08 - Microprocessor-Based Systems

CEG 4330/6330 - Microprocessor-Based Embedded Systems
Max Gilson

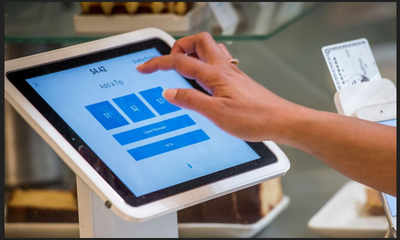
Microcontroller vs Microprocessor

- The Arduino Uno is a microcontroller
- Microcontrollers are usually an “all-in-one” solution
  - Usually contains memory, CPU, I/O, etc. all on one chip
  - Can have really small circuit board design
- Microprocessors are more powerful
  - Usually requires external RAM, storage, I/O, etc.
  - Results in larger circuit board, but more capabilities

Microcontroller vs Microprocessor (cont.)

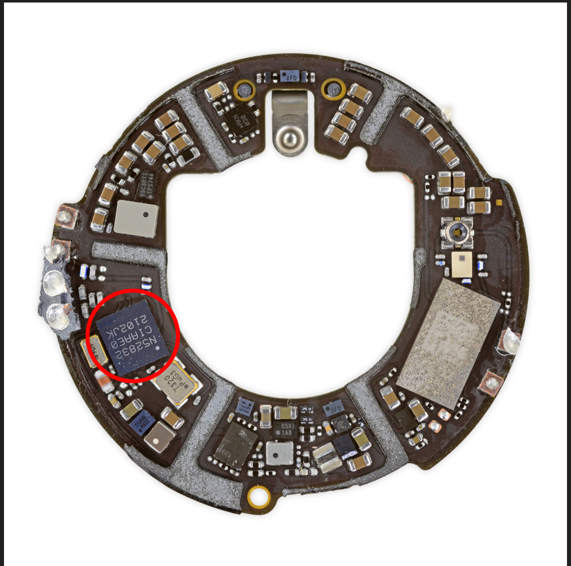
- Microcontroller
  - Used for “simple” applications:
  - control systems, washing machines, remote control, etc.

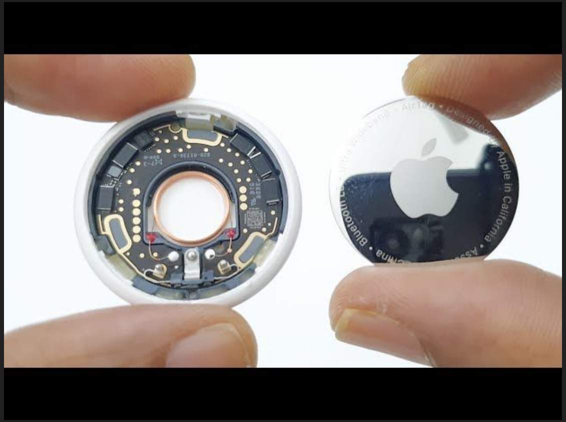
- Microprocessor
  - Used for “complex” applications:
  - computer vision, networking, smart watch, virtual reality headset, etc.



System On Chip

- A system on chip (SoC) is a specialized device that contains many functions and usually a microprocessor
- Example: N52832
  - Bluetooth SoC used in Apple's Airtag
  - Contains ARM processor and has many functions
  - The SoC is specialized for BLE capabilities, networking, and low power





Design Considerations

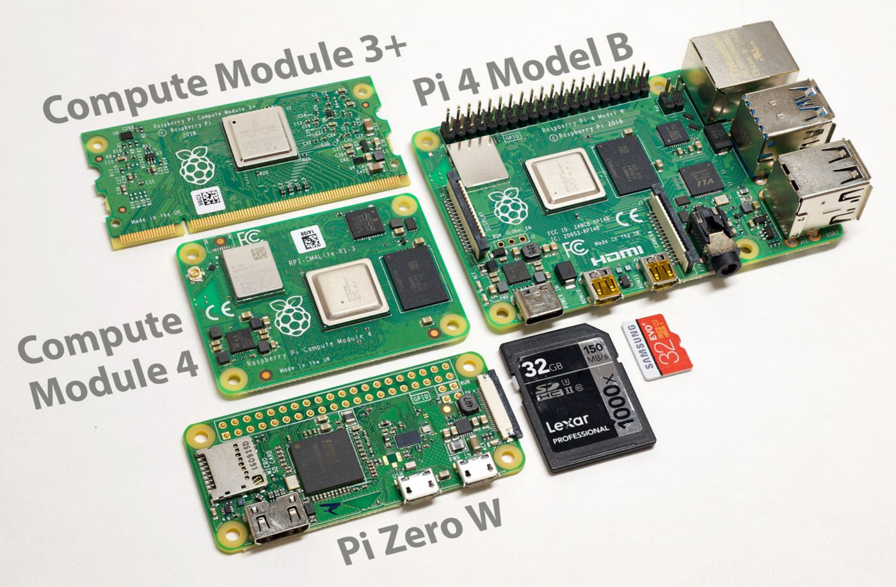
- When designing an embedded system many considerations have to be made when choosing a microprocessor:
    - Does it support my desired peripherals?
        - Bluetooth? Communication protocols? WiFi? Camera?
    - How much power consumption?
        - Is low power a requirement of your design?
    - How large is the device + external components?
        - How small does your circuit board need to be?
    - How much processing power or dedicated hardware?
        - High quality graphics might require integrated GPU

Many Choices

- Since there are so many considerations to make, there are many microprocessors available
- Many varieties of size, processing power, power consumption, I/O support, etc.

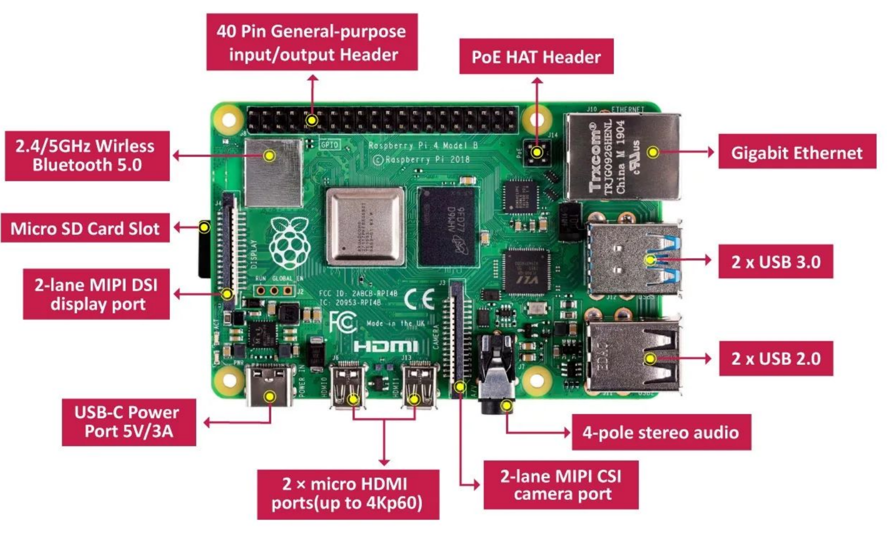
Raspberry Pi

- Single Board Computer (SBC)
- Open source software
- Closed source hardware
- “Off-the-shelf” solution for many applications
- There are many types of Raspberry Pi, and knock offs, with different functionality
- Lots of community support



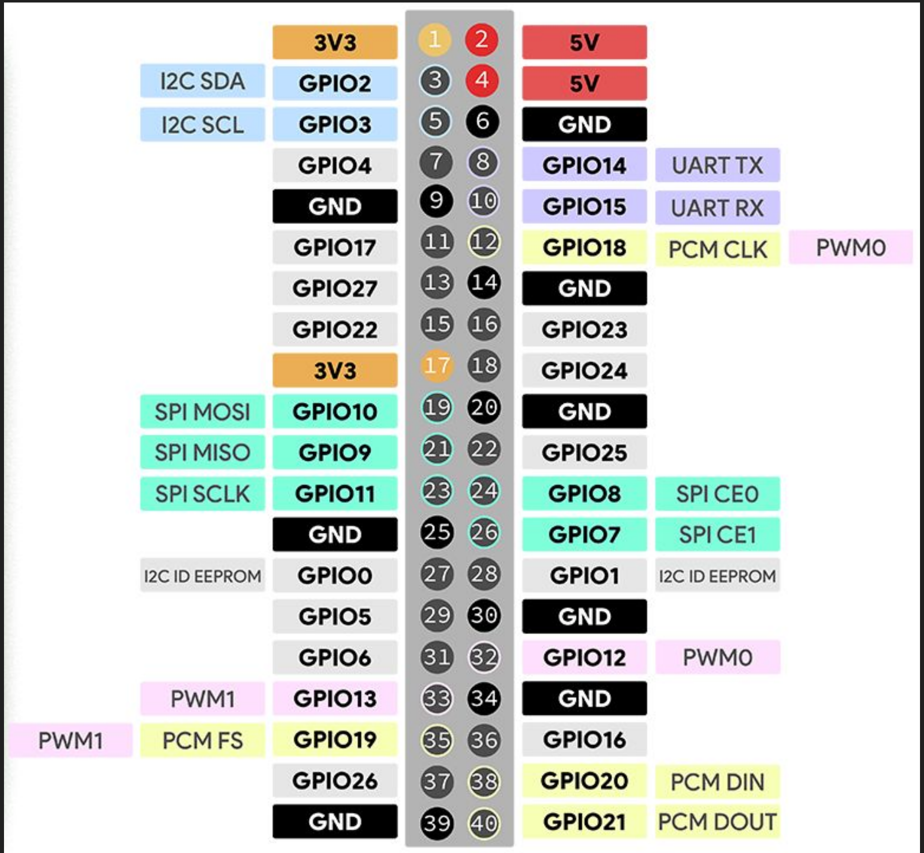
Raspberry Pi 4 Model B (RPI4B)

- Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.8GHz
- Ethernet
- USB3
- Dual 4K HDMI
- Camera port
- GPIO
  - Includes many functionalities
- Up to 8GB RAM


40 Pin General-purpose input/output Header
PoE HAT Header
2.4/5GHz Wirless Bluetooth 5.0
Gigabit Ethernet
Micro SD Card Slot
2 x USB 3.0
2-lane MIPI DSI display port
2 x USB 2.0
USB-C Power Port 5V/3A
4-pole stereo audio
2 × micro HDMI ports(up to 4Kp60)
2-lane MIPI CSI camera port

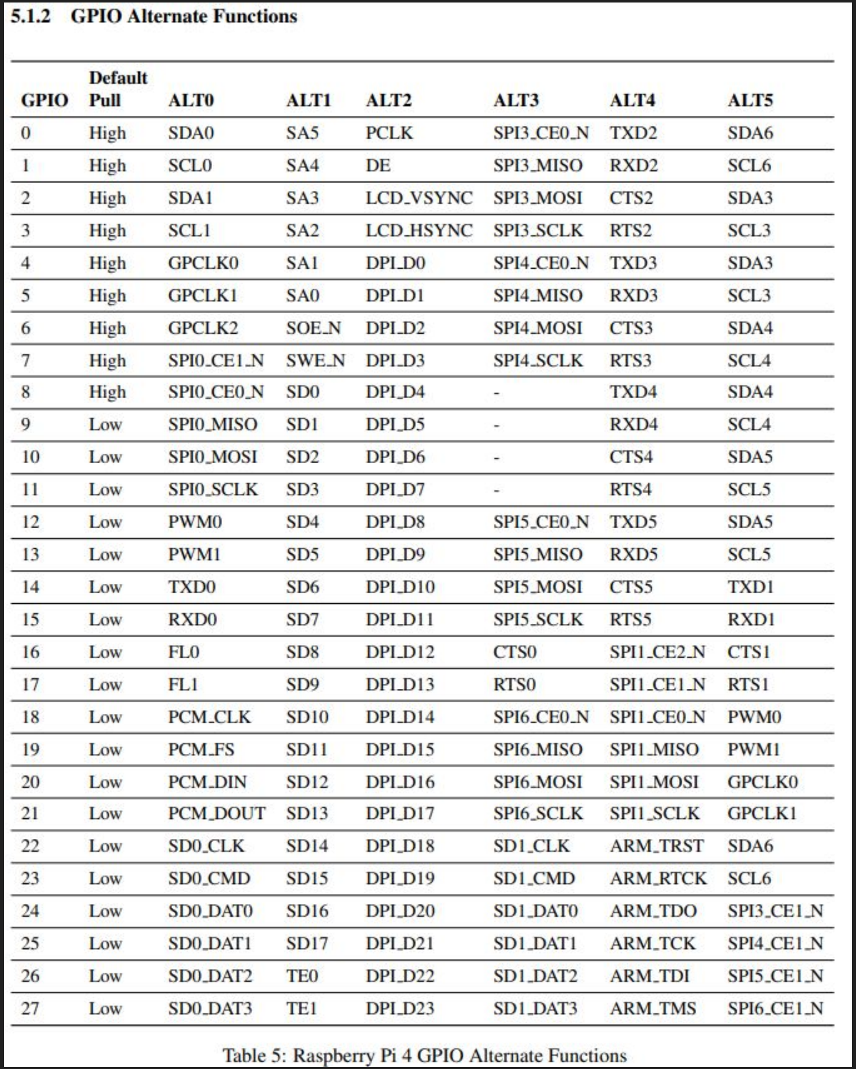
GPIO

- General Purpose I/O (GPIO)
- Pins can have multiple functions
- Some pins have dedicated hardware for communication, PWM, etc.



		3V3	1	2	5V		
	I2C SDA	GPIO2	3	4	5V		
	I2C SCL	GPIO3	5	6	GND		
		GPIO4	7	8	GPIO14	UART TX	
		GND	9	10	GPIO15	UART RX	
		GPIO17	11	12	GPIO18	PCM CLK	PWM0
		GPIO27	13	14	GND		
		GPIO22	15	16	GPIO23		
		3V3	17	18	GPIO24		
	SPI MOSI	GPIO10	19	20	GND		
	SPI MISO	GPIO9	21	22	GPIO25		
	SPI SCLK	GPIO11	23	24	GPIO8	SPI CE0	
		GND	25	26	GPIO7	SPI CE1	
	I2C ID EEPROM	GPIO0	27	28	GPIO1	I2C ID EEPROM	
		GPIO5	29	30	GND		
		GPIO6	31	32	GPIO12	PWM0	
	PWM1	GPIO13	33	34	GND		
PWM1	PCM FS	GPIO19	35	36	GPIO16		
		GPIO26	37	38	GPIO20	PCM DIN	
		GND	39	40	GPIO21	PCM DOUT	

GPIO (cont.)



5.1.2 GPIO Alternate Functions

GPIO	Default Pull	ALT0	ALT1	ALT2	ALT3	ALT4	ALT5
0	High	SDA0	SA5	PCLK	SPI3_CE0_N	TXD2	SDA6
1	High	SCL0	SA4	DE	SPI3_MISO	RXD2	SCL6
2	High	SDA1	SA3	LCD_VSYNC	SPI3_MOSI	CTS2	SDA3
3	High	SCL1	SA2	LCD_HSYNC	SPI3_SCLK	RTS2	SCL3
4	High	GPCLK0	SA1	DPI_D0	SPI4_CE0_N	TXD3	SDA3
5	High	GPCLK1	SA0	DPI_D1	SPI4_MISO	RXD3	SCL3
6	High	GPCLK2	SOE_N	DPI_D2	SPI4_MOSI	CTS3	SDA4
7	High	SPI0_CE1_N	SWE_N	DPI_D3	SPI4_SCLK	RTS3	SCL4
8	High	SPI0_CE0_N	SD0	DPI_D4	-	TXD4	SDA4
9	Low	SPI0_MISO	SD1	DPI_D5	-	RXD4	SCL4
10	Low	SPI0_MOSI	SD2	DPI_D6	-	CTS4	SDA5
11	Low	SPI0_SCLK	SD3	DPI_D7	-	RTS4	SCL5
12	Low	PWM0	SD4	DPI_D8	SPI5_CE0_N	TXD5	SDA5
13	Low	PWM1	SD5	DPI_D9	SPI5_MISO	RXD5	SCL5
14	Low	TXD0	SD6	DPI_D10	SPI5_MOSI	CTS5	TXD1
15	Low	RXD0	SD7	DPI_D11	SPI5_SCLK	RTS5	RXD1
16	Low	FL0	SD8	DPI_D12	CTS0	SPI1_CE2_N	CTS1
17	Low	FL1	SD9	DPI_D13	RTS0	SPI1_CE1_N	RTS1
18	Low	PCM_CLK	SD10	DPI_D14	SPI6_CE0_N	SPI1_CE0_N	PWM0
19	Low	PCM_FS	SD11	DPI_D15	SPI6_MISO	SPI1_MISO	PWM1
20	Low	PCM_DIN	SD12	DPI_D16	SPI6_MOSI	SPI1_MOSI	GPCLK0
21	Low	PCM_DOUT	SD13	DPI_D17	SPI6_SCLK	SPI1_SCLK	GPCLK1
22	Low	SD0_CLK	SD14	DPI_D18	SD1_CLK	ARM_TRST	SDA6
23	Low	SD0_CMD	SD15	DPI_D19	SD1_CMD	ARM_RTCK	SCL6
24	Low	SD0_DAT0	SD16	DPI_D20	SD1_DAT0	ARM_TDO	SPI3_CE1_N
25	Low	SD0_DAT1	SD17	DPI_D21	SD1_DAT1	ARM_TCK	SPI4_CE1_N
26	Low	SD0_DAT2	TE0	DPI_D22	SD1_DAT2	ARM_TDI	SPI5_CE1_N
27	Low	SD0_DAT3	TE1	DPI_D23	SD1_DAT3	ARM_TMS	SPI6_CE1_N

Table 5: Raspberry Pi 4 GPIO Alternate Functions

Operating System

- Microprocessors (including the RPI4B) require an operating system to run
- This operating system is usually some variety of Linux
- Linux is open source and usually free to use
- RPI4B uses Raspberry Pi OS (previously called Raspbian) as the officially supported operating system
  - Open source, Debian based, Linux operating system
  - Not as quick/easy as writing a simple program (like Arduino)
- Stored on SD card or EMMC

Accessing GPIO for Simple Functions

- First add permission to access GPIO:

```
sudo usermod -a -G gpio <username>
```

- Make sure Python and gpiozero is installed and run the following code to turn on an external LED (GPIO17) with a button press (GPIO2):

```
from gpiozero import LED, Button

led = LED(17)
button = Button(2)

while True:
    if button.is_pressed:
        led.on()
    else:
        led.off()
```

gpiozero Library

- https://gpiozero.readthedocs.io/en/stable/api_input.html
- The button class provides many different arguments
  - Pullup, pulldown, debounce, etc.
- There are many other sensors that are available
  - HC-SR04 distance sensor
  - TRCT5000 proximity sensor
  - SPI devices
- Outputs, inputs, and other helpful tools

Interfacing with Any I/O

- It is impossible to support interfacing with every I/O device
- The most popular are already implemented by the community or have an equivalent implemented
- Camera Solutions:
  - Raspberry Pi Camera Module
  - Arducam
- I2C IMU:
  - MPU9250
- Before buying a sensor, make sure you can find some available code for it for the RPI, otherwise, you’ll have to write your own drivers!

Interfacing with Any I/O

- It is impossible to support interfacing with every I/O device
- The most popular are already implemented by the community or have an equivalent implemented
- Camera Solutions:
  - Raspberry Pi Camera Module
  - Arducam
- I2C IMU:
  - MPU9250
- Before buying a sensor, make sure you can find some available code for it for the RPI, otherwise, you’ll have to write your own drivers!

IMU Python Code

```
import FaBo9Axis_MPU9250
import time
import sys

mpu9250 = FaBo9Axis_MPU9250.MPU9250()

try:
    while True:
        accel = mpu9250.readAccel()
        print " ax = " , ( accel['x'] )
        print " ay = " , ( accel['y'] )
        print " az = " , ( accel['z'] )

        gyro = mpu9250.readGyro()
        print " gx = " , ( gyro['x'] )
        print " gy = " , ( gyro['y'] )
        print " gz = " , ( gyro['z'] )

        mag = mpu9250.readMagnet()
        print " mx = " , ( mag['x'] )
        print " my = " , ( mag['y'] )
        print " mz = " , ( mag['z'] )
        print

        time.sleep(0.1)

except KeyboardInterrupt:
    sys.exit()
```

Low Power Modes

- The RPI 4B uses 2.7W while idle, 6.4W @ 400% CPU load
  - Pi-zero under light load consumes about 0.335W
- The most efficient x86 computer will consume 10W to 25W at idle
- Most SBC's allow for changing the clock speed, reducing power consumption for less processing power:
- RPI 4B clock speeds:
  - 1.8 Ghz (max overclocking)
  - 1.5 Ghz (nominal)
  - 0.6 Ghz (min underclocking)

Datasheet

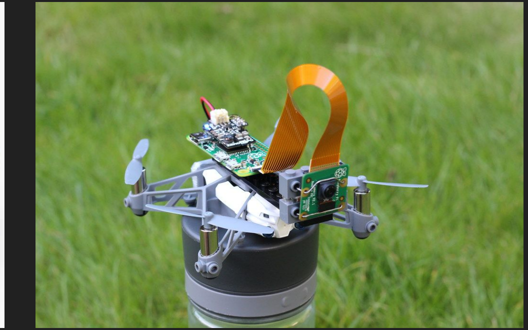
- Datasheet is very limited:
  - https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf
- Processor is closed source and documentation is protected under NDA
- Thankfully most of the tools and software is well documented and available so getting the RPI to work is no problem for most applications

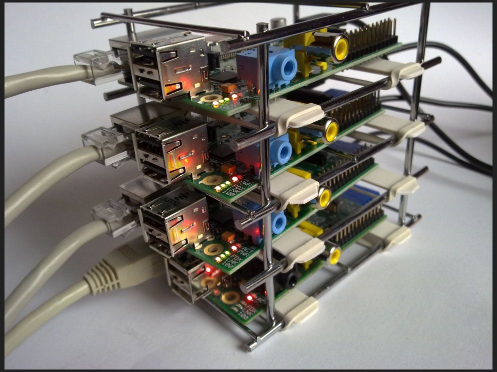
Raspberry Pi as a Desktop

- The Raspberry Pi is a fully capable computer
- Can be used as a general purpose computer
- Fully customizable

Many Applications

- Security Camera System
- Drone
- Web Server
- Smart Home / Home Automation
- Arcade Cabinet / Retro Console
- Open Source Phone





Alternatives to Raspberry Pi

- Many alternatives exist, providing specialized uses
- Small Form Factor:
  - NanoPi NEO Air
- AI and Image Processing:
  - Jetson Nano
- Open Source Hardware:
  - Banana Pi