



# MICROCONTROLLERS WORKSHOP

University of Strathclyde Eco-Vehicle Team

Martin Riis

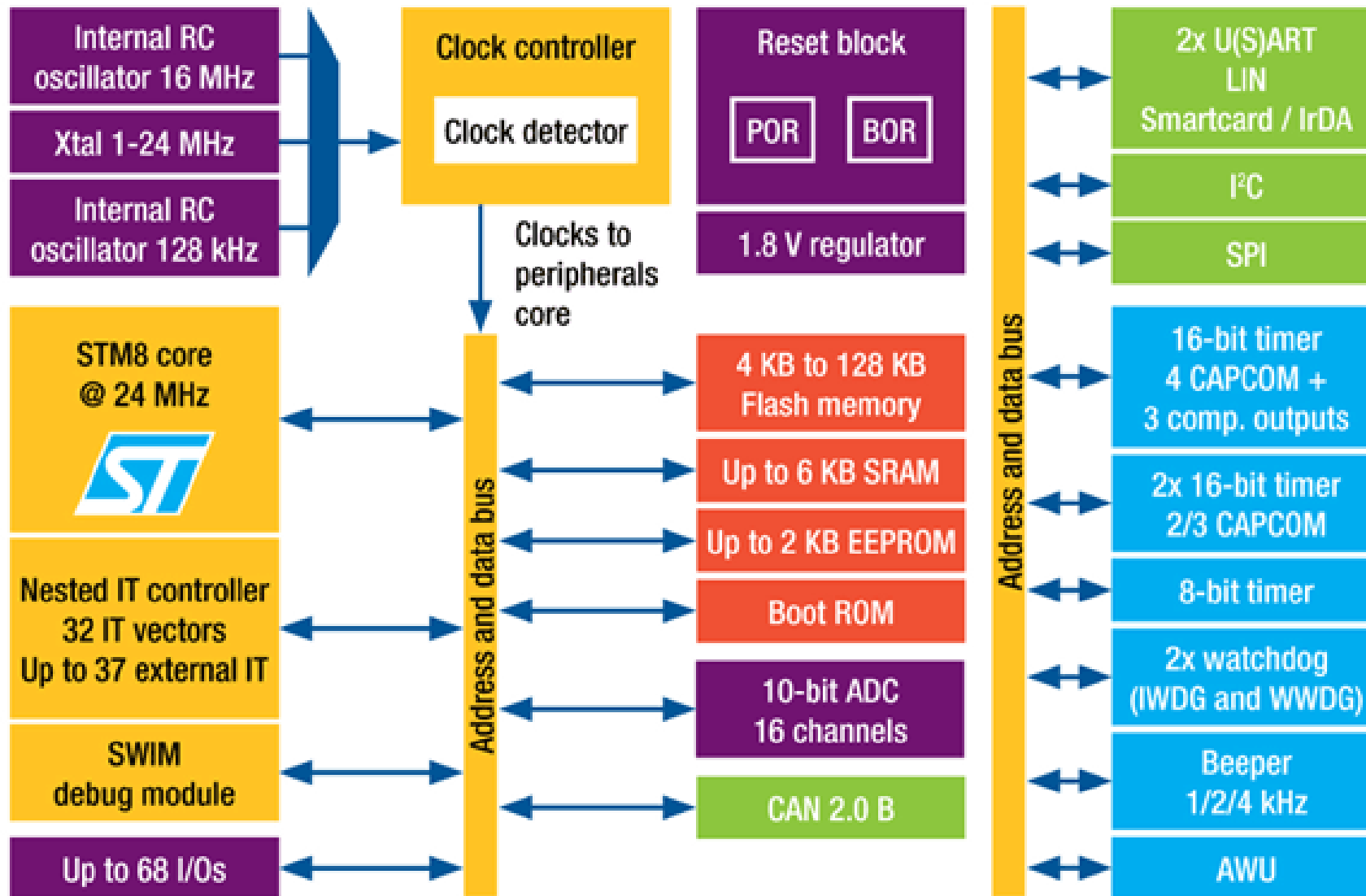
October 2020

# AGENDA

- Short presentation
- Arduino coding demo
- Tutorials for different skill levels

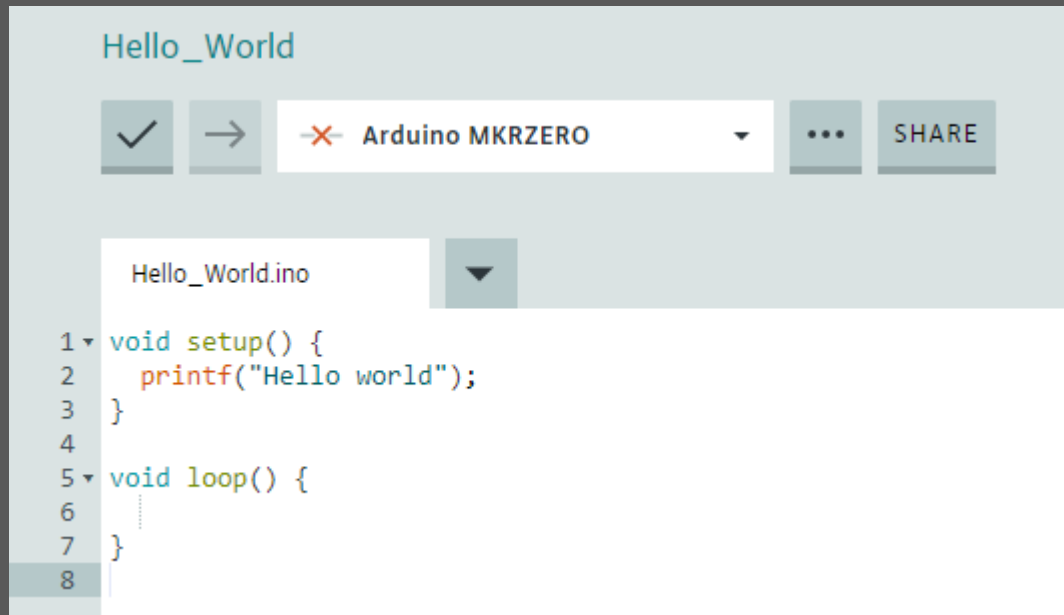
# WHAT IS A MICROCONTROLLER?

- Small, low power processor
- Integrates CPU, memory and peripherals
- Well suited to interfacing with simple sensors and basic control tasks
- Designed to run basic programs written in low level software language (C/C++)



# ARDUINO

- Framework for easily programming microcontrollers
- Language is based on C++
- Provides functions and classes to make programming simpler



The screenshot shows the Arduino IDE interface. At the top, the title bar reads 'Hello\_World'. Below it, there is a toolbar with a checkmark icon, a right-pointing arrow icon, a dropdown menu currently set to 'Arduino MKRZERO' with a red 'X' icon, a three-dot menu icon, and a 'SHARE' button. Below the toolbar, the sketch name 'Hello\_World.ino' is displayed next to a dropdown arrow. The main area contains the C++ code for a 'Hello World' sketch:

```
1 void setup() {  
2   printf("Hello world");  
3 }  
4  
5 void loop() {  
6   ...  
7 }  
8
```

# ARDUINO EXAMPLE CODE

```
#define ANALOG_THRESHOLD 128 // Pre-processor constant

int analogIn; // Variable declaration
int buttonIn;

void setup() // Setup function - runs once
{
    pinMode(1, INPUT); // Digital pin setup
    pinMode(13, OUTPUT);
}

void loop() // Loop function - runs indefinitely
{
    analogIn = analogRead(A0); // Reads analog value from pin A0
    buttonIn = digitalRead(1); // Reads digital value from pin 1

    if (analogIn > ANALOG_THRESHOLD && buttonIn != 0) { // Conditions
        digitalWrite(13, HIGH); // Write logic high to digital pin 13
    }
    else {
        delay(1000); // Waits for 1,000 ms (1 s)
        digitalWrite(13, LOW); // Write logic low to digital pin 13
    }
}
```

# HOW DOES THIS RUN ON A MICROCONTROLLER?

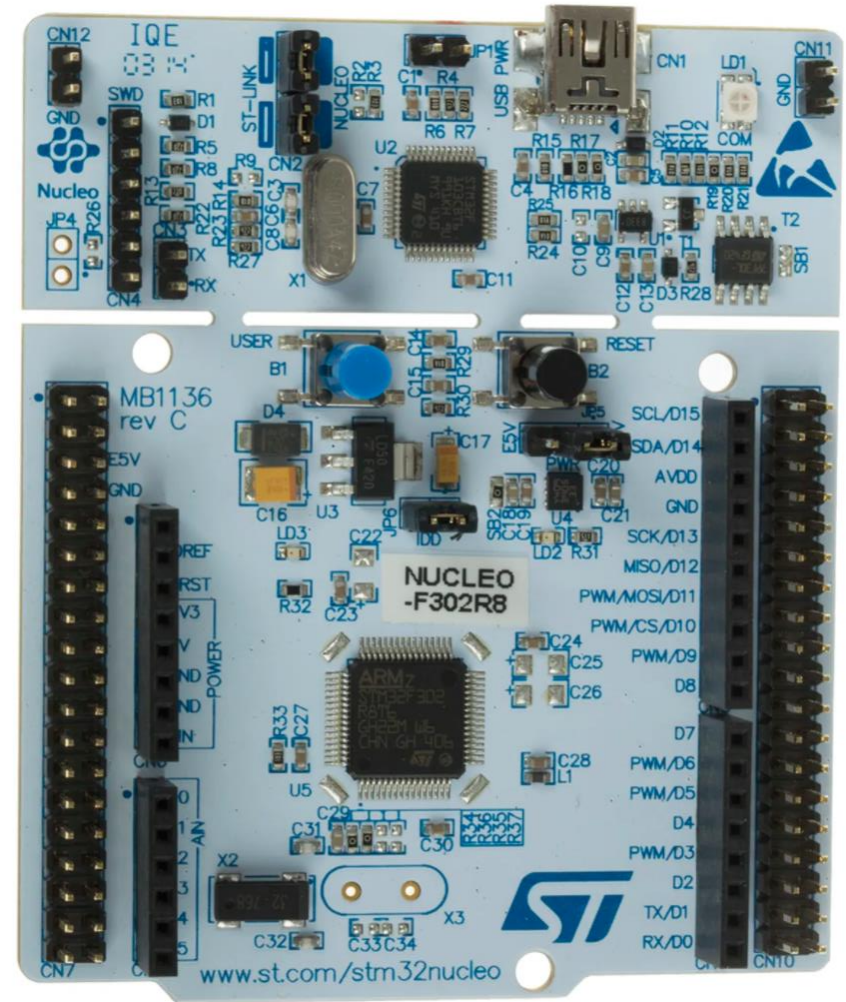
- Compiler translates Arduino code into assembly
- Assembly is low level code containing only labels and processor instructions that can run directly on the microcontroller's processor

```
int a = 10;
int b = 12;
int c = a + b;
c = a * c;
if (c > *d) {
    c = c + 2;
}
else {
    c = c + 1;
}
```

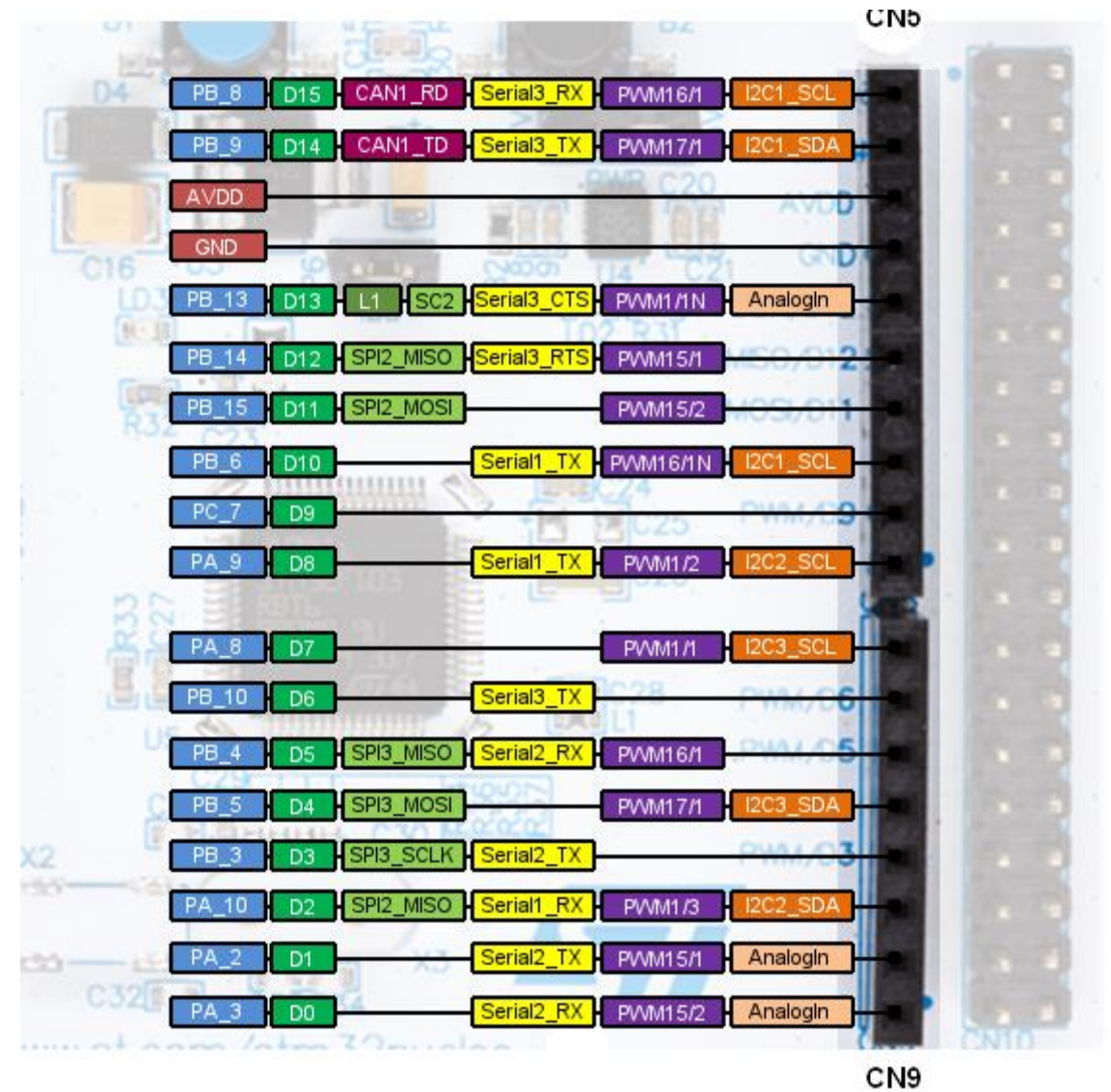
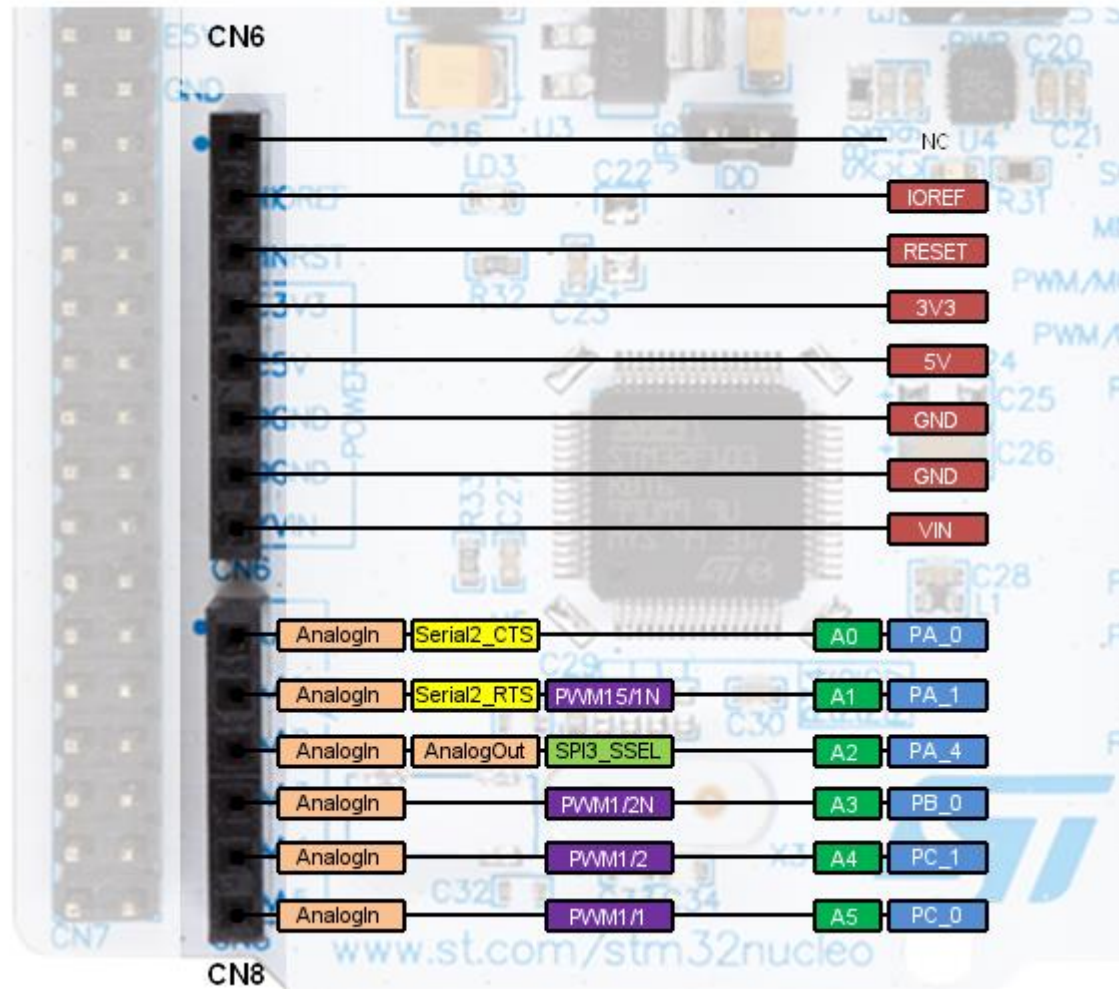
```
L0:
    ldi x4, #10
    ldi x5, #12
    add x6, x4, x5
    mul x6, x4, x6
    ld x7, 12(sp)
    blt x7, x6, L1
    addi x6, x6, #2
    j EXIT
L1:
    addi x6, x6, #1
EXIT:
    <end of program>
```

# STM32F302R8

- ARM Cortex-M4 at 72 MHz
- Floating Point Unit
- 64 KB Flash
- 16 KB RAM
- ADC, DAC
- 9 timers
- I2C, UART, SPI, USB 2.0, CANbus
- Analog comparators, op-amp
- 51 I/O pins
- 3.3 V!!!







<https://os.mbed.com/platforms/ST-Nucleo-F302R8/>