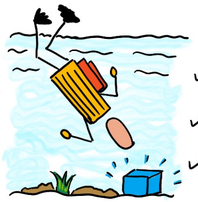


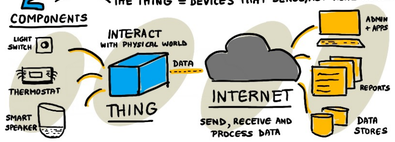
A DEEP DIVE!

- COMPONENTS OF IOT APPLICATIONS
- MICRO-CONTROLLERS A DEEPER DIVE
- SINGLE BOARD COMPUTERS A DEEPER DIVE

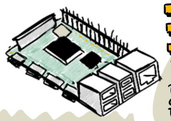
COMPONENTS OF IOT APPS

2 COMPONENTS

- THE INTERNET = APPS THAT RECEIVE/SEND DATA
- THE THING = DEVICES THAT SENSE/ACT REAL WORLD



THE THING



- LOW POWER
- LOW COST
- LOW SPEED

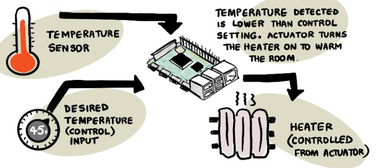
COMPUTERS

Run for long periods
Gather data (sensors)
Take actions (actuators)

Example
Microcontroller
RAM in KB
SPEED in MHz

A DEVICE THAT CAN INTERACT WITH PHYSICAL WORLD

EX: A THERMOSTAT!



DO THE RESEARCH

WHAT OTHER SYSTEMS ARE AROUND YOU THAT READ SENSOR DATA & USE IT TO MAKE DECISIONS

THE INTERNET

CONSISTS OF APPS THAT

- PROCESS DATA
- SEND/RECEIVE MESSAGES
- TAKE DECISIONS ON REQUESTS TO SEND TO ACTUATORS

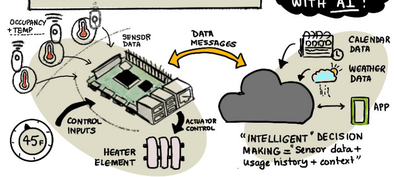
TYPICAL SETUP

- CLOUD SERVICE INTERMEDIARY
- HANDLE SECURITY
- INTERACT WITH SENSORS ON ONE SIDE
- CONNECT TO APPS ON THE OTHER

DEVICES CONNECT TO EACH OTHER (BT, WiFi) AND TO "CONNECTED" HUB

A 'SMARTER' THERMOSTAT

USE CLOUD WITH AI!



DO THE RESEARCH

WHAT OTHER DATA COULD MAKE THERMOSTAT EVEN SMARTER?

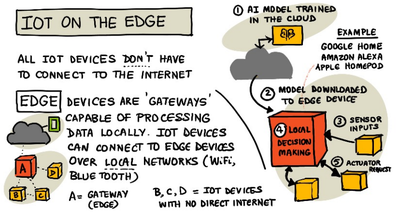
IOT ON THE EDGE

ALL IOT DEVICES DON'T HAVE TO CONNECT TO THE INTERNET

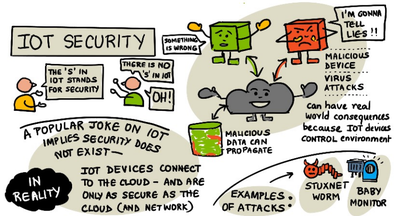
EDGE DEVICES ARE 'GATEWAYS' CAPABLE OF PROCESSING DATA LOCALLY. IOT DEVICES CAN CONNECT TO EDGE DEVICES OVER LOCAL NETWORKS (Wi-Fi, BLUE TOOTH)

A = GATEWAY (EDGE)

B, C, D = IOT DEVICES WITH NO DIRECT INTERNET



IOT SECURITY



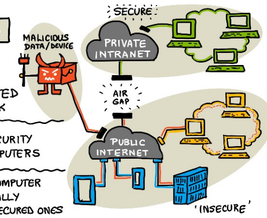
AIR GAPPING

aka

- ✓ AIR WALL
- ✓ AIR GAP
- ✓ DISCONNECTED NETWORK

IS A NETWORK SECURITY MEASURE FOR COMPUTERS

WHERE A SECURE COMPUTER NETWORK IS PHYSICALLY ISOLATED FROM UNSECURED ONES



DO THE RESEARCH

WHAT OTHER SECURITY CHALLENGES OR SCENARIOS CAN YOU THINK OF FOR IOT SYSTEMS AROUND YOU?

MICROCONTROLLERS: CPU

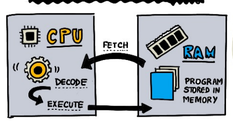
THE CPU (CENTRAL PROCESSING UNIT) IS THE BRAIN OF THE MICROCONTROLLER.

SENDS/RECEIVES MESSAGES

EXECUTES ONE INSTRUCTION PER CLOCK TICK → MILLIONS OR BILLIONS OF TICKS PER SEC

HIGHER THE 'SPEED' THE MORE INSTRUCTIONS RUN/SEC

MICROCONTROLLERS: CPU



ON CLOCK TICK

CPU FETCHES INSTRUCTION, DECODES IT

THEN EXECUTES THE TASK

SOME INSTRUCTIONS WILL TAKE MULTIPLE CLOCK TICKS...

DO THE RESEARCH

EVERY CLOCK TICK DRAWS POWER, GENERATES HEAT!

MICROCONTROLLERS RUN AT LOWER SPEEDS OR SWITCH TO LOW POWER CORES TO REDUCE OVERHEATING...

MICROCONTROLLERS: MEMORY

THERE ARE 2 TYPES OF MEMORY

PROGRAM MEMORY

- STORES YOUR CODE (PROGRAM)
- PERSISTS WHEN THERE IS NO POWER

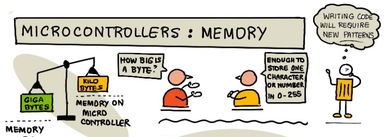
HERE ALWAYS

RANDOM ACCESS MEMORY

- USED TO RUN YOUR CODE WHEN POWERED
- RESETS WHEN THERE IS NO POWER

I NEED POWER TO BE ACTIVE

MICROCONTROLLERS: MEMORY



MEMORY ON PC

W.I.O TERMINAL = 192 kB RAM

BASE PC = 8 GB RAM

PROGRAM STORAGE is also smaller compared to PC.

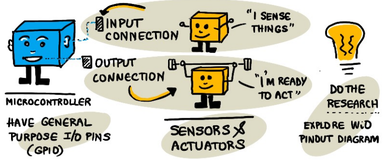
W.I.O = 4MB STORAGE

PC = 500 GB STORAGE

DO THE RESEARCH

HOW MUCH RAM AND STORAGE DOES YOUR COMPUTER HAVE? HOW DOES IT COMPARE TO W.I.O?

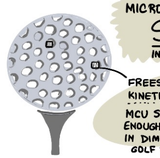
MICROCONTROLLERS: INPUT/OUTPUT



HAVE GENERAL PURPOSE I/O PINS (GPIO)

SENSORS & ACTUATORS

MICROCONTROLLERS: PHYSICAL SIZE



MICROCONTROLLERS ARE SMALL IN PHYSICAL SIZE

FREESCALE KINETIS KL03 MCU SMALL ENOUGH TO FIT IN DIMPLE OF GOLF BALLS

1.6mm × 2mm × 1mm

PC CPU 136 mm × 145 mm × 103 mm

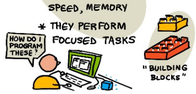
W.I.O 72 mm × 57 mm × 12 mm

FRAMEWORKS & OPERATING SYSTEMS

SEE ARDUINO FOR EXAMPLE

MICROCONTROLLERS DON'T RUN A TRADITIONAL OPERATING SYSTEM...

- THEY HAVE LOW SPEED, MEMORY
- THEY PERFORM FOCUSED TASKS



USE FRAMEWORKS

- USE TOOLS TO BUILD CODE IN A WAY THAT WILL RUN ON TARGET MICROCONTROLLER
- USE APIs TO TALK TO PERIPHERALS
- MANUFACTURERS SUPPORT STANDAR 'FRAMEWORKS' = RECIPES THAT DEVS USE TO RUN CODE ACROSS DIFFERENT MICROCONTROLLER PLATFORMS.

REAL TIME OPERATING SYSTEMS

USE A REAL TIME OPERATING SYSTEM

DESIGNED TO HANDLE REAL-TIME SEND/RECEIVE MESSAGE TASKS

RTOS → LIGHTWEIGHT CORE FEATURES

MULTITHREADED

RUN MULTIPLE BLOCKS OF CODE IN PARALLEL, ON A SINGLE OR MULTIPLE CORES.

MCU INTERNET APP

NETWORKING

COMMUNICATE SECURELY OVER THE INTERNET

GUI COMPONENTS FOR SCREENS

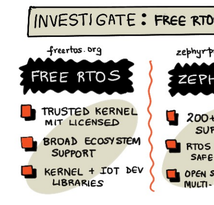
INVESTIGATE: AZURE RTOS

WHAT IS AZURE RTOS?

AN EMBEDDED DEVELOPMENT SUITE WITH

- POWERFUL RTOS
- RELIABLE, PERFORMANT
- SUPPORTS POPULAR 32bit MICROCONTROLLERS

ONE EXAMPLE SUITE



INVESTIGATE: FREE RTOS, ZEPHYR

freertos.org

FREE RTOS

- TRUSTED KERNEL MIT LICENSED
- BROAD ECOSYSTEM SUPPORT
- KERNEL + IOT DEV LIBRARIES

zephyrproject.org

ZEPHYR

- 200+ BOARDS SUPPORTED
- RTOS FOR SECURE, SAFE IOT APPS
- OPEN SOURCE WITH MULTI-PROTOCOL SUPPORT

DO YOUR RESEARCH

Explore and compare RTOS options for IOT

ARDUINO MICROCONTROLLER FRAMEWORK

BUY BOARDS FROM ARDUINO OR FROM OTHER MANUFACTURERS

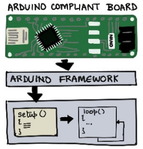
CODE USING C/C++

ARDUINO IS AN OPEN SOURCE ELECTRONICS PLATFORM COMBINING HARDWARE & SOFTWARE

CODE USING THE ARDUINO FRAMEWORK

- COMPILED CODE IS SMALL IN SIZE
- RUNS FAST EVEN ON RESOURCE-LIMITED DEVICE PLATFORMS

ARDUINO: CORE SETUP



2 CORE FUNCTIONS setup() loop()

WHEN BOARD POWERS UP

- RUNS setup() ONCE
- THEN RUNS loop() CONTINUOUSLY (till power off)

ARCHITECTURE: EVENT LOOP

SETUP IS FOR ONE-TIME INITIALIZATION CODE
→ connect to Wifi, Cloud services etc.

LOOP IS FOR PROCESSING CODE - ADD DELAY TO SAVE POWER (sleep/wake cycle)
→ Sensor read
Send/receive messages

PROGRAM ARCHITECTURE

CALLED 'EVENT LOOP' OR 'MESSAGE LOOP'

loop() LISTENS FOR

- MESSAGES FROM UI (button clicks, keyboard...)
- MESSAGES FROM NETWORK (actuator requests)

ARDUINO: STANDARD LIBRARIES

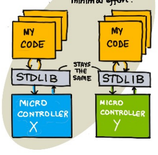
ARDUINO PROVIDES STANDARD LIBRARIES FOR INTERACTING WITH I/O PINS AND MICRO-CONTROLLERS

EXPOSES CONSISTENT API ACROSS DIVERSE MCU-SPECIFIC IMPLEMENTATION

delay() PAUSE PROGRAM FOR GIVEN PERIOD OF TIME

digitalRead() READ VALUE ON I/O PIN (HIGH OR LOW)

Code can be re-compiled for new compliant hardware with minimal effort!



INVESTIGATE: W.I.O TERMINAL

RE-READ YOUR CODE FROM LAST LESSON

W.I.O TERMINAL

IF YOU USED A W.I.O TERMINAL

TRY TO OBSERVE EVENT LOOP IN ACTION

1. FIND setup() / loop() FUNCTIONS IN CODE
2. MONITOR SERIAL OUTPUT
   → loop() CALLED REPEATEDLY
3. WRITE TO SERIAL OUT IN setup()
   → IS THIS CALLED ONLY ONCE (reboot)

SINGLE BOARD COMPUTERS

RASPBERRY PI FOUNDATION

2009 UK CHARITY

MISSION

PROMOTE THE STUDY OF COMPUTER SCIENCE AT SCHOOL LEVELS...

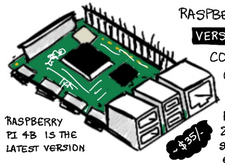
RASPBERRY PI SINGLE BOARD COMPUTER

3 VARIANTS

- FULL VERSION
- PI "ZERO"
- COMPUTE MODULE THAT CAN BE BUILT INTO YOUR IOT DEVICE

RASPBERRY PI 4

COMPARABLE TO DESKTOP PC/MAC — BUT CHEAPER



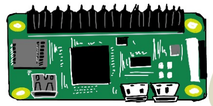
RASPBERRY PI 4B IS THE LATEST VERSION

$35!

RASPBERRY PI 4 IS A FULL VERSION SINGLE BOARD COMPUTER WITH QUAD-CORE CPU, 2, 4 or 8 GB of RAM, Wi-Fi, Gigabit Ethernet, 2 HDMI ports, 2 USB 2.0 ports, 2 USB 3.0 ports, 40 GPIO pins, SD card slot, camera connector etc.

RASPBERRY PI ZERO

BY COMPARISON IS SMALLER AND HAS LOWER POWER THAN Pi-4B



1 CORE 1GHz CPU
512 MB RAM
1 HDMI PORT
1 MICRO USB PORT
40 GPIO PINS
SD CARD SLOT
CAMERA CONNECTOR

ALL Pi VARIANTS RUN RASPBERRY PI OS = VERSION OF DEBIAN LINUX

- LITE VERSION "HEADLESS"
- FULL VERSION DESKTOP ENV

BOTH Pi-ZERO AND Pi-4B USE ARM PROCESSORS = USED IN MOST MOBILE PHONES, MICROSOFT SURFACE X &...

PROGRAMMING: SINGLE BOARD COMPUTERS

WANT TO PROGRAM SINGLE BOARD COMPUTERS?

WHAT PROGRAMMING LANG DO YOU USE? ARE THEY SUPPORTED ON LINUX?

THERE IS A WIDE RANGE OF PROGRAMMING LANGUAGES, TOOLS AND FRAMEWORKS FOR SBC — BECAUSE THEY RUN A FULL OPERATING SYSTEM

Most languages have libraries to access GPIO pins and send, receive data.

MOST COMMON LANGUAGE FOR IOT APPS = PYTHON!

LARGE ECOSYSTEM OF HARDWARE TO EXTEND PI

'HATS' = SIT ON PI, CONNECT TO 40 GPIO PINS

USE OF SINGLE BOARD COMPUTERS

SINGLE BOARD COMPUTERS ARE USED FOR BOTH DEV KITS AND PROFESSIONAL DEPLOYMENTS

USE CASES

- CONTROL HARDWARE
- RUN COMPLEX TASKS (e.g. MACHINE LEARNING MODELS)

RASPBERRY PI COMPUTE MODULE 4 designed for those building custom PCB

COMPUTE MODULE PROVIDES A WAY TO MOVE PROTOTYPES TO PRODUCTION

ALL THE POWER OF R-Pi 4 (BUT) IN A COMPACT AND CHEAPER FORM FACTOR

WHAT'S NEXT?

INTERACT WITH THE PHYSICAL WORLD USING SENSORS AND ACTUATORS

PROJECT TIME!

- GATHER DATA
- SEND FEEDBACK
- BUILD NIGHTLIGHT

A DEEP DIVE!

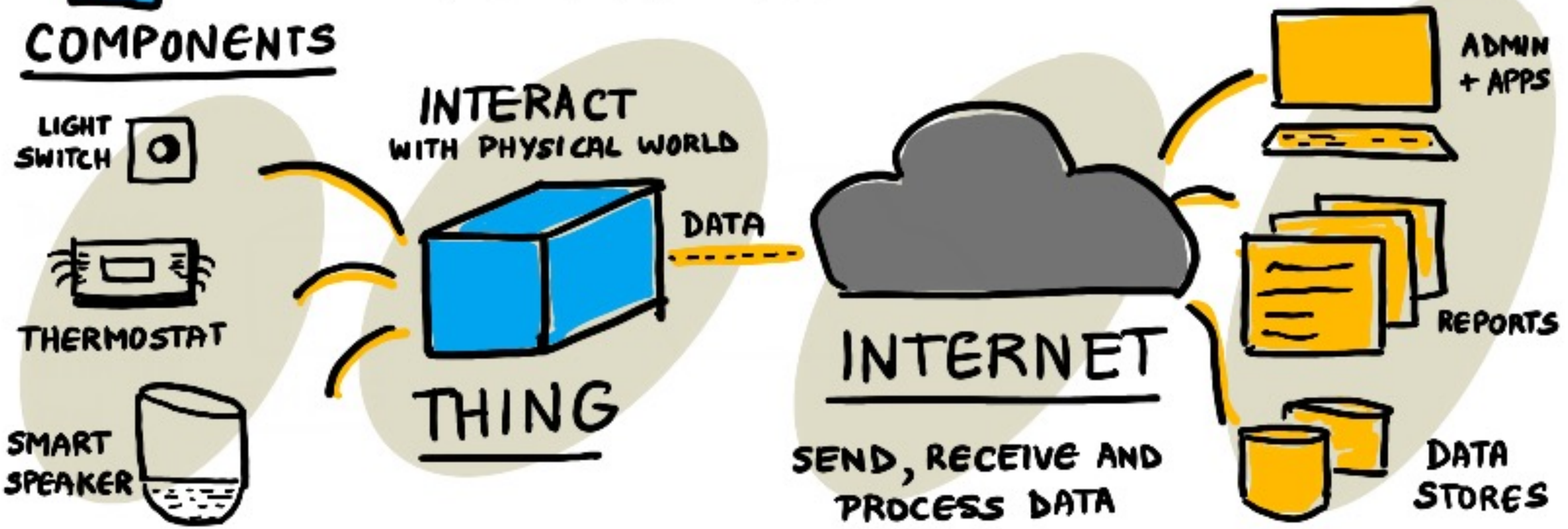
- COMPONENTS
  OF IOT APPLICATIONS
- MICRO-CONTROLLERS
  A DEEPER DIVE
- SINGLE BOARD COMPUTERS
  A DEEPER DIVE

COMPONENTS OF IOT APPS

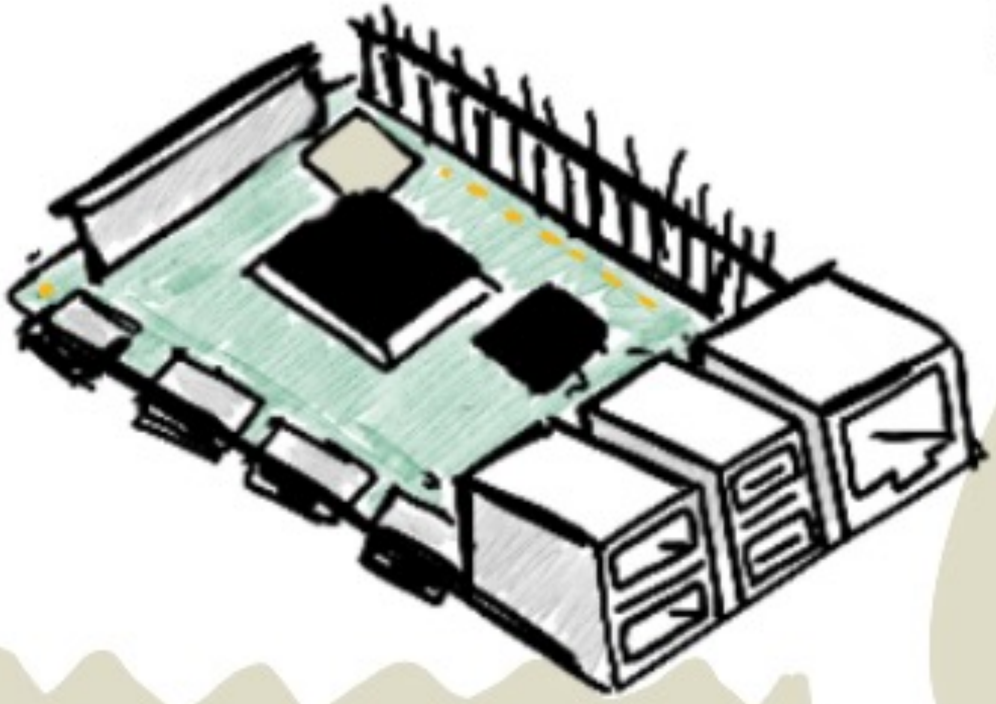
2 COMPONENTS

THE INTERNET = APPS THAT RECEIVE/SEND DATA

THE THING = DEVICES THAT SENSE/ACT REAL WORLD



THE THING



- LOW POWER
- LOW COST
- LOW SPEED

COMPUTERS

Run for long periods
Gather data (sensors)
Take actions (actuators)

Example

Microcontroller

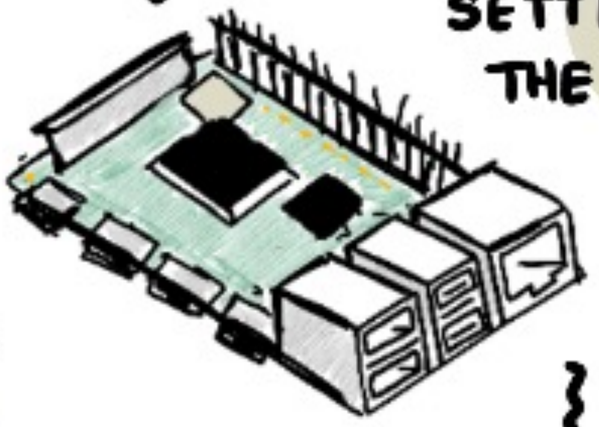
- RAM in kB
- SPEED in MHz

A DEVICE THAT CAN INTERACT WITH PHYSICAL WORLD

EX: A THERMOSTAT!

TEMPERATURE SENSOR

TEMPERATURE DETECTED IS LOWER THAN CONTROL SETTING. ACTUATOR TURNS THE HEATER ON TO WARM THE ROOM.



DESIRED TEMPERATURE (CONTROL) INPUT

HEATER (CONTROLLED FROM ACTUATOR)

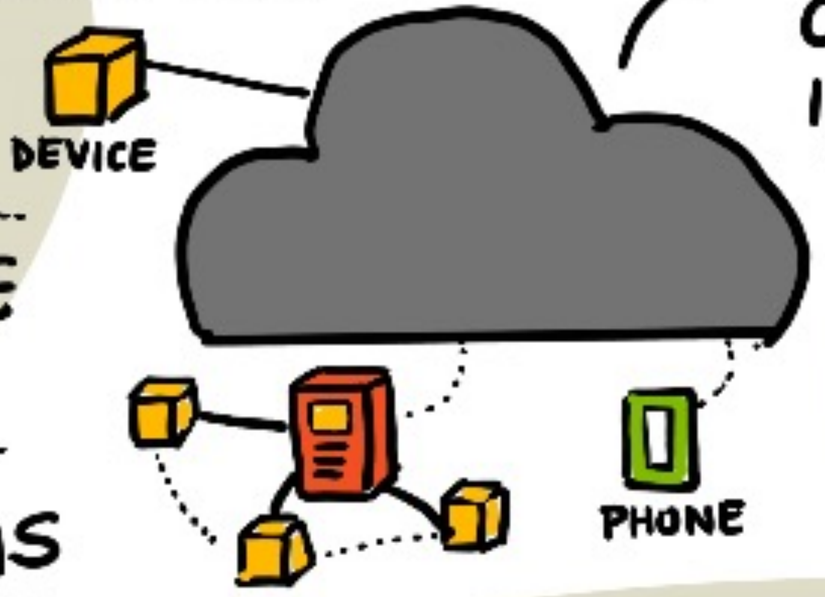
DO THE RESEARCH

WHAT OTHER SYSTEMS ARE AROUND YOU THAT READ SENSOR DATA & USE IT TO MAKE DECISIONS

THE INTERNET

CONSISTS OF APPS THAT

- PROCESS DATA
- SEND/RECEIVE MESSAGES
- TAKE DECISIONS ON REQUESTS TO SEND TO ACTUATORS



TYPICAL SETUP

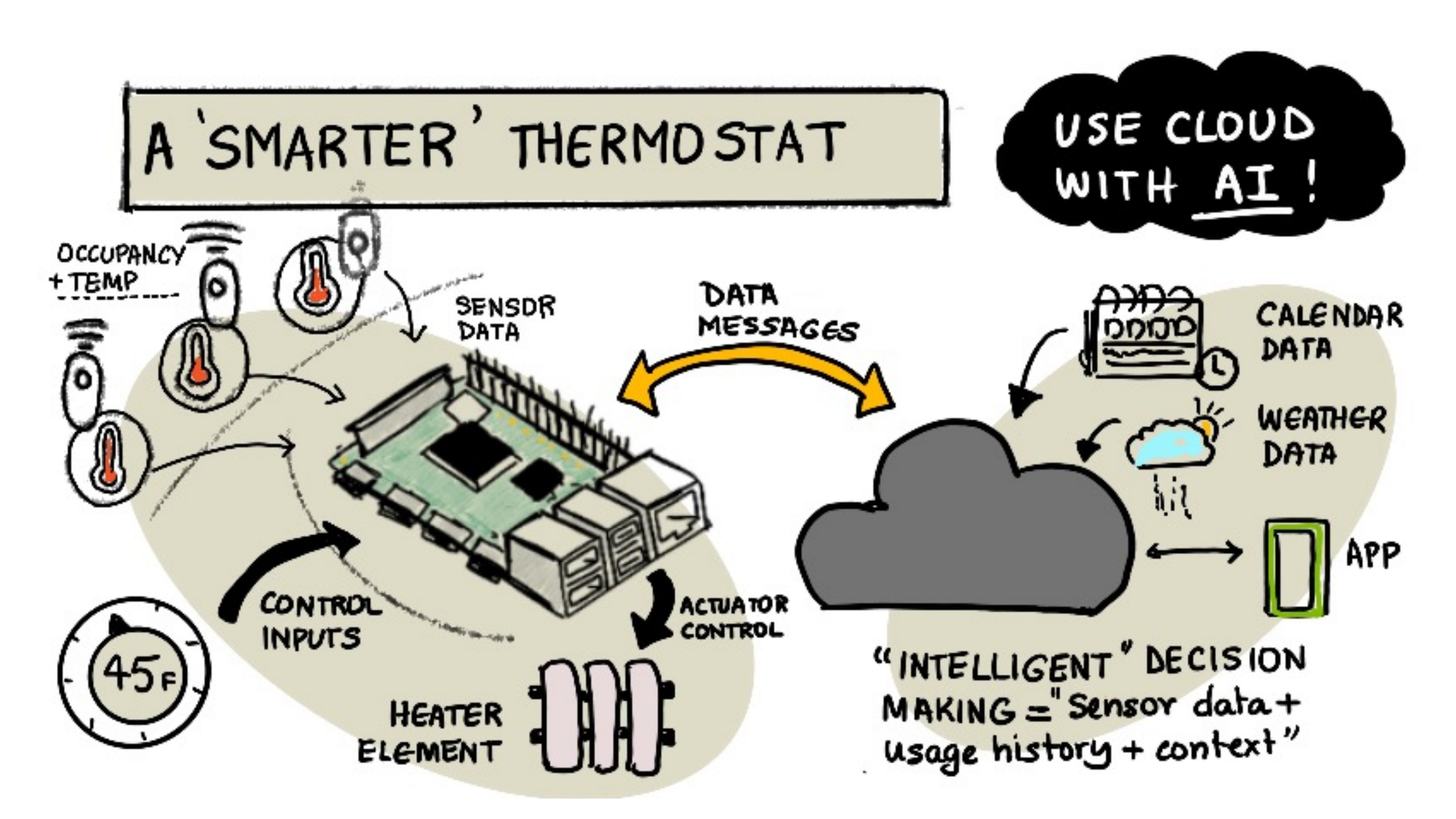
CLOUD SERVICE INTERMEDIARY

- HANDLE SECURITY
- INTERACT WITH SENSORS ON ONE SIDE
- CONNECT TO APPS ON THE OTHER

MESH NETWORKS

PEER TO PEER ROUTES

DEVICES CONNECT TO EACH OTHER (BT, WiFi) AND TO 'CONNECTED' HUB


A 'SMARTER' THERMOSTAT
USE CLOUD WITH AI!
OCCUPANCY + TEMP
SENSOR DATA
DATA MESSAGES
CALENDAR DATA
WEATHER DATA
APP
CONTROL INPUTS
ACTUATOR CONTROL
45F
HEATER ELEMENT
"INTELLIGENT" DECISION MAKING = "Sensor data + usage history + context"

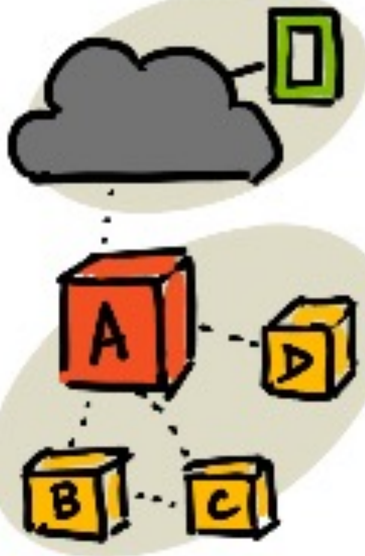
DO THE RESEARCH

WHAT OTHER DATA COULD MAKE THERMOSTAT EVEN SMARTER?

IOT ON THE EDGE

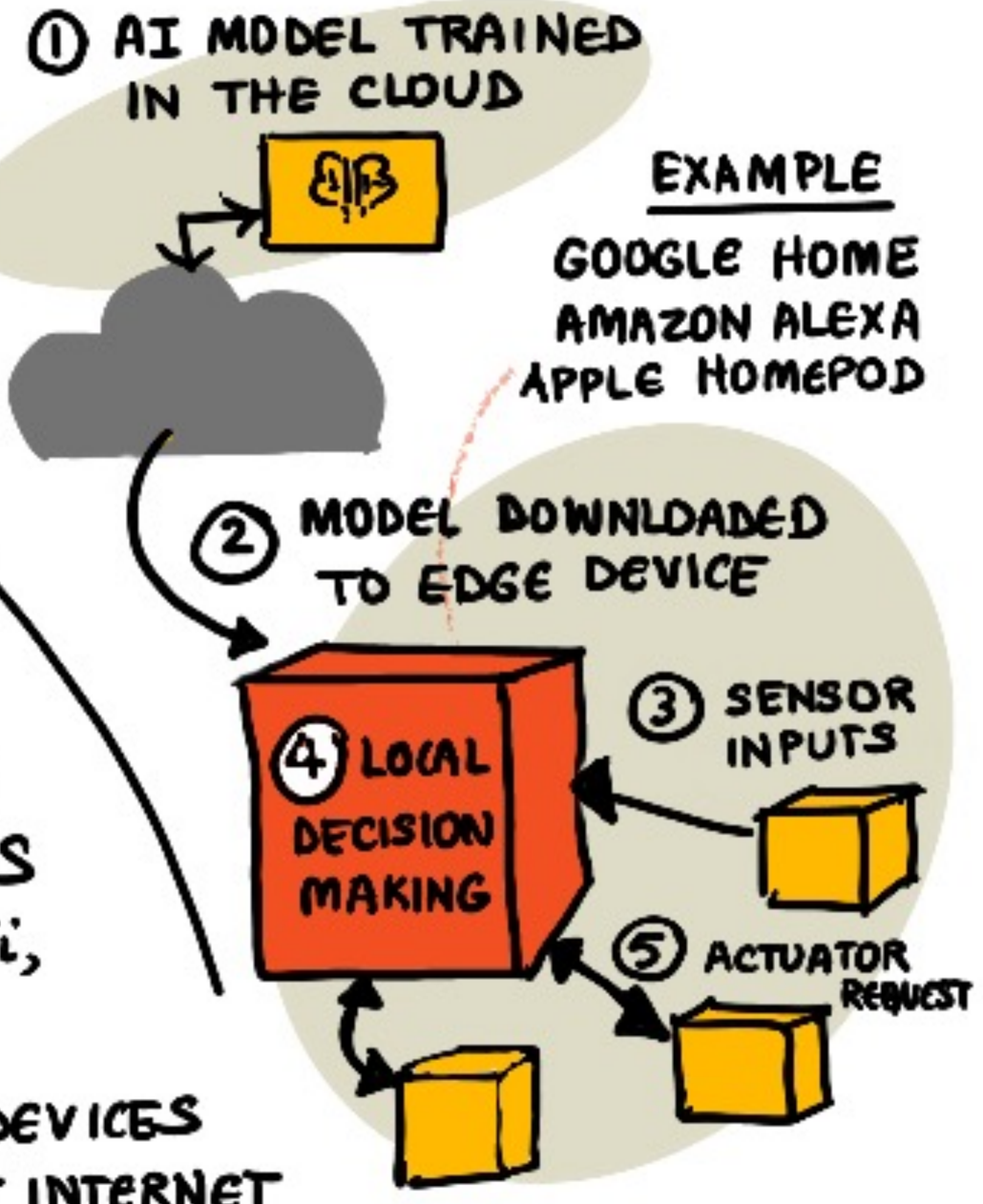
ALL IOT DEVICES DON'T HAVE TO CONNECT TO THE INTERNET

EDGE DEVICES ARE 'GATEWAYS' CAPABLE OF PROCESSING DATA LOCALLY. IOT DEVICES CAN CONNECT TO EDGE DEVICES OVER LOCAL NETWORKS (WiFi, BLUETOOTH)



A = GATEWAY (EDGE)

B, C, D = IOT DEVICES WITH NO DIRECT INTERNET

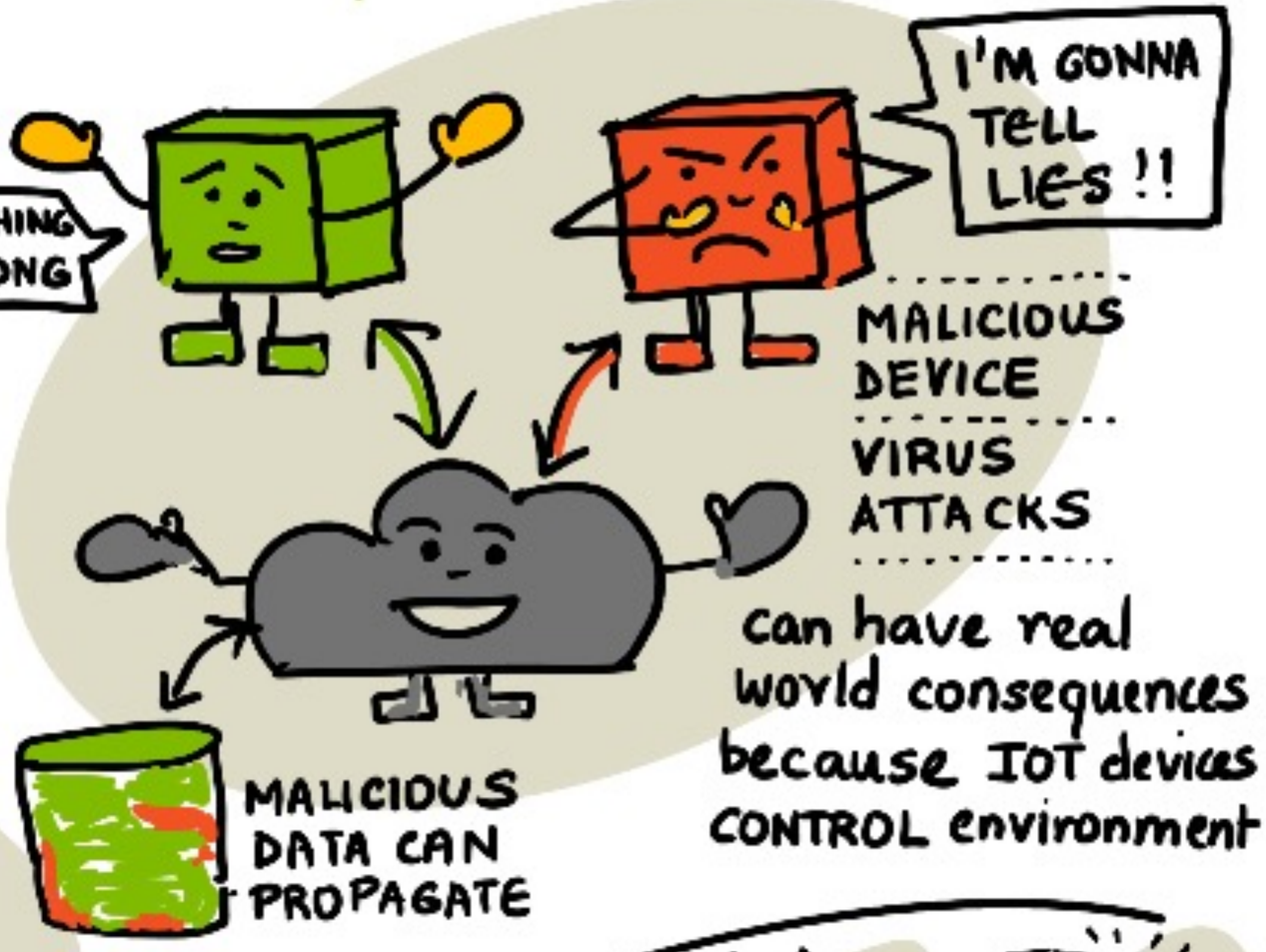


IOT SECURITY

A POPULAR JOKE ON IOT IMPLIES SECURITY DOES NOT EXIST—

IN REALITY

IOT DEVICES CONNECT TO THE CLOUD - AND ARE ONLY AS SECURE AS THE CLOUD (AND NETWORK)



can have real world consequences because IOT devices CONTROL environment

EXAMPLES OF ATTACKS:

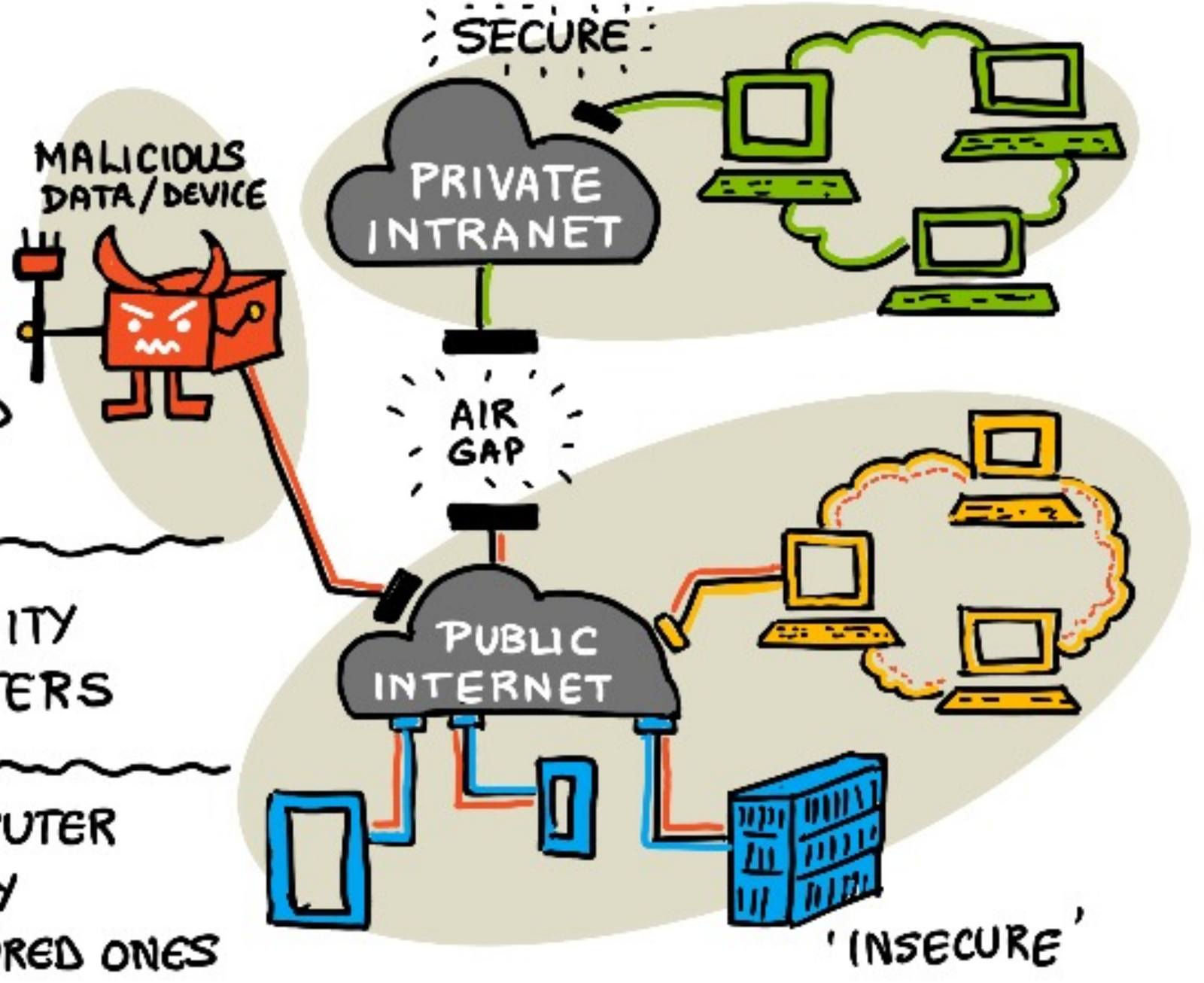
AIR GAPPING

aka

- ✓ AIR WALL
- ✓ AIR GAP
- ✓ DISCONNECTED NETWORK

IS A NETWORK SECURITY MEASURE FOR COMPUTERS

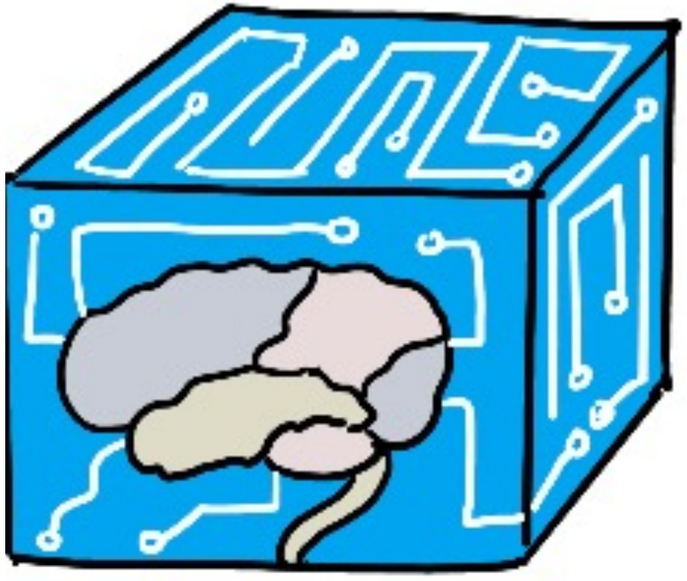
WHERE A SECURE COMPUTER NETWORK IS PHYSICALLY ISOLATED FROM UNSECURED ONES



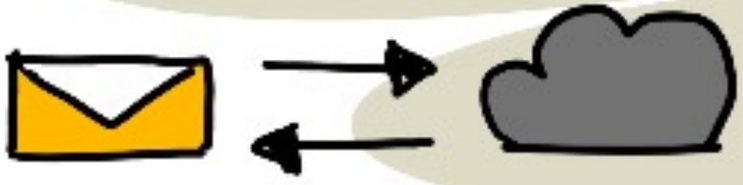
DO THE RESEARCH

WHAT OTHER SECURITY CHALLENGES OR SCENARIOS CAN YOU THINK OF FOR IOT SYSTEMS AROUND YOU?

MICROCONTROLLERS : CPU



THE CPU (CENTRAL PROCESSING UNIT) IS THE **BRAIN** OF THE MICROCONTROLLER.



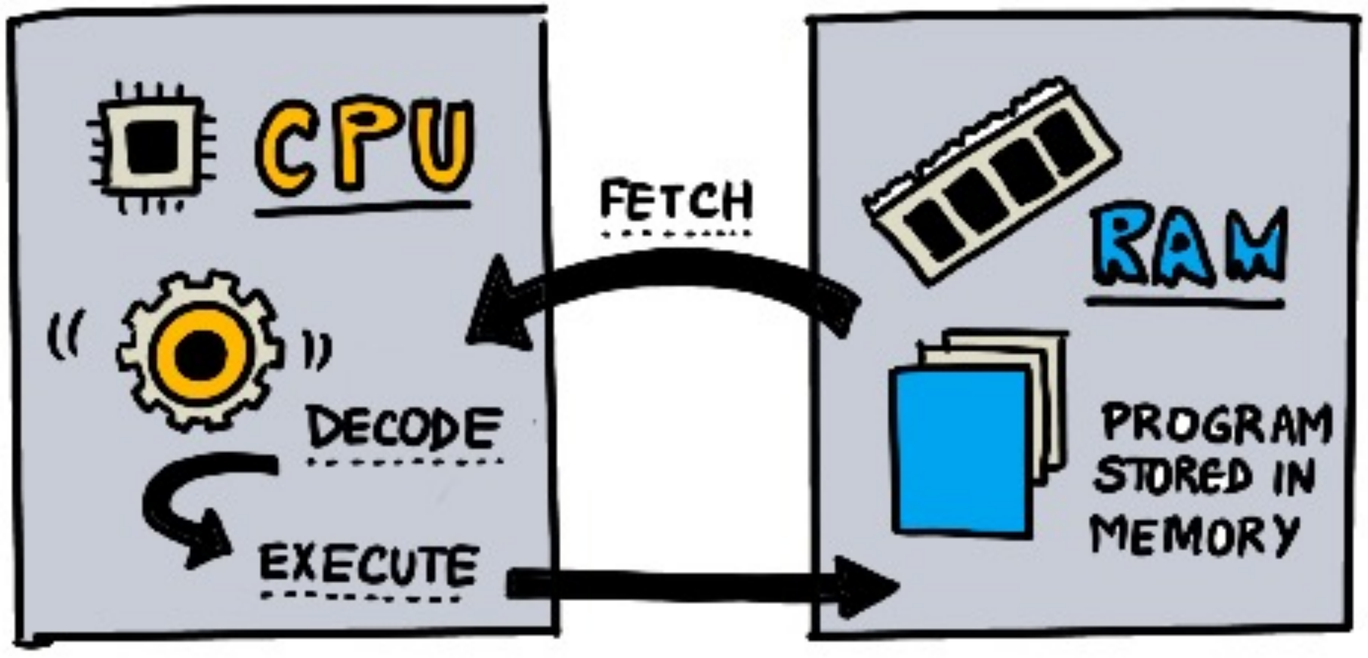
SENDS / RECEIVES MESSAGES

EXECUTES ONE INSTRUCTION PER CLOCK TICK → MILLIONS OR BILLIONS OF TICKS PER SEC

HIGHER THE **SPEED** THE MORE INSTRUCTIONS RUN/SEC

MICROCONTROLLERS : CPU

FETCH - DECODE - EXECUTE



ON CLOCK TICK

CPU FETCHES INSTRUCTION,

DECODES IT

THEN EXECUTES THE TASK

SOME INSTRUCTIONS WILL TAKE MULTIPLE CLOCK TICKS ...

INVESTIGATE WiO TERMINAL

DO THE RESEARCH

EVERY CLOCK TICK DRAWS POWER, GENERATES HEAT!

MICROCONTROLLERS RUN AT LOWER SPEEDS OR SWITCH TO LOW POWER CORES TO REDUCE OVERHEATING...

MICROCONTROLLERS : MEMORY

THERE ARE 2 TYPES OF MEMORY

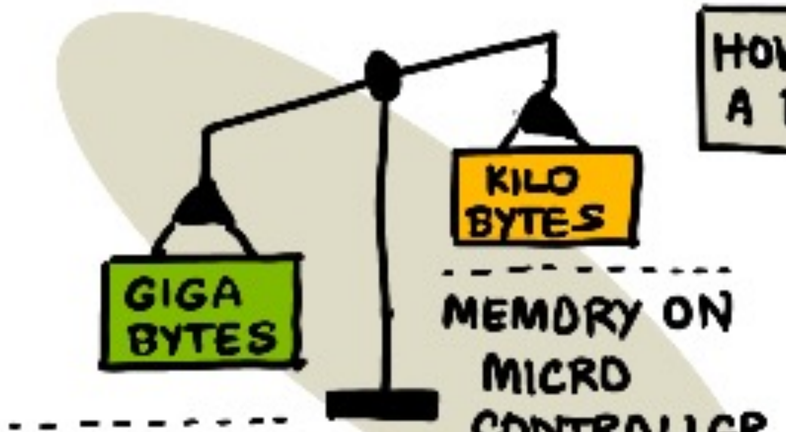
PROGRAM MEMORY

- STORES YOUR CODE (PROGRAM)
- PERSISTS WHEN THERE IS NO POWER

RANDOM ACCESS MEMORY

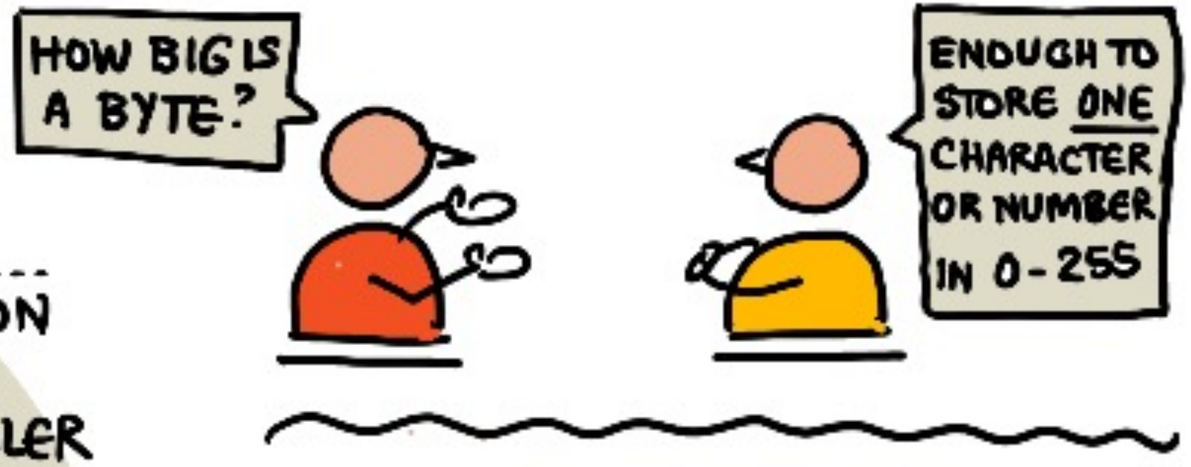
- USED TO RUN YOUR CODE WHEN POWERED
- RESETS WHEN THERE IS NO POWER

MICROCONTROLLERS : MEMORY



MEMORY ON PC

MEMORY ON MICRO CONTROLLER



WiO TERMINAL = 192 kB RAM

BASE PC = 8 GB RAM

PROGRAM STORAGE is also smaller compared to PC.

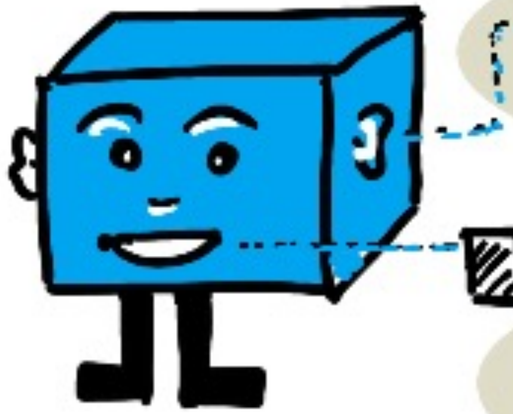
WiO = 4MB STORAGE

PC = 500 GB STORAGE

DO THE RESEARCH

HOW MUCH RAM AND STORAGE DOES YOUR COMPUTER HAVE? HOW DOES IT COMPARE TO W10?

MICROCONTROLLERS: INPUT/OUTPUT



INPUT CONNECTION

OUTPUT CONNECTION

"I SENSE THINGS"

"I'M READY TO ACT"

DO THE RESEARCH

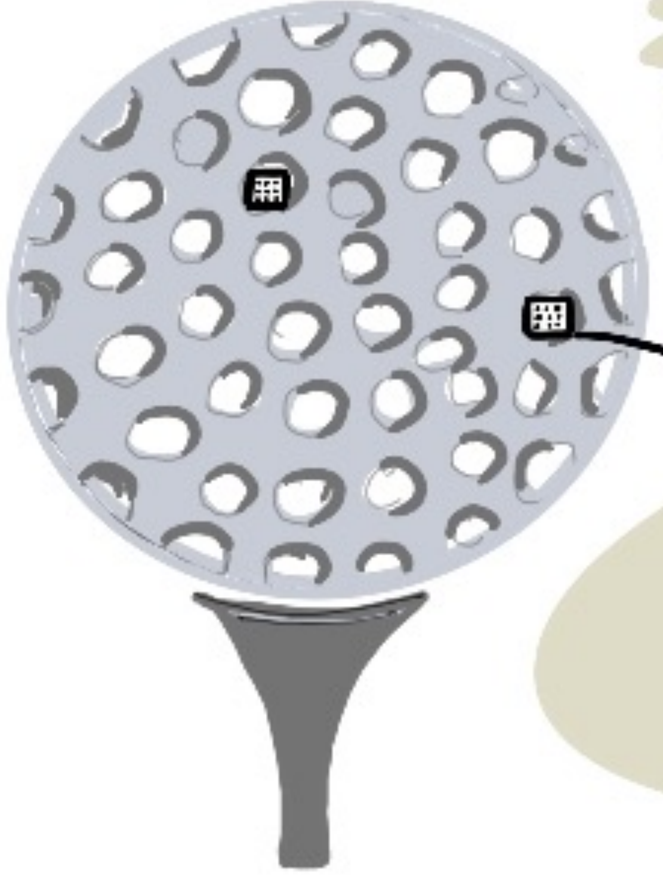
MICROCONTROLLER

HAVE GENERAL PURPOSE I/O PINS (GPIO)

SENSORS & ACTUATORS

EXPLORE W:D PINOUT DIAGRAM

MICROCONTROLLERS: PHYSICAL SIZE



MICROCONTROLLERS ARE

SMALL

IN PHYSICAL SIZE

FREESCALE KINETIS KL03

1.6mm x 2mm x 1mm

MCU SMALL ENOUGH TO FIT IN DIMPLE OF GOLF BALLS

136 mm x 145 mm x 103 mm

72 mm x 57 mm x 12 mm

FRAMEWORKS & OPERATING SYSTEMS

SEE ARDUINO FOR EXAMPLE

MICROCONTROLLERS DON'T RUN A TRADITIONAL OPERATING SYSTEM..

* THEY HAVE LOW SPEED, MEMORY
* THEY PERFORM FOCUSED TASKS

"BUILDING BLOCKS"

USE FRAMEWORKS

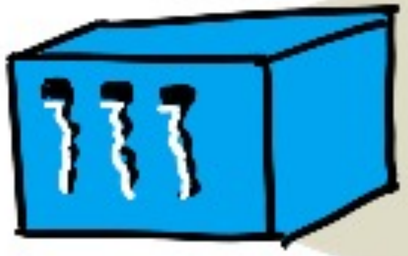
* USE TOOLS TO BUILD CODE IN A WAY THAT WILL RUN ON TARGET MICROCONTROLLER
* USE APIs TO TALK TO PERIPHERALS
* MANUFACTURERS SUPPORT STANDAR 'FRAMEWORKS' = RECIPES THAT DEVS USE TO RUN CODE ACROSS DIFFERENT MICROCONTROLLER PLATFORMS.

REAL TIME OPERATING SYSTEMS

USE A REAL TIME OPERATING SYSTEM

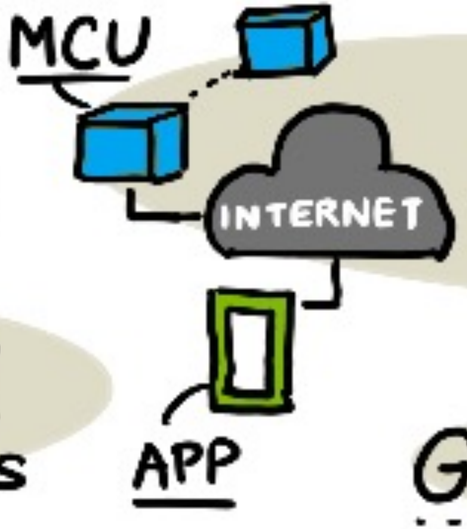
DESIGNED TO HANDLE REAL-TIME SEND/RECEIVE MESSAGE TASKS

RTOS → LIGHTWEIGHT CORE FEATURES



MULTITHREADED

RUN MULTIPLE BLOCKS OF CODE IN PARALLEL, ON A SINGLE OR MULTIPLE CORES.

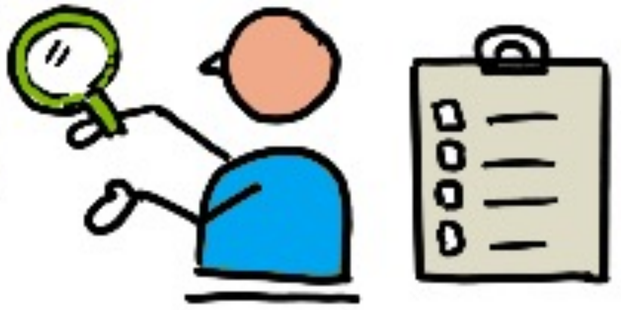


NETWORKING

COMMUNICATE SECURELY OVER THE INTERNET

GUI COMPONENTS FOR SCREENS

INVESTIGATE: AZURE RTOS



ONE EXAMPLE SUITE

AN EMBEDDED DEVELOPMENT SUITE WITH

- POWERFUL RTOS
- RELIABLE, PERFORMANT
- SUPPORTS POPULAR 32 bit MICROCONTROLLERS

THREADX — Real Time Multithreading

FILEX — High Performance File System

TRACEX — Analyse real-time events, behaviors

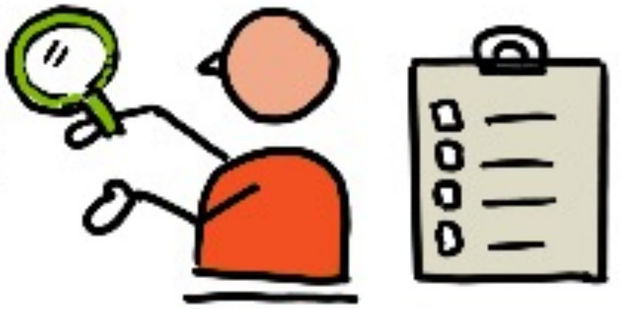
GUIX — GUI Library for runtime

GUIX STUDIO — + GUI Design environment

NETX — Piconet architecture network connectivity

USBX — USB host and device interface

INVESTIGATE: FREE RTOS, ZEPHYR



freertos.org

FREE RTOS

- TRUSTED KERNEL MIT LICENSED
- BROAD ECOSYSTEM SUPPORT
- KERNEL + IOT DEV LIBRARIES

zephyrproject.org

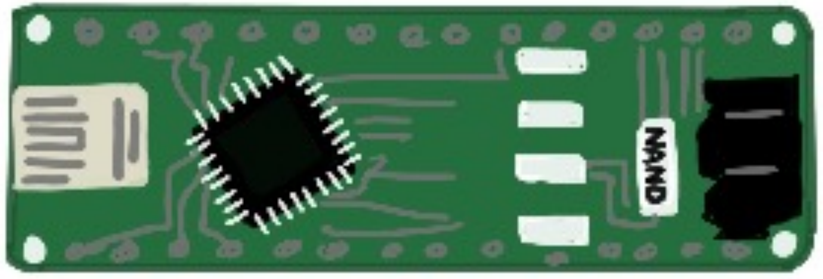
ZEPHYR

- 200+ BOARDS SUPPORTED
- RTOS FOR SECURE, SAFE IOT APPS
- OPEN SOURCE WITH MULTI-PROTOCOL SUPPORT

DO YOUR RESEARCH

Explore and compare RTOS options for IOT

ARDUINO MICROCONTROLLER FRAMEWORK



BUY BOARDS FROM ARDUINO (OR) FROM OTHER MANUFACTURERS

CODE USING C/C++

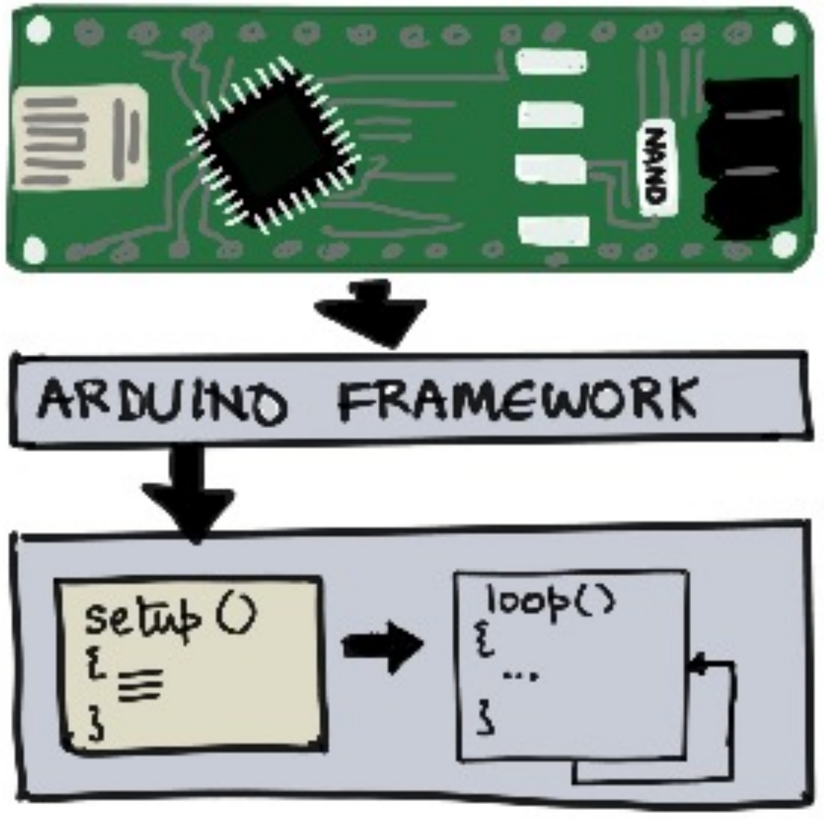
ARDUINO IS AN OPEN SOURCE ELECTRONICS PLATFORM COMBINING HARDWARE & SOFTWARE

CODE USING THE ARDUINO FRAMEWORK

- COMPILED CODE IS SMALL IN SIZE
- RUNS FAST EVEN ON RESOURCE-LIMITED DEVICE PLATFORMS

ARDUINO : CORE SETUP

ARDUINO COMPLIANT BOARD



2 CORE FUNCTIONS

setup()

loop()

WHEN BOARD POWERS UP

- RUNS setup() ONCE
- THEN RUNS loop() CONTINUOSLY (till power off)

ARCHITECTURE : EVENT LOOP

SETUP IS FOR ONE-TIME INITIALIZATION CODE

→ connect to Wifi, Cloud services etc.

LOOP IS FOR PROCESSING CODE – ADD DELAY TO SAVE POWER (sleep/wake cycle)

→ sensor read
send/receive messages

PROGRAM ARCHITECTURE

CALLED 'EVENT LOOP'
OR 'MESSAGE LOOP'

---

loop() LISTENS FOR

- MESSAGES FROM UI (button clicks, Keyboard...)
- MESSAGES FROM NETWORK (actuator requests)

ARDUINO : STANDARD LIBRARIES

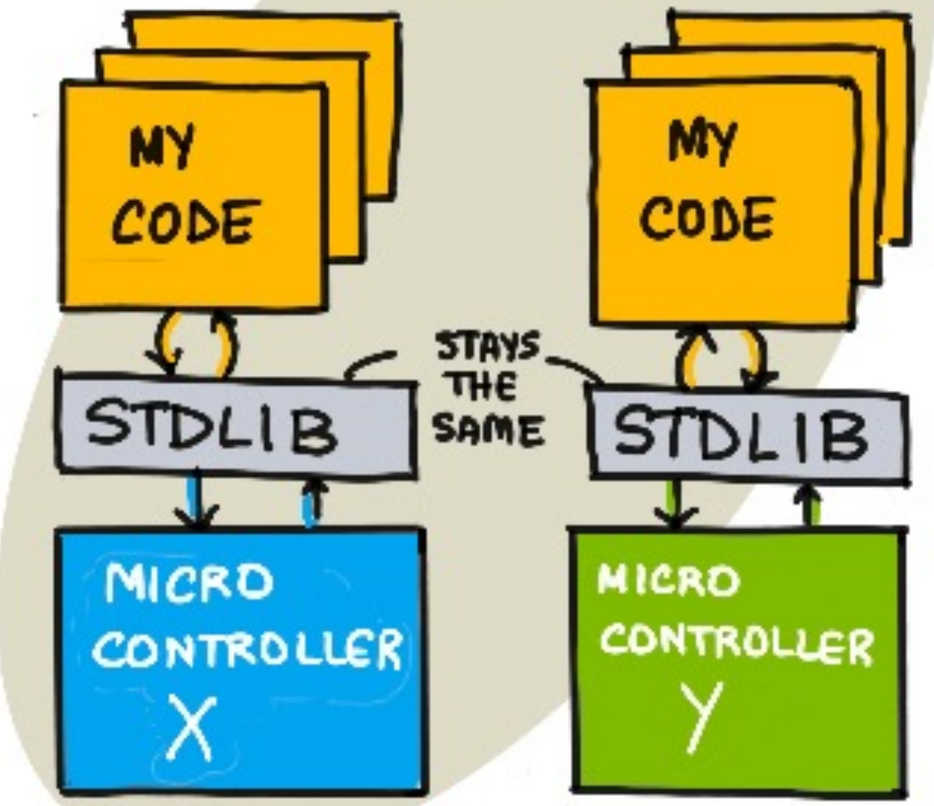
ARDUINO PROVIDES STANDARD LIBRARIES FOR INTERACTING WITH I/O PINS AND MICRO-CONTROLLERS

EXPOSES CONSISTENT API ACROSS DIVERSE MCU-SPECIFIC IMPLEMENTATION

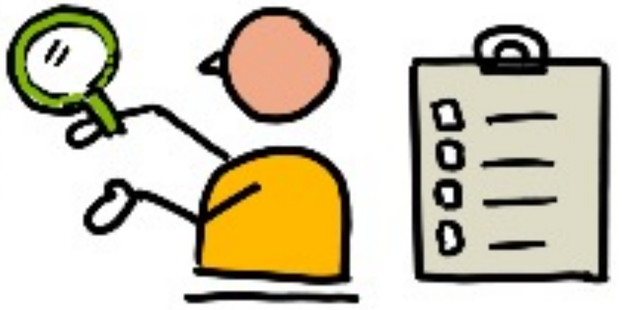
delay() — PAUSE PROGRAM FOR GIVEN PERIOD OF TIME

digitalRead() — READ VALUE ON I/O PIN (HIGH OR LOW)

Code can be recompiled for new compliant hardware with minimal effort!

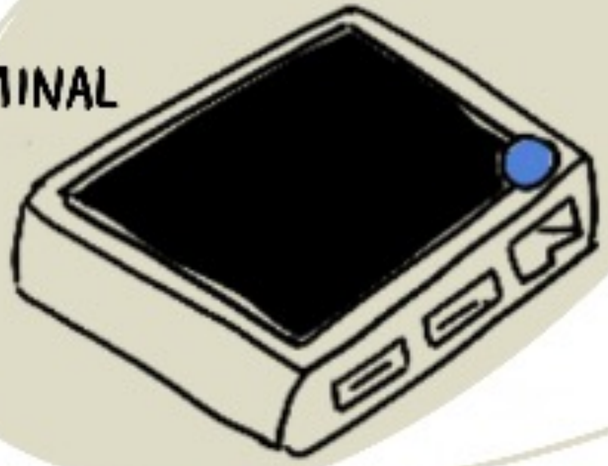


INVESTIGATE: WiO TERMINAL



RE-READ YOUR CODE FROM LAST LESSON

WiO TERMINAL

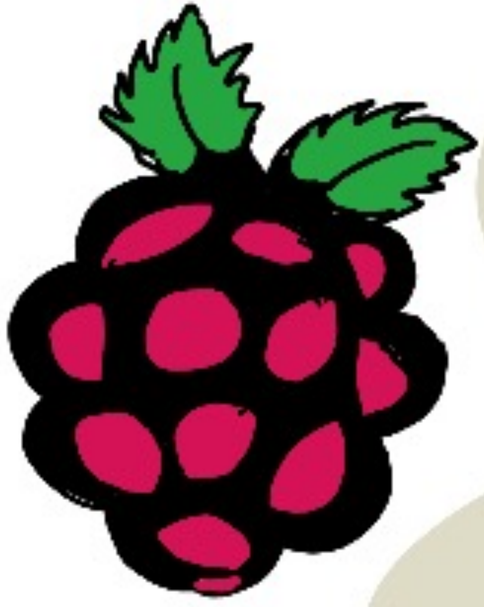


IF YOU USED A WiO TERMINAL

TRY TO OBSERVE EVENT LOOP IN ACTION

1. FIND SETUP() / LOOP() FUNCTIONS IN CODE
2. MONITOR SERIAL OUTPUT
   ↳ IS LOOP() CALLED REPEATEDLY
3. WRITE TO SERIAL OUT IN SETUP()
   ↳ IS THIS CALLED ONLY ONCE (on reboot)

SINGLE BOARD COMPUTERS



RASPBERRY PI FOUNDATION

2009 UK CHARITY

MISSION

PROMOTE THE STUDY OF COMPUTER SCIENCE AT SCHOOL LEVELS...

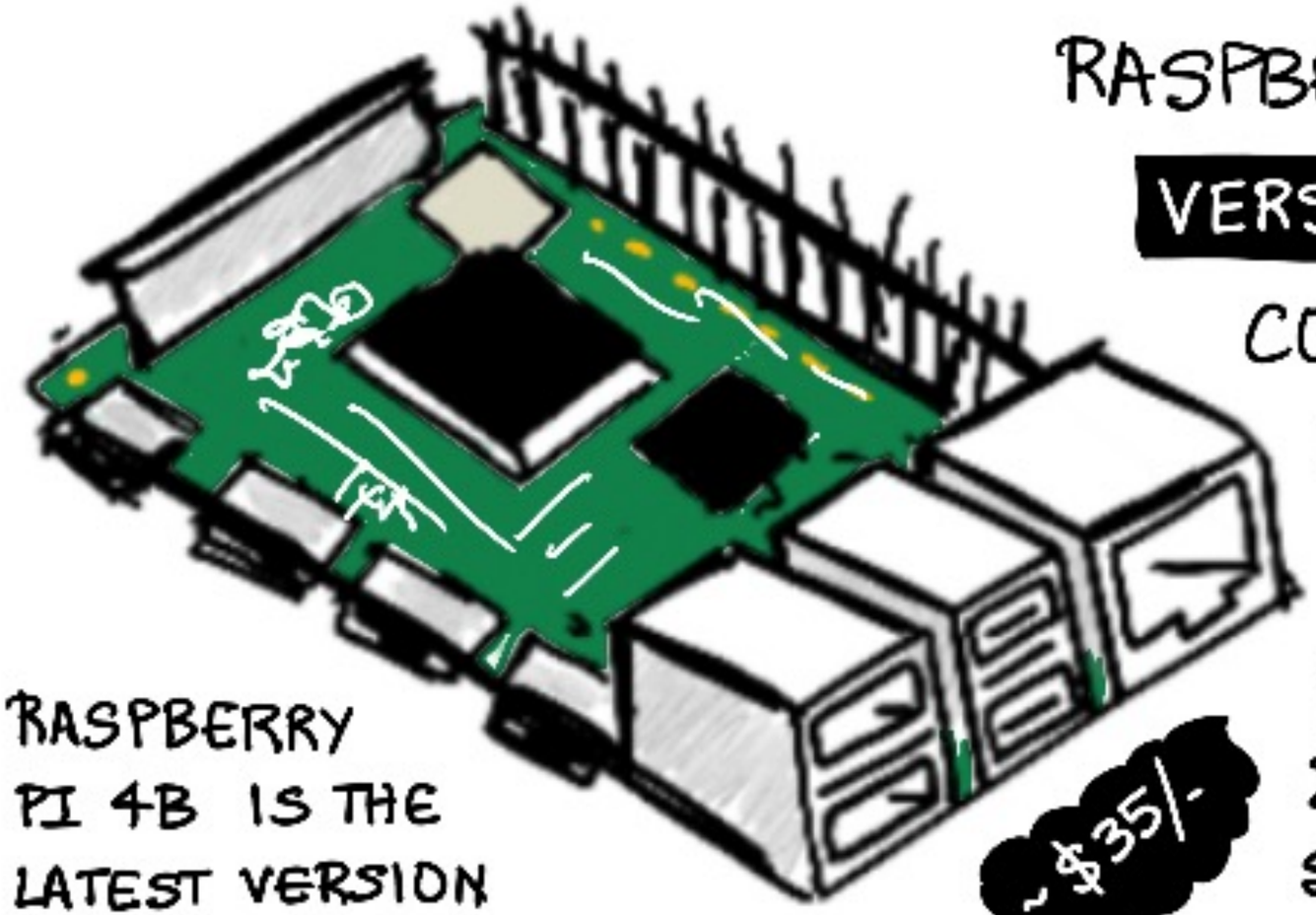
RASPBERRY PI SINGLE BOARD COMPUTER

3 VARIANTS

- FULL VERSION
- PI "ZERO"
- COMPUTE MODULE THAT CAN BE BUILT INTO YOUR IOT DEVICE

RASPBERRY PI 4

COMPARABLE TO DESKTOP PC/MAC — BUT CHEAPER



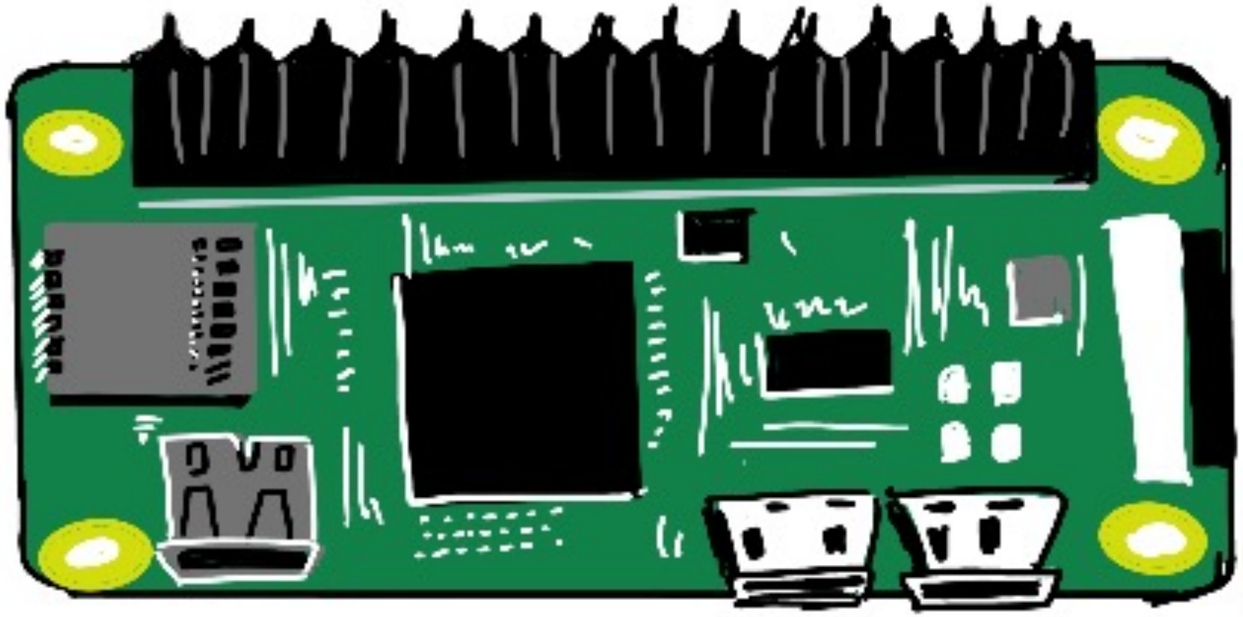
RASPBERRY PI 4B IS THE LATEST VERSION

~$35/-

RASPBERRY PI 4 IS A **FULL VERSION** SINGLE BOARD COMPUTER WITH **QUAD-CORE** CPU, **2, 4 or 8** GB of RAM, WiFi, Gigabit Ethernet, 2 HDMI ports, 2 USB 2.0 ports, 2 USB 3.0 ports, 40 GPIO pins, SD card slot, camera connector etc.

RASPBERRY PI ZERO

BY COMPARISON IS **SMALLER** AND HAS **LOWER POWER** THAN PI-4B



- 1 CORE 1GHZ CPU
- 512 MB RAM
- 1 HDMI PORT
- 1 MICRO USB PORT
- 40 GPIO PINS
- SD CARD SLOT
- CAMERA CONNECTOR

ALL PI VARIANTS RUN RASPBERRY PI OS – VERSION OF DEBIAN LINUX

- LITE VERSION – "HEADLESS"
- FULL VERSION – DESKTOP ENV

BOTH PI-ZERO AND PI-4B USE ARM PROCESSORS! = USED IN MOST MOBILE PHONES, MICROSOFT SURFACE X etc.

PROGRAMMING: SINGLE BOARD COMPUTERS

WANT TO PROGRAM SINGLE BOARD COMPUTERS?

WHAT PROGRAMMING LANG DO YOU USE? ARE THEY SUPPORTED ON LINUX?

THERE IS A WIDE RANGE OF PROGRAMMING LANGUAGES, TOOLS AND FRAMEWORKS FOR SBC — BECAUSE THEY RUN A FULL OPERATING SYSTEM

Most languages have libraries to access GPIO pins and send, receive data

LARGE ECOSYSTEM OF HARDWARE TO EXTEND PI

'HATS' = SIT ON PI, CONNECT TO 40 GPIO PINS

MOST COMMON LANGUAGE FOR IOT APPS = PYTHON!

USE OF SINGLE BOARD COMPUTERS

SINGLE BOARD COMPUTERS ARE USED FOR BOTH **DEV KITS** AND **PROFESSIONAL DEPLOYMENTS**

USE CASES

* CONTROL HARDWARE
* RUN COMPLEX TASKS (e.g MACHINE LEARNING MODELS)

RASPBERRY PI COMPUTE MODULE 4 — Designed for those building custom PCB

ALL THE POWER OF R-Pi 4 (BUT) IN A COMPACT AND CHEAPER FORM FACTOR

COMPUTE MODULE PROVIDES A WAY TO MOVE **PROTOTYPE** TO **PRODUCTION**

WHAT'S NEXT?

INTERACT

WITH THE PHYSICAL WORLD USING SENSORS AND ACTUATORS

PROJECT TIME!

- GATHER DATA
- SEND FEEDBACK
- BUILD NIGHTLIGHT


CONGRATULATIONS
CREATED BY
@ SKETCHTHEDOCS