

BASTELTRUPPE ARDUINO/C

PETER ZDANKIN
WS 2022/2023

HI I'M PETER!

PROFESSIONAL SCIENTIST (TVL E12)

PhD Student at Prof. Weis

Doing voluntary programming courses for years

Learned Embedded/C while working for Prof.
Schiele (don't tell him I didn't really know either
when I started working for him)

Here to teach you C using Arduino



How much there is about C



How much there is about Embedded

WHAT IS THIS COURSE ABOUT?

I love teaching C

Problem: C is hard to teach

Reason: You mostly just see some printf output

Idea: Teach C using Arduino for more interactivity and contact with C weirdnesses

(The C runtime on your laptops works too well..)

HISTORIC STUFF

C was created in 1972

Computers back then had few MHz, KB of RAM and little if any hard drives

The goal was to develop a programming language that is simpler to use than machine code/assembly, but still resource-efficient

Nowadays Computer have virtually limitless resources, few places *need* the performance from C

Where is C still relevant?

Operating Systems, drivers, graphics cards, embedded systems

WHAT'S AN EMBEDDED SYSTEM?

Various definitions exist

My definition: „It's a thing that you deploy somewhere and hope to never touch again”

Implications:

- This thing was designed to do a task
- It either has limited or limitless energy
- The cheapest hardware that allows to do the task is the best

THIS THING WAS DESIGNED TO DO A TASK

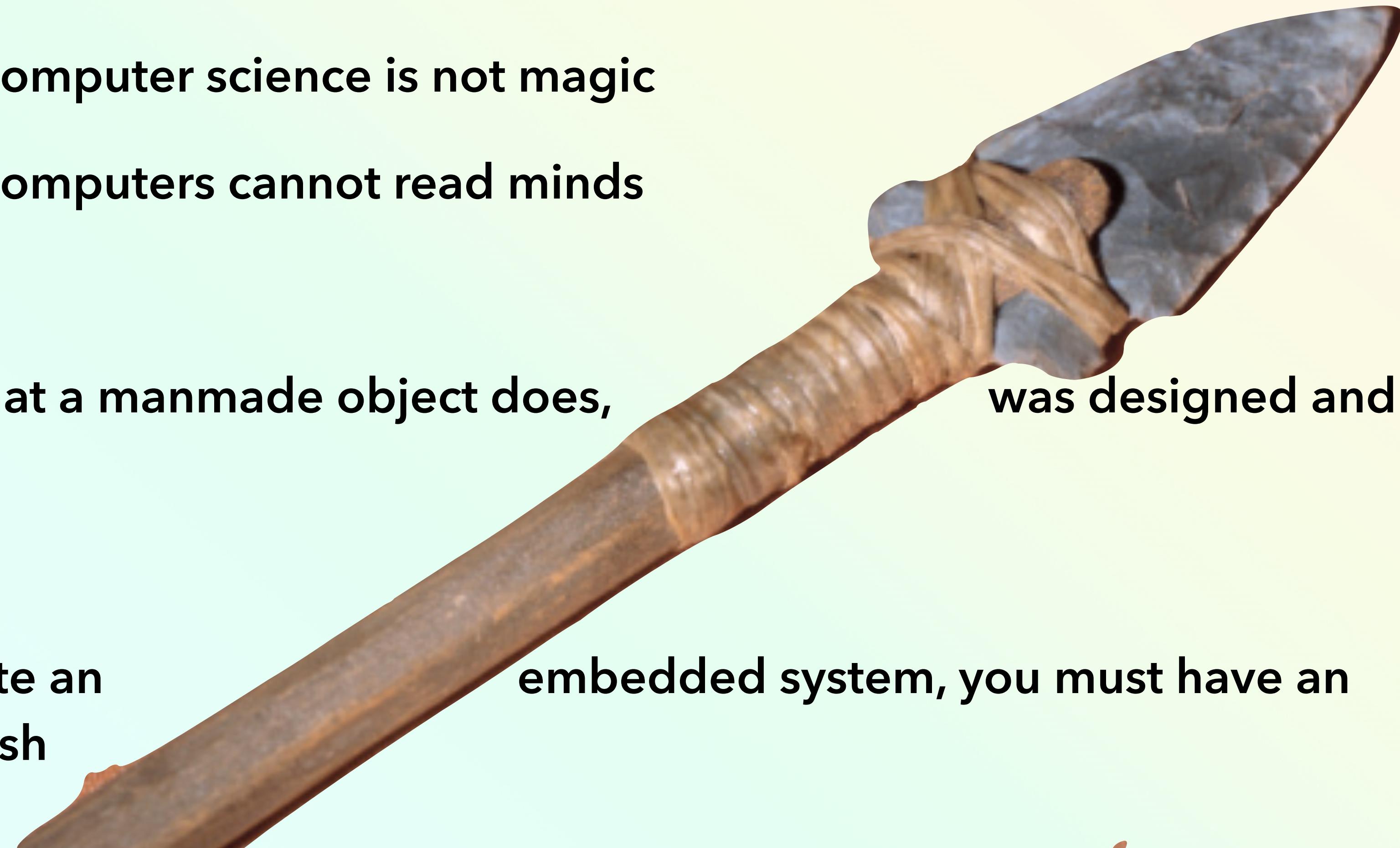
Contrary to popular belief, computer science is not magic

Contrary to popular belief, computers cannot read minds

Consequence: Everything that a manmade object does,
planned before it existed.

Before you head out to create an
idea of what it will accomplish

was designed and
embedded system, you must have an



LIMITED OR LIMITLESS ENERGY

Energy is not free (citation needed)

When a thing does a thing, it needs energy

You must determine if your embedded system can run from the electric power grid, or if it requires a battery

Electric power grid -> What if power surge/outage?

Battery power -> How long will your system last?



CHEAPEST HARDWARE IS THE BEST

Cheap in terms of

- Price per Unit
- Energy cost
- Resource efficiency



You don't want to waste, especially if the system is battery powered

IMPLICATIONS

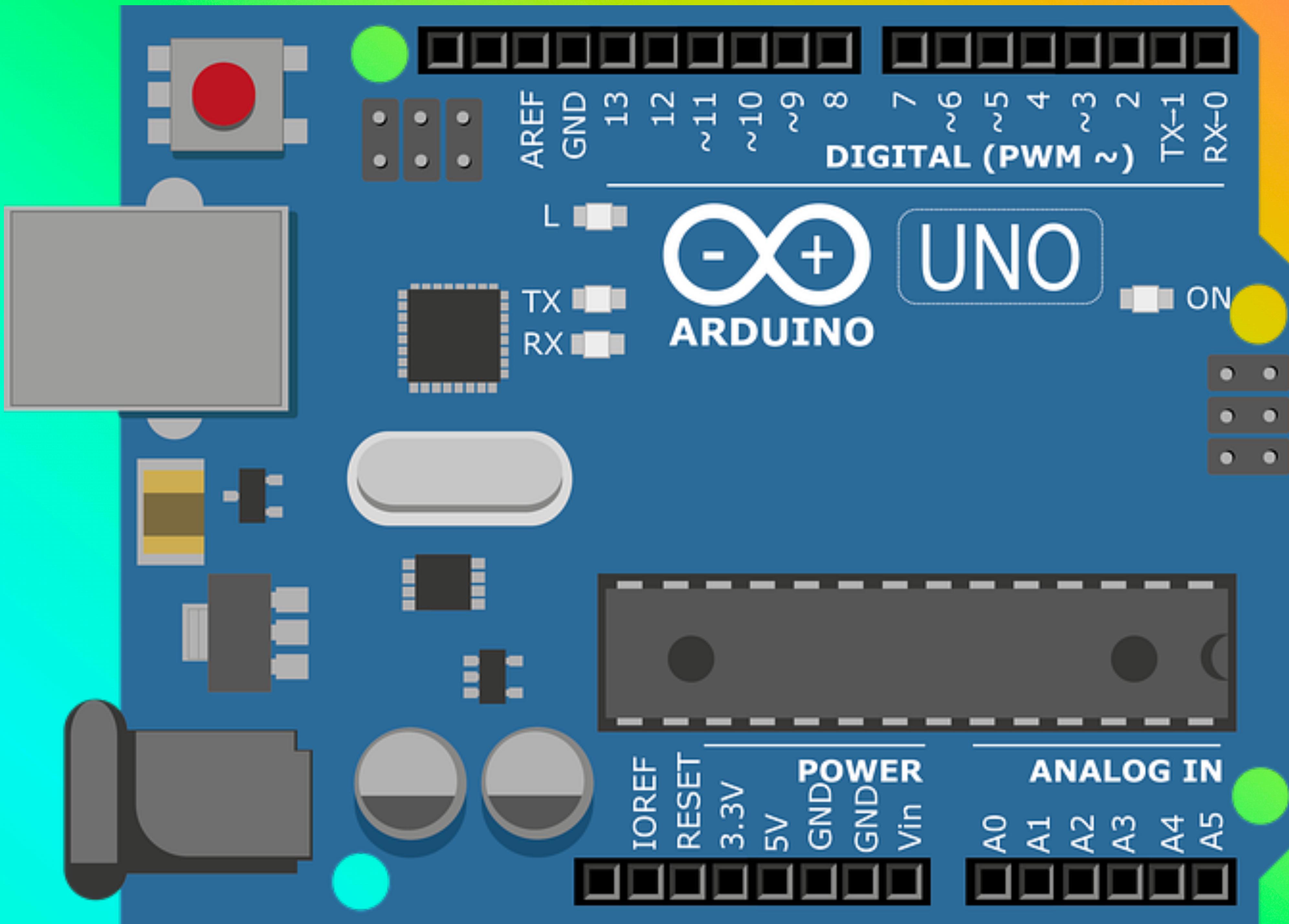
For many embedded projects, e.g. temperature sensor, you won't need much computer

You may not even need a computer, but a microcontroller unit (MCU)

Pros: Potentially very little power consumption => Long battery life

Cons: Can be daunting

MEET YOUR NEW BEST FRIEND



```
void setup() {  
    // put your setup code here, to run once:  
  
}
```

```
void loop() {  
    // put your main code here, to run repeatedly:  
  
}
```

INFORMATION

SETUP: `pinMode(13,INPUT/OUTPUT);` configure a Pin as input or output

`digitalWrite(13, HIGH/LOW);` turns on board LED on or off

`delay(1000);` sleeps for a number of milliseconds

SETUP: `Serial.begin(9600);`

`for(int i = 0; i < 200; i++){`

`Serial.println(„Test”);`

`}` //Initialize i with 0, do the things in the curly braces while i is smaller than 200, increase i at the end of the curly brace

CHALLENGES

- 1) Turn the on-board LED on
- 2) Tell the user something
- 3) Turn the LED on, wait for 2 seconds, Turn it off after 2 more seconds
- 4) Do challenge 3 but in a for loop
- 5) What if you have a really short off time? What if you have a really short on time?
- 6) Try to led the LED fade, from off, to dim, to bright, to on and back