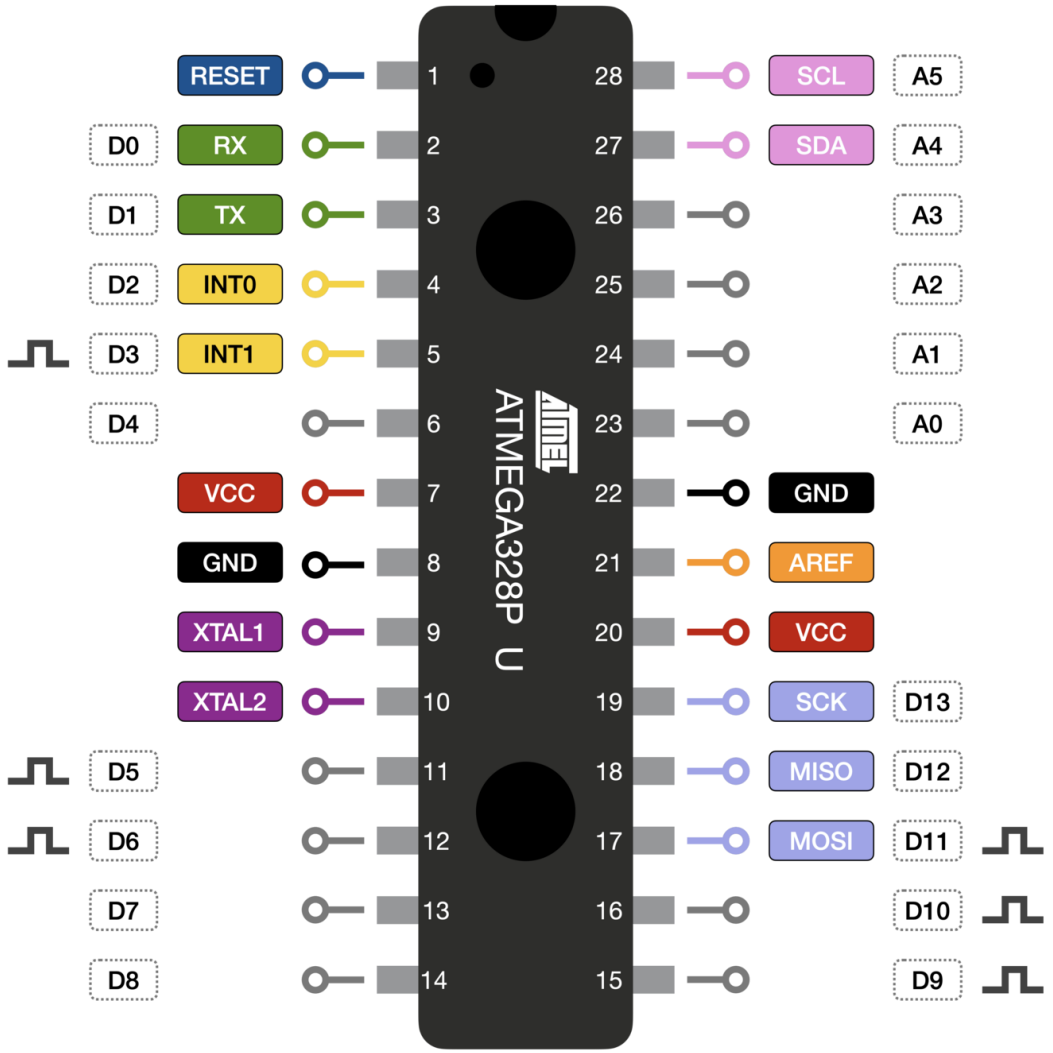
The microcontroller – ATMEGA328

- ATMEGA328 is an 8-bit microcontroller from Atmel Corporation (acquired by Microchip Systems)
- It has a modified Harvard architecture 8-bit RISC processor core (AVR)
- It includes General Purpose Input and Output (GPIO's), Serial Communication, External Interrupts, PWM, ANALOG INPUTS (6 Analog inputs with 10 bits resolution (1024 different values)) and SPI, I2C support
- No DACs, all analog write operations are PWM
- Typically, 16 MHz clock speed, max 20 MHz
- There are other IoT specialized options:
  - AREF: Reference voltage (0 to 5V only) for the analog inputs. Used with analogReference()
  - RESET: Bring this line LOW to reset the microcontroller. They are typically used to add a reset button to shields that block the one on the board



The microcontroller – ATMEGA328

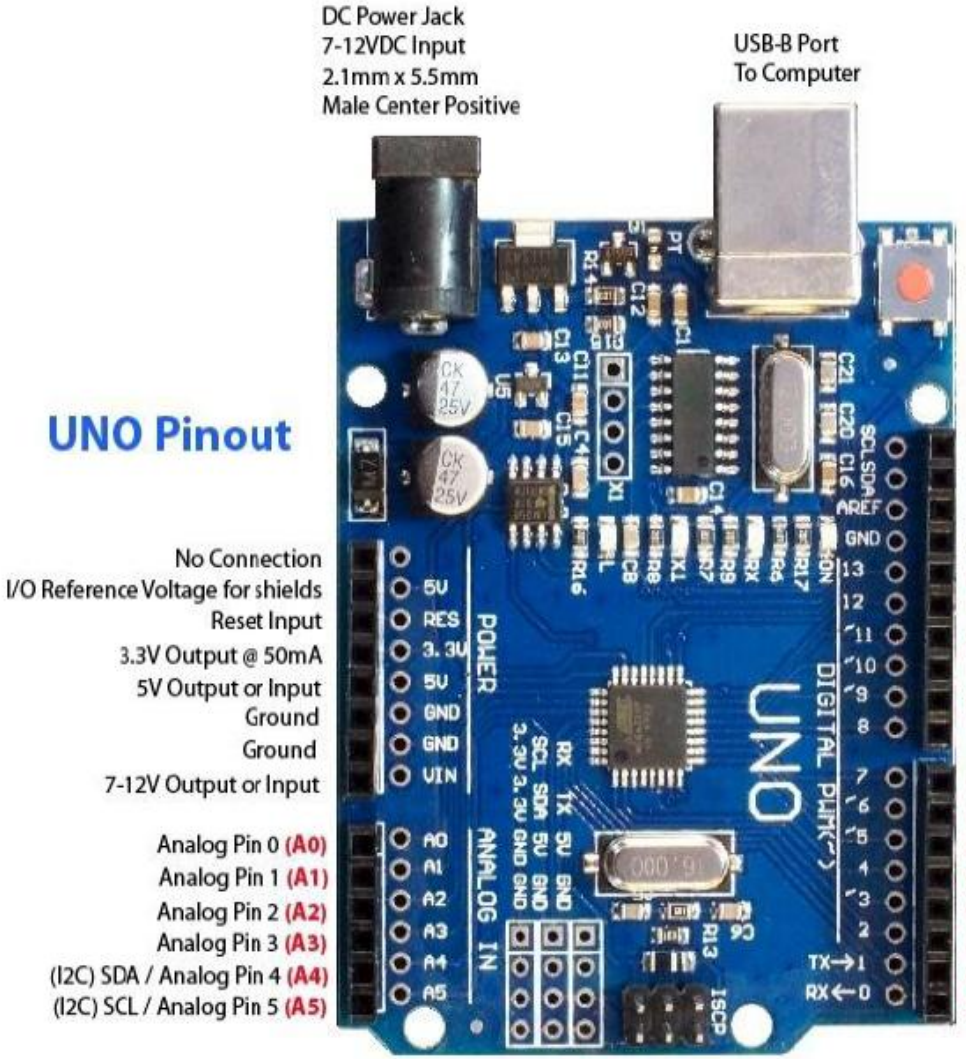
- The controller has a dedicated memory space allocated to store the program – Flash memory (revolutionary in itself!)
- This is the compiled version of the C++ code written in the IDE and includes all the information need to run the entire code, i.e., the libraries, functions, their addresses etc. (assembly level code)
- The controller has a RAM as a temporary storage of information during program execution
- ATMEGA328 has Flash 32kB (of which .5k is used for the bootloader), SRAM 2kB and EEPROM 1kB



Arduino Uno board

Specifications

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 (ATmega328)
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

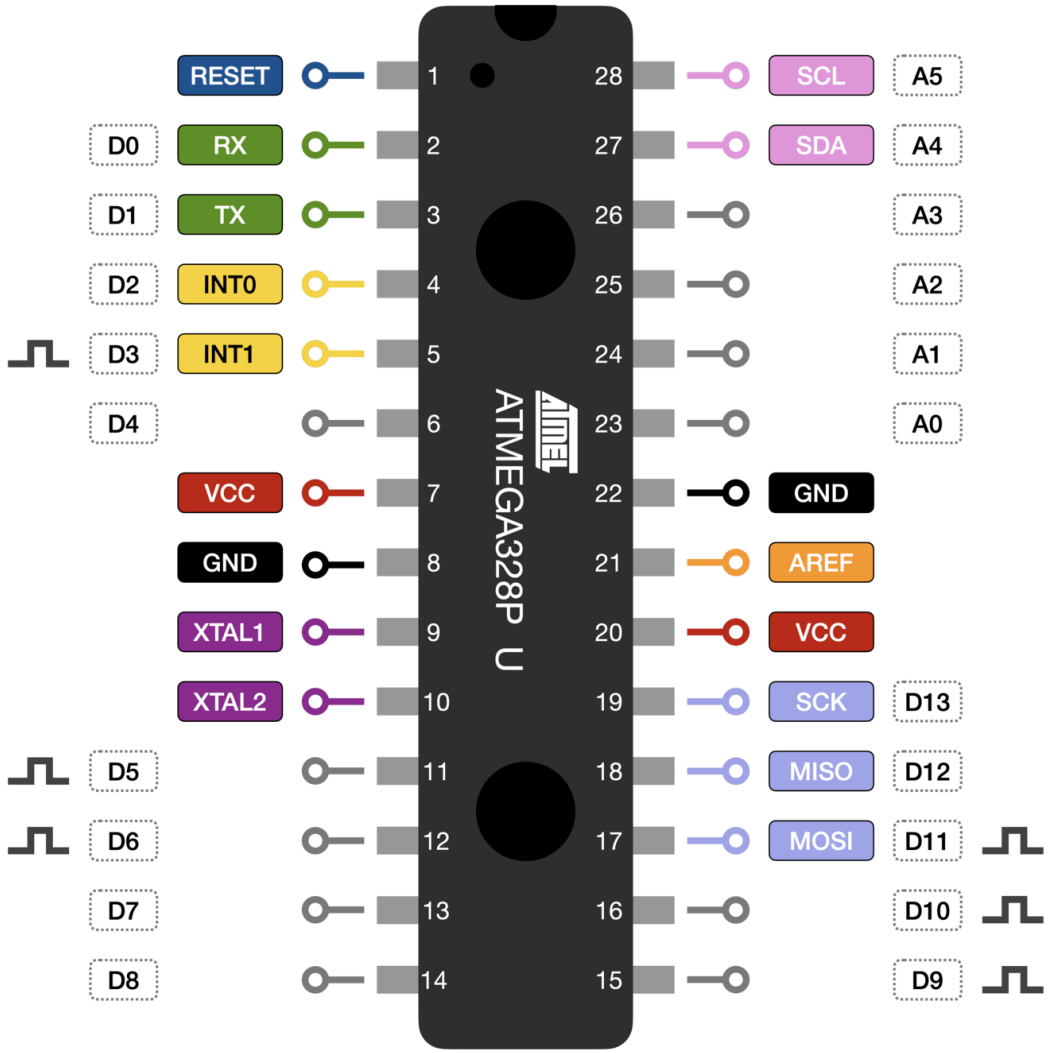


Red numbers in paranthesis are the name to use when referencing that pin.
Analog pins are references as A0 thru A5 even when using as digital I/O

Arduino UNO Pinout

The microcontroller – ATMEGA328

- The bootloader is the program that helps boot the controller and informs it of the first set of instructions it needs to perform
- This is separate from the setup() function in the program code – the bootloader makes sure that the binary file complied by the IDE is executed on controller reset
- How do you load the bootloader?
  - The program couter inside the controller is reset to “0x00” when the controller is reset. The program code for the bootloader sits at this location, this makes sure it is the first thing to be executed.



ESP32 controller board

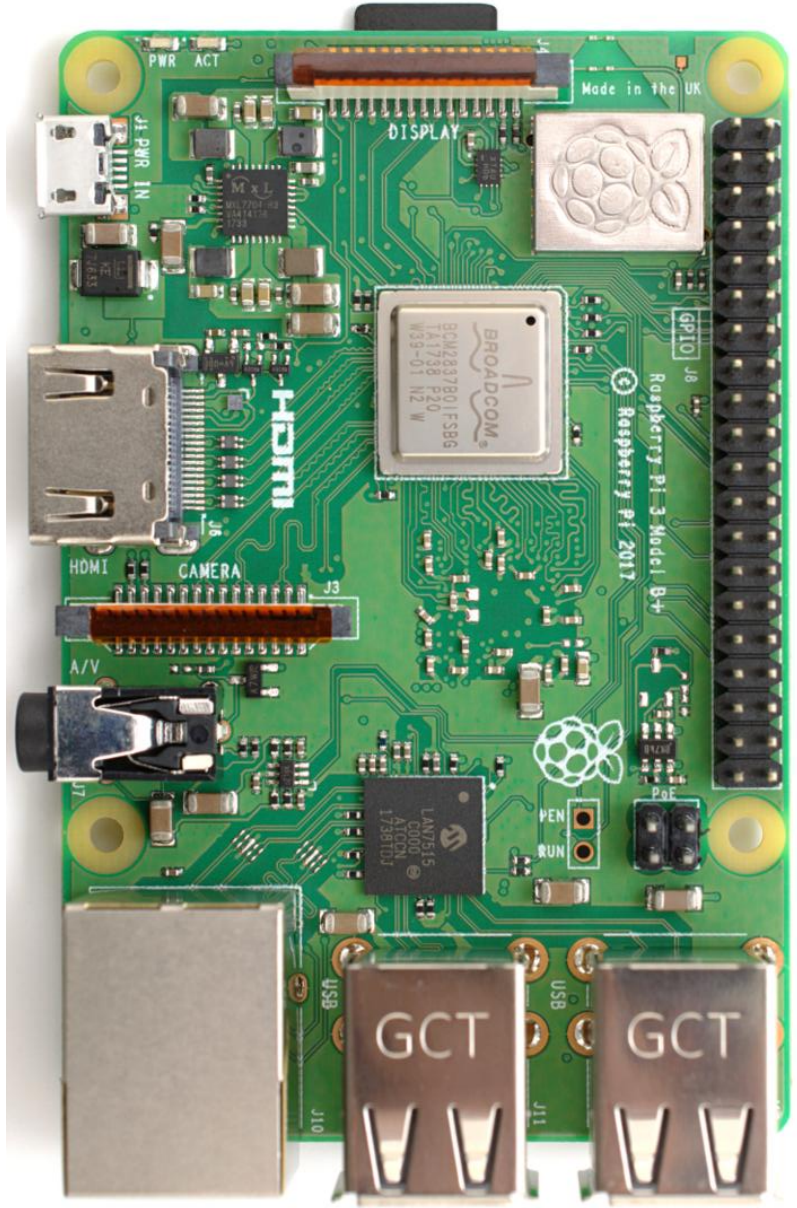
- Arduino uno was followed by the Mega, nano and other versions with support for various hobby projects
- This included support for BLE, WiFi, increased number of pins, reduced footprint and so on
- Apart from the Arduino upgrades, we have the nodeMCU board based on the ESP8266 and the ESP32 wroom board based on the ESP32 controller, both from Espressif Systems

ESP32 controller board

- ESP32 is a new generation, 32-bit, microcontroller from Espressif Systems
- It is a System-on-Chip (SoC) that includes the core, memory, and many of the wireless transceiver capabilities (like BLE and WiFi) on chip
- Meaning, we only need to connect it with an antenna for the BLE and WiFi to work
- It has 4 MB flash with a 320 kB RAM
- 18 ADC pins with 12-bit resolution
- 2 x 8-bit DAC pins, 4 × SPI, 2 × I²S interfaces, 2 × I²C interfaces, 3 × UART
- Maximum clock speed 240 MHz

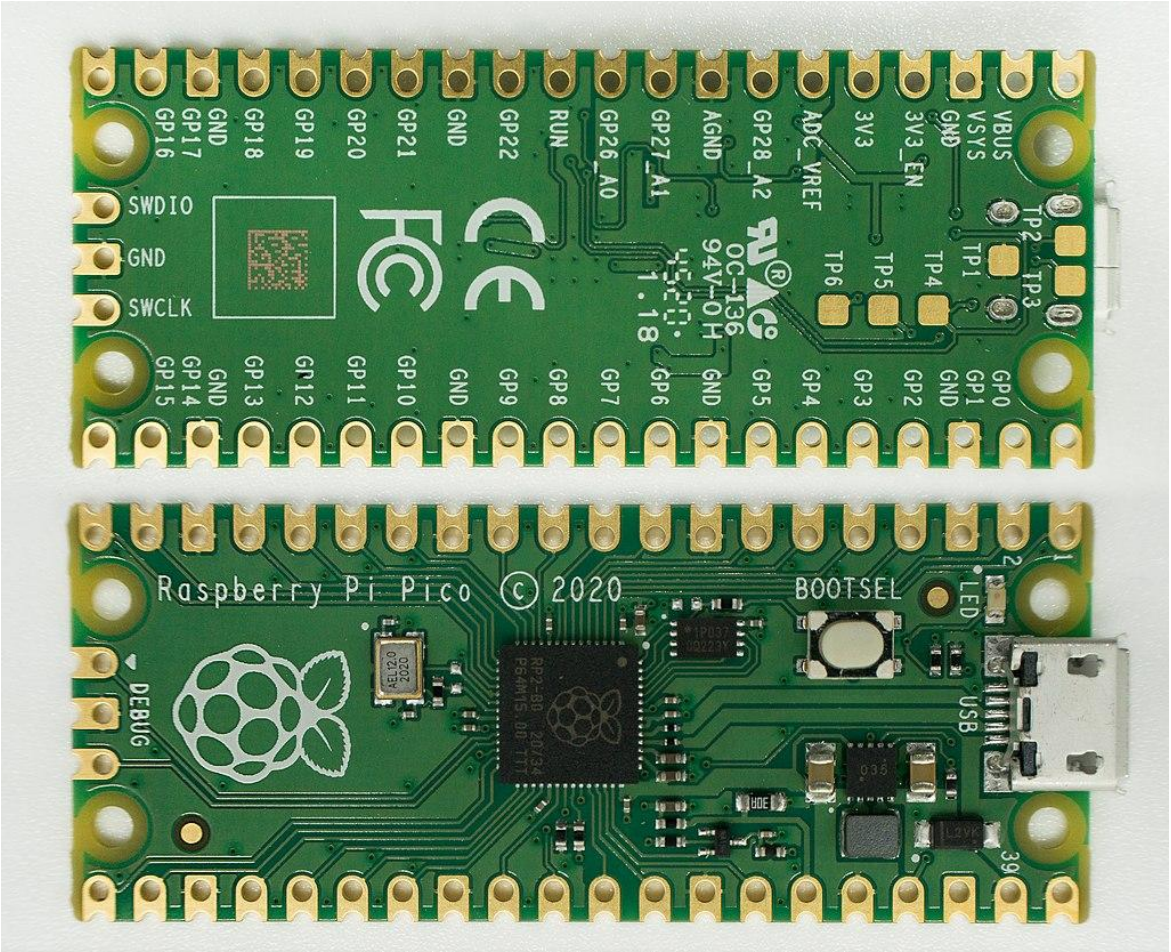
Raspberry Pi

- Going one step further, we can get an entire computer system working on through a small electronic board, if we have enough RAM to run an OS
- We then move to the Broadcomm *BCM2835* processor that was used in the first Pi released in 2012
- The RAM size was 256 MB, with a 1 GHz clock and several peripheral options
- Thus, the board could support a Linux kernel, USB interface, display, Ethernet interface



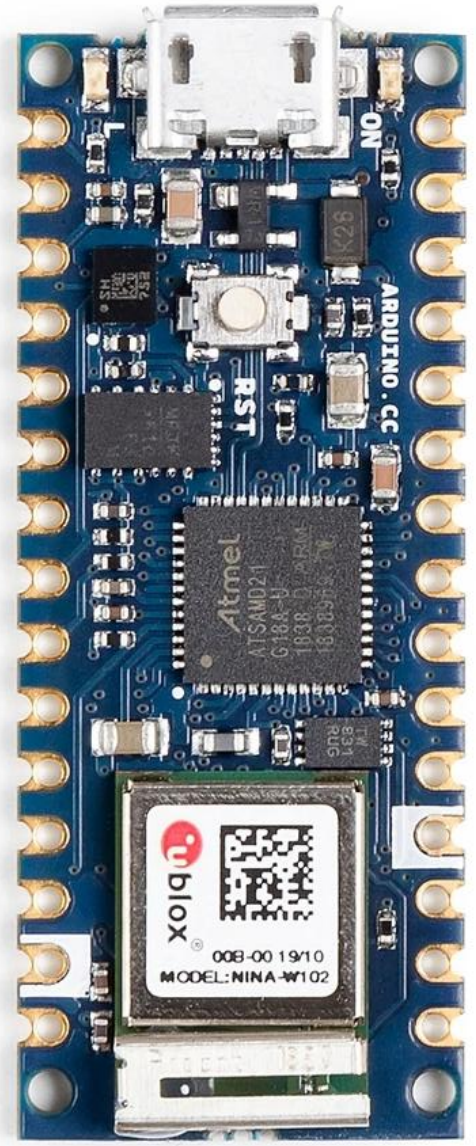
Raspberry Pi

- Many iteration of boards have been released since, including Pi zero, Pi pico etc
- It is based on the RP2040 - the first microcontroller designed by the Raspberry Pi Foundation
- 133 MHz clock, dual core, 264 kB RAM, support for SPI, I2C, UART etc.
- Huge Con:
  - No internal Flash or EEPROM memory (after reset, the boot-loader loads firmware from either external flash memory or USB bus into internal SRAM)



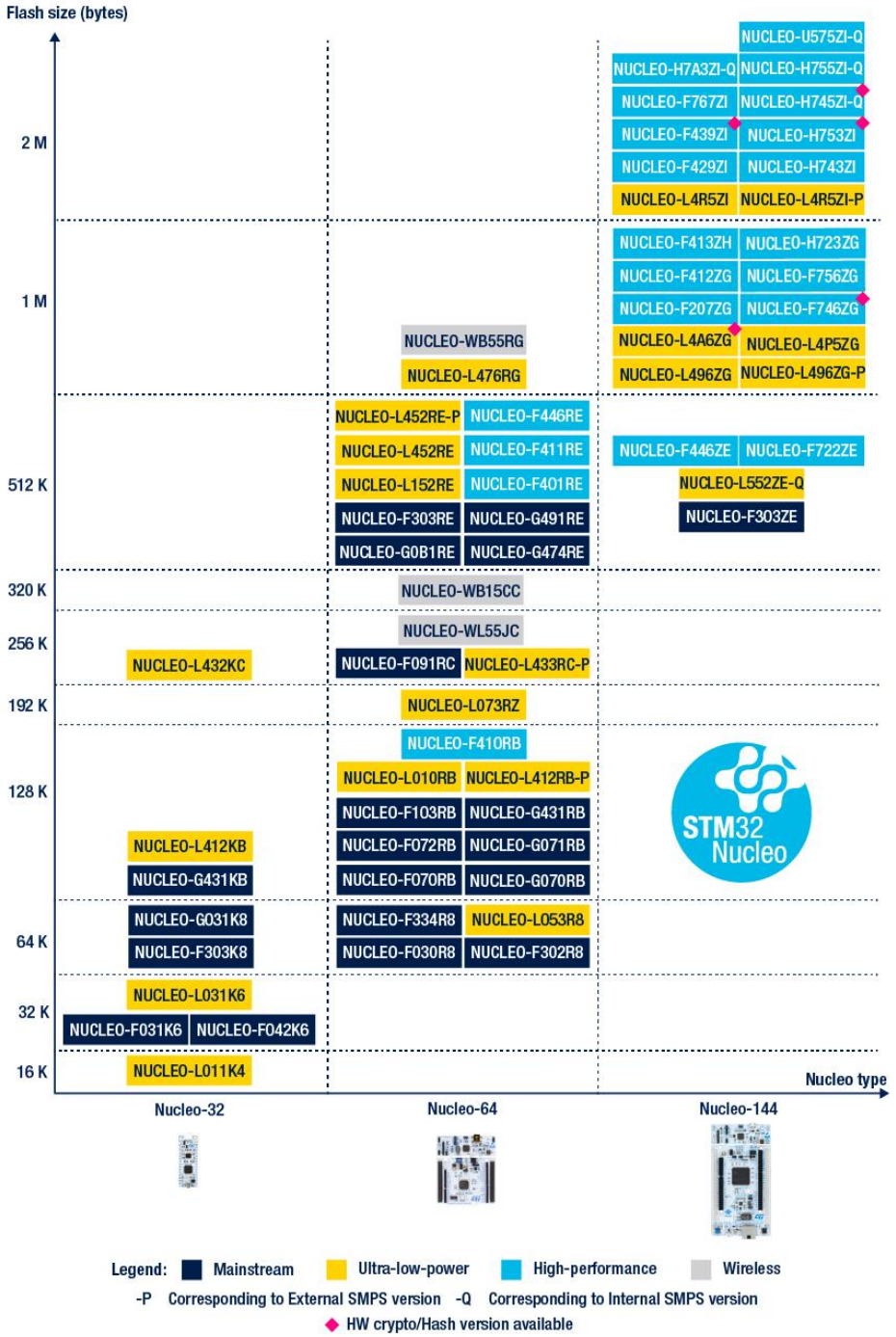
Arduino nano 33 IoT

- Meanwhile, the Arduino series has come with their own upgrade – the controller is ATSAMD21 from Microchip Technology
- 32 kB RAM, 256 kB flash, ARM® Cortex®-M0+ controller with 48 MHz clock
- 7xAnalog inputs with upto 12-bit resolution
- External ublox BLE and Wi-Fi module
- Works seamlessly with Arduino IDE and libraries
- Many variants in the market now have ATSAMD21 controller



Nucleo boards

- Not to be left behind in the boards-race, ST Microsystems has a line of Nucleo boards using their in-house controllers
- There are innumerable boards with tiny variations in the board design or change in the controller
- They can be programmed using the online ARM mbed compiler
  - Less intuitive interface than Arduino IDE for the basic user
  - Good for advanced user



Summary

	Arduino Uno	ESP32 wroom 32D	R Pi Pico	Arduino Nano 33 IoT
Microcontroller	ATMEGA328	ESP32	RP2040	ATSAMD21
Flash	32 kB	4 MB	No flash	256 kB
RAM	2 kB	320 kB	264 kB	32 kB
ADCs	10-bit	12-bit	12-bit	12-bit
Clock speed	16 MHz	Up to 240 MHz	133 MHz	48 MHz
SPI, I2C, UART?	Yes	Yes	Yes	Yes
I2S (hardware)	No	Yes	No	No
BLE/Wi-Fi	No	Yes (on-chip)	No	Yes (on-board)
Price	Rs. 550	Rs. 549	Rs. 349	Rs. 1899