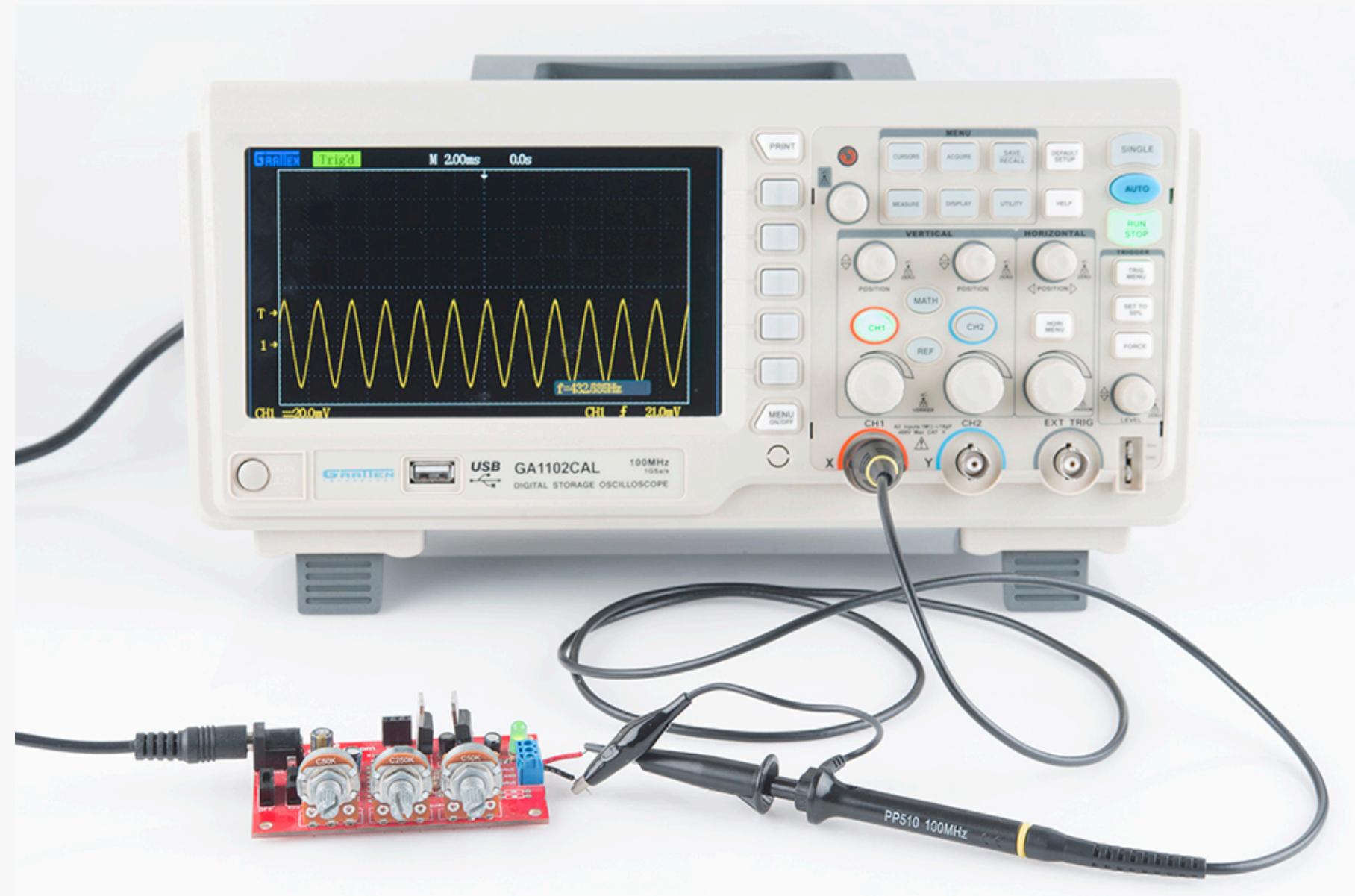


HARDWARE SECURITY

101

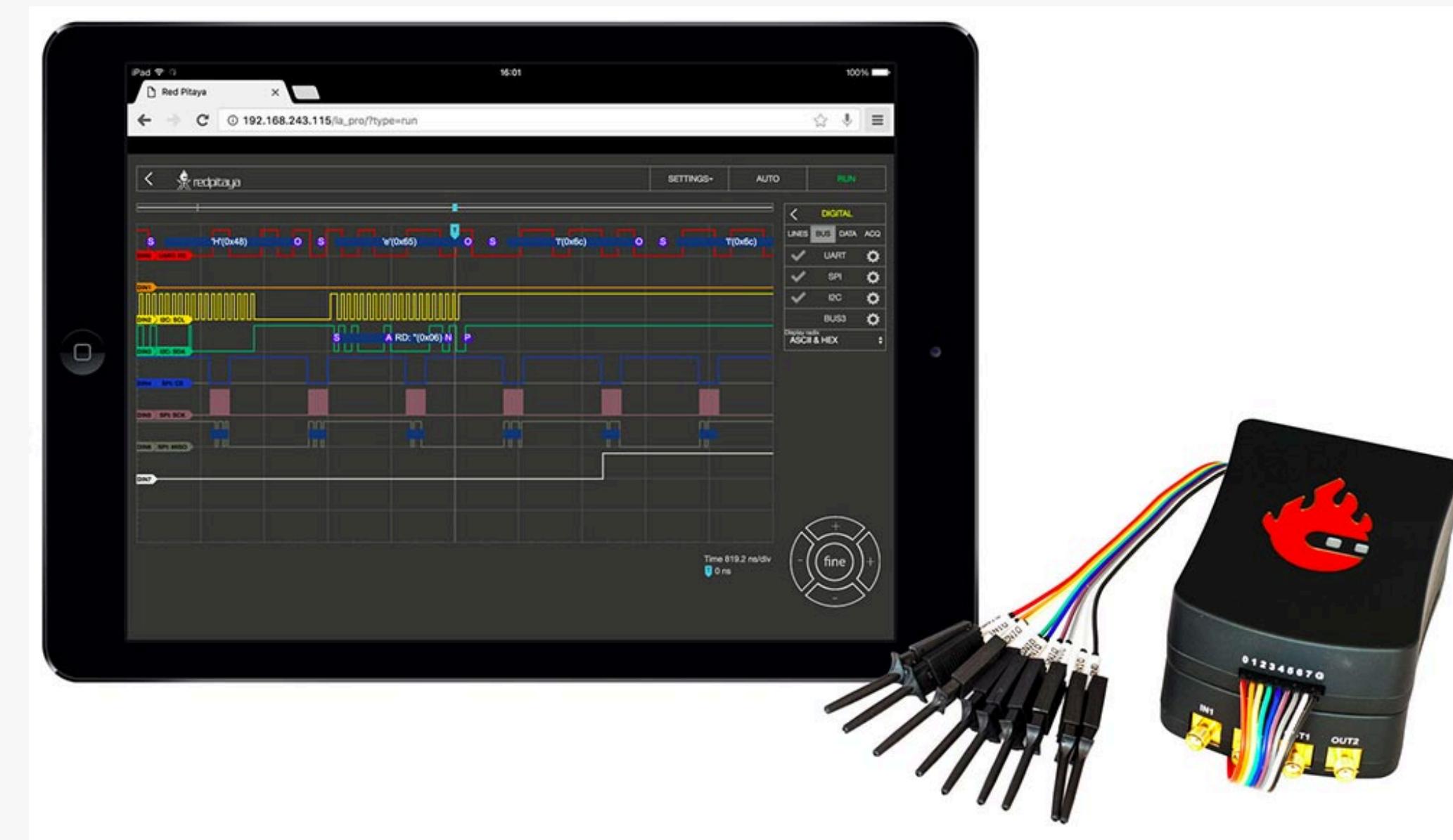
HACKING HARDWARE

TOOLS AND EQUIPMENT



Oscilloscope

TOOLS AND EQUIPMENT



Logic analyzer

TOOLS AND EQUIPMENT



Signal generator

TOOLS AND EQUIPMENT



Multimeter

TOOLS AND EQUIPMENT



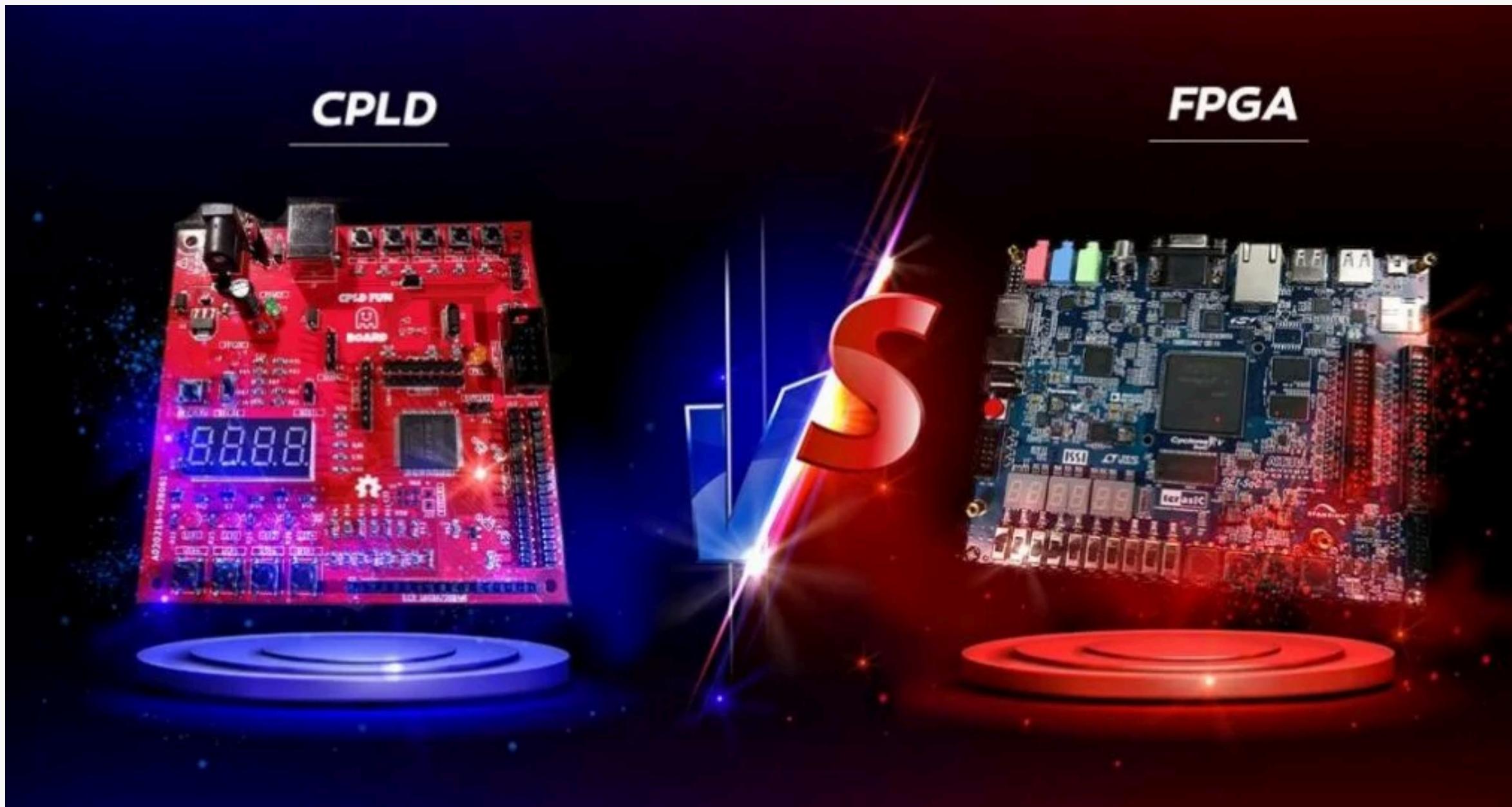
Spectrum Analyzer

TOOLS AND EQUIPMENT



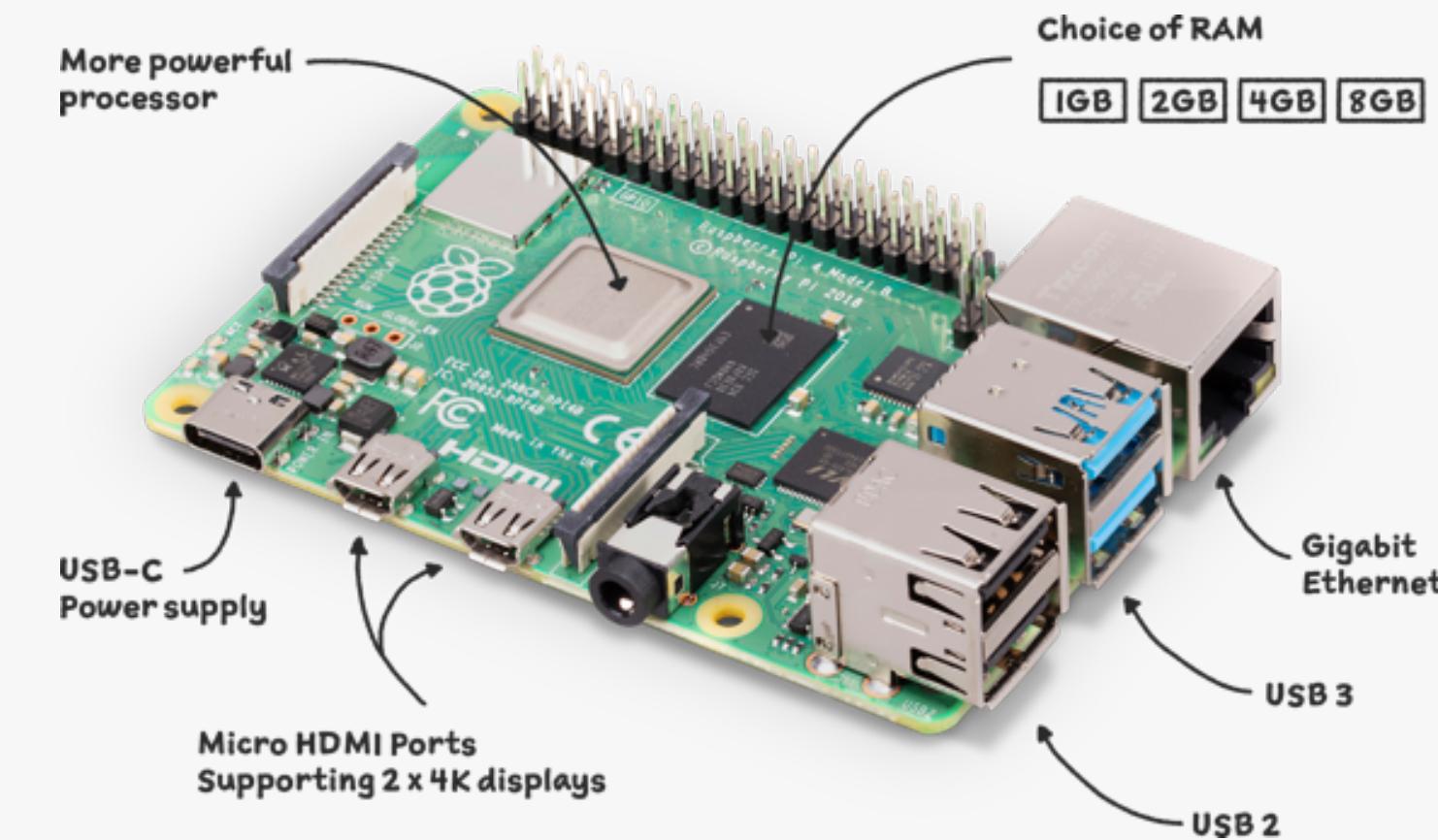
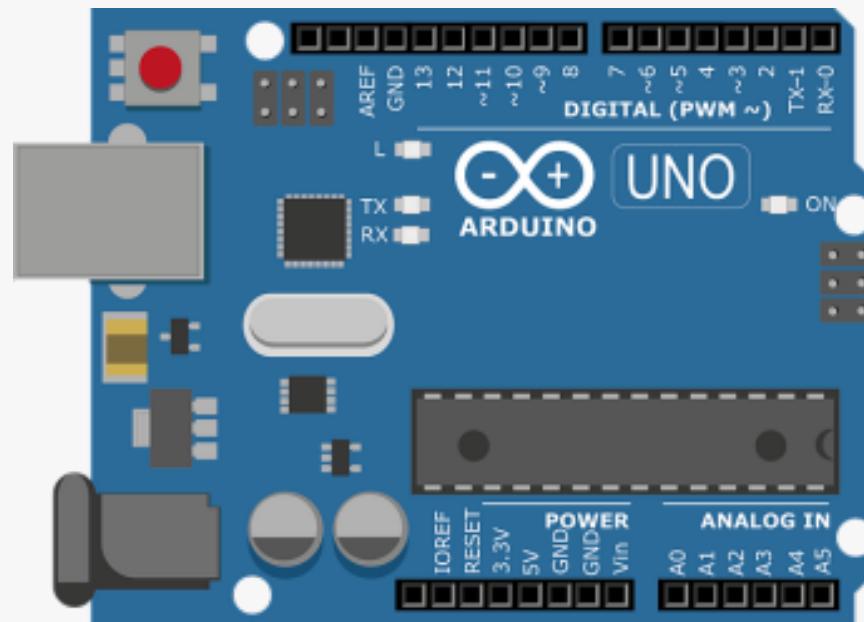
Power Supply

TOOLS AND EQUIPMENT



FPGA & CPLD

TOOLS AND EQUIPMENT



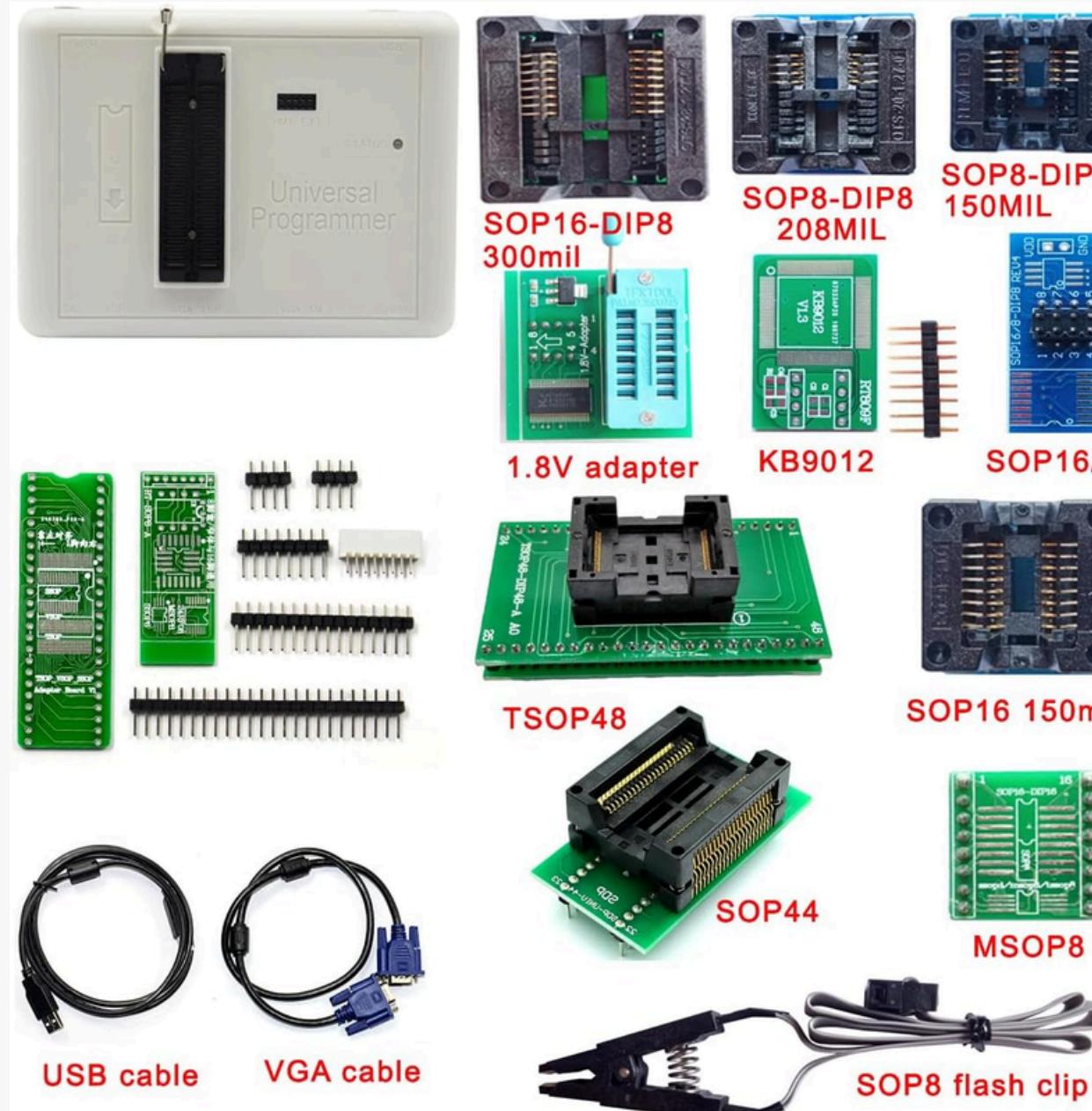
Open Source Hardware

TOOLS AND EQUIPMENT



Open Source Tools

TOOLS AND EQUIPMENT



Universal Programmer



Microchip PIC programmer

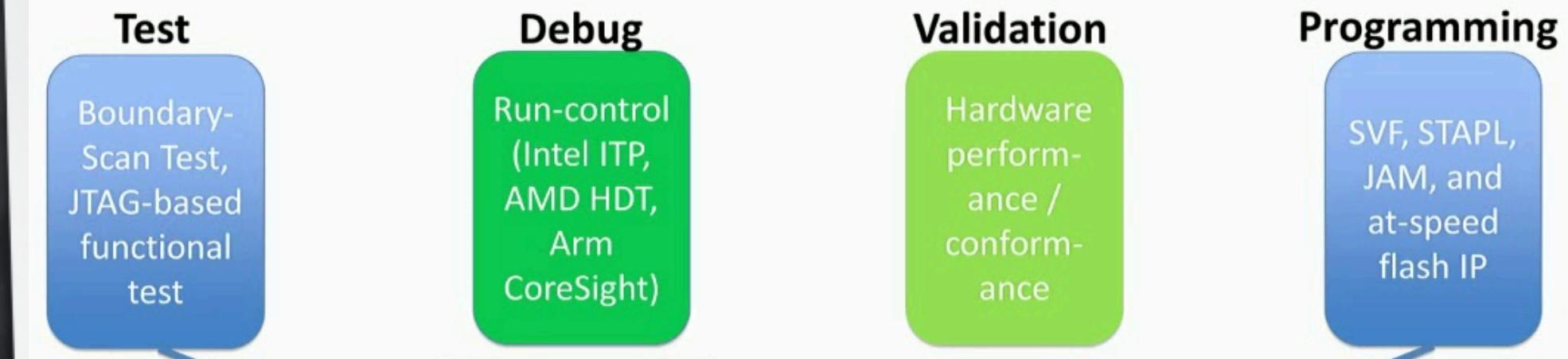


Atmel AVR programmer

TOOLS AND EQUIPMENT



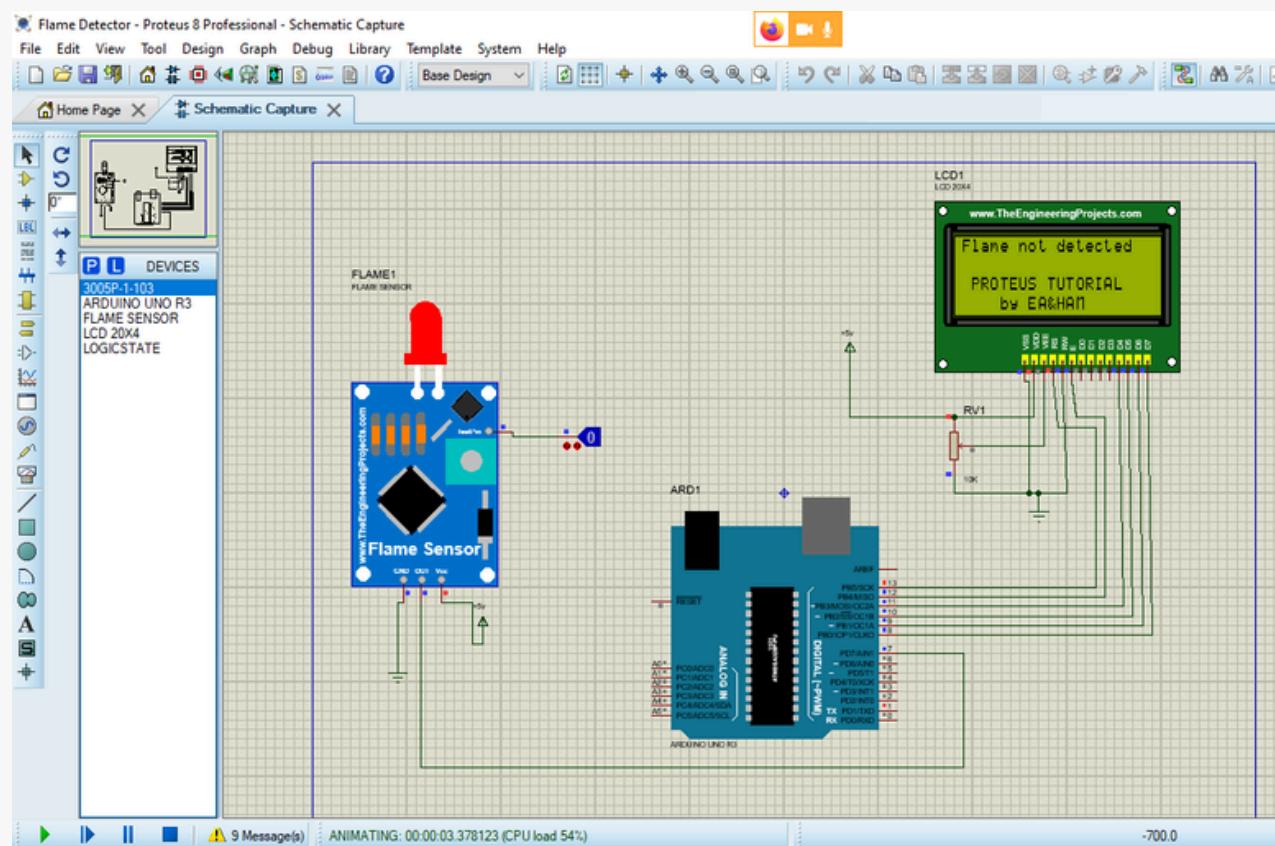
JTAG Applications



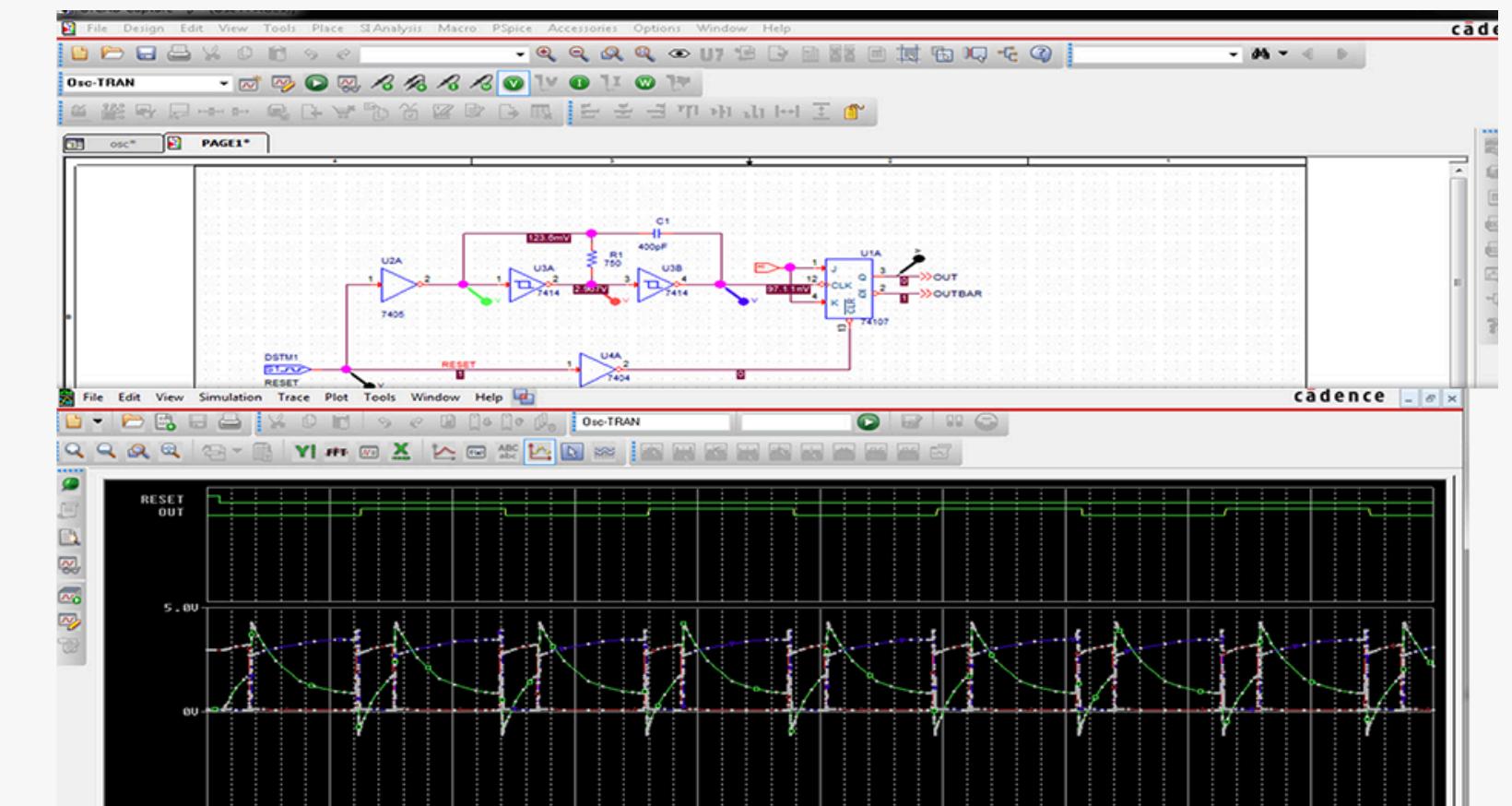
ST electronics STM programmer

ARM JTAG

SOFTWARE

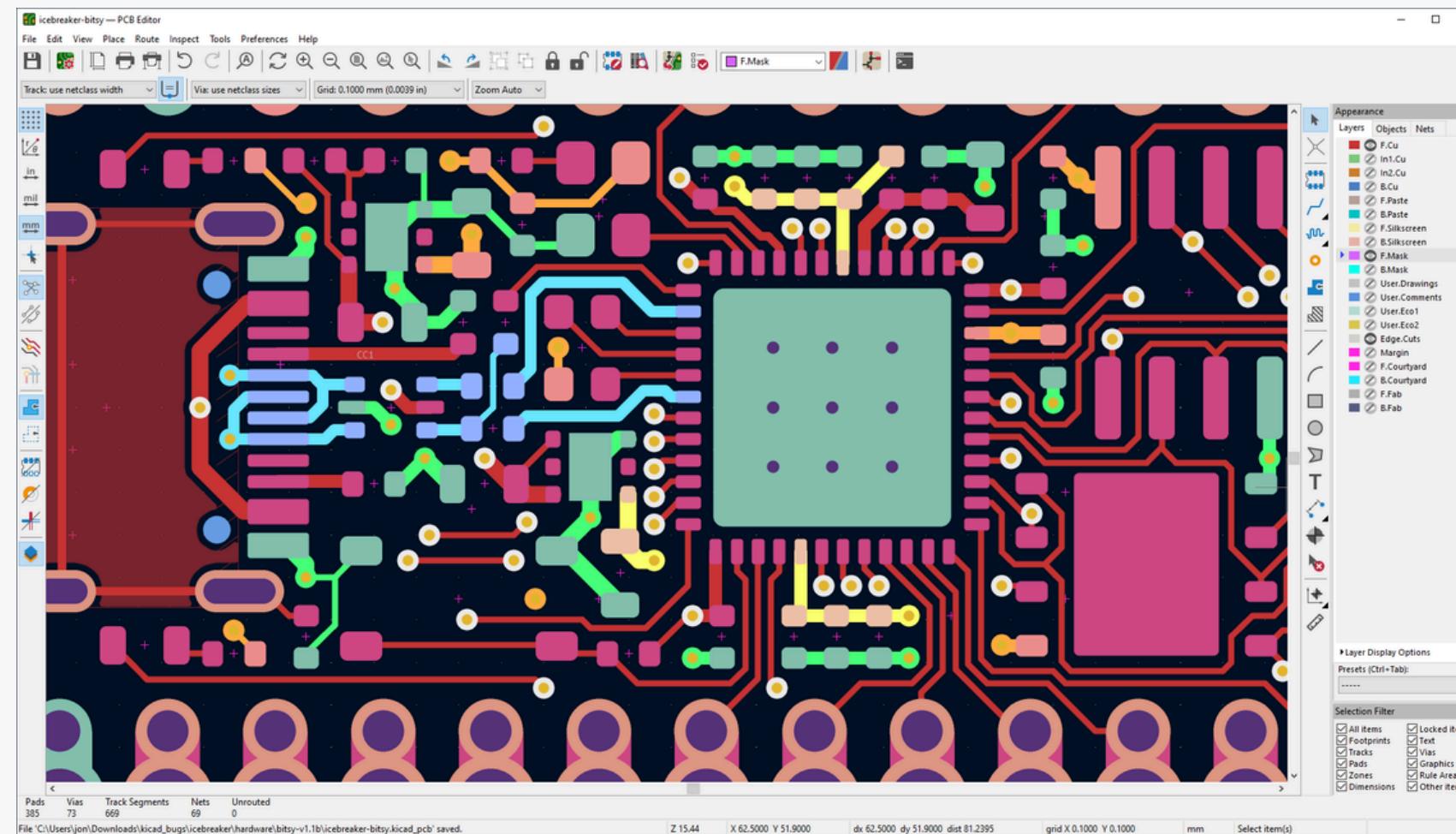


Proteus

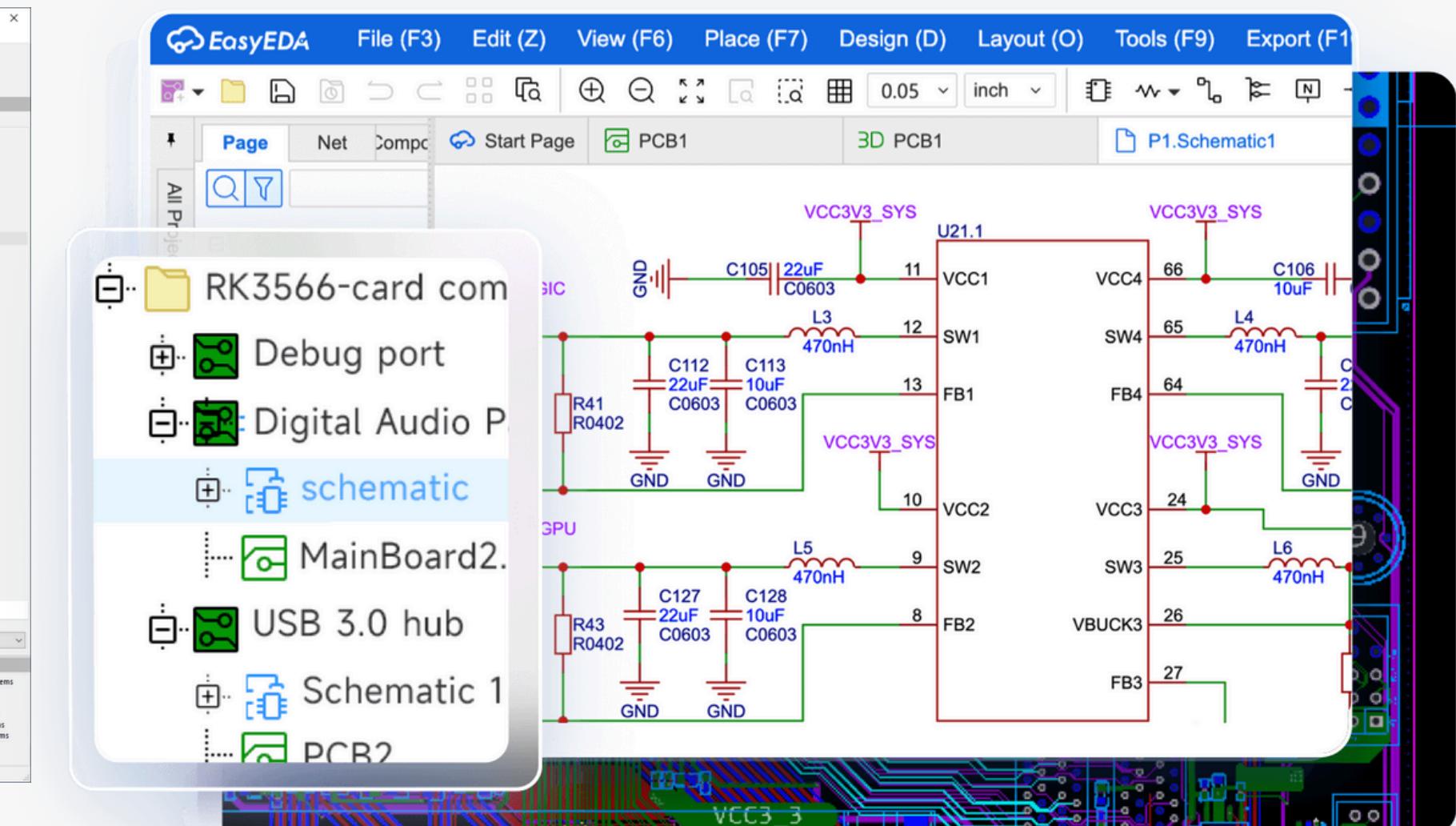


Pspice

SOFTWARE

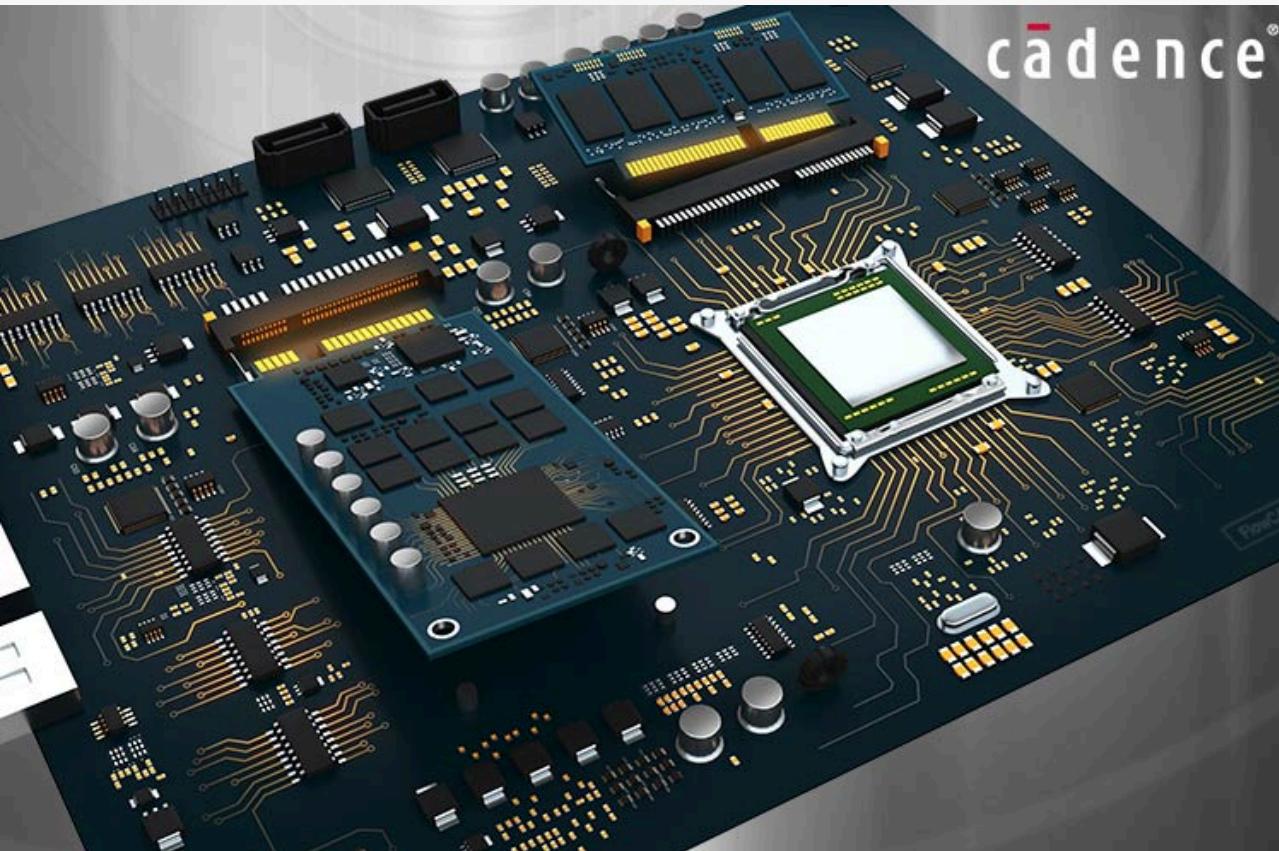


KiCad

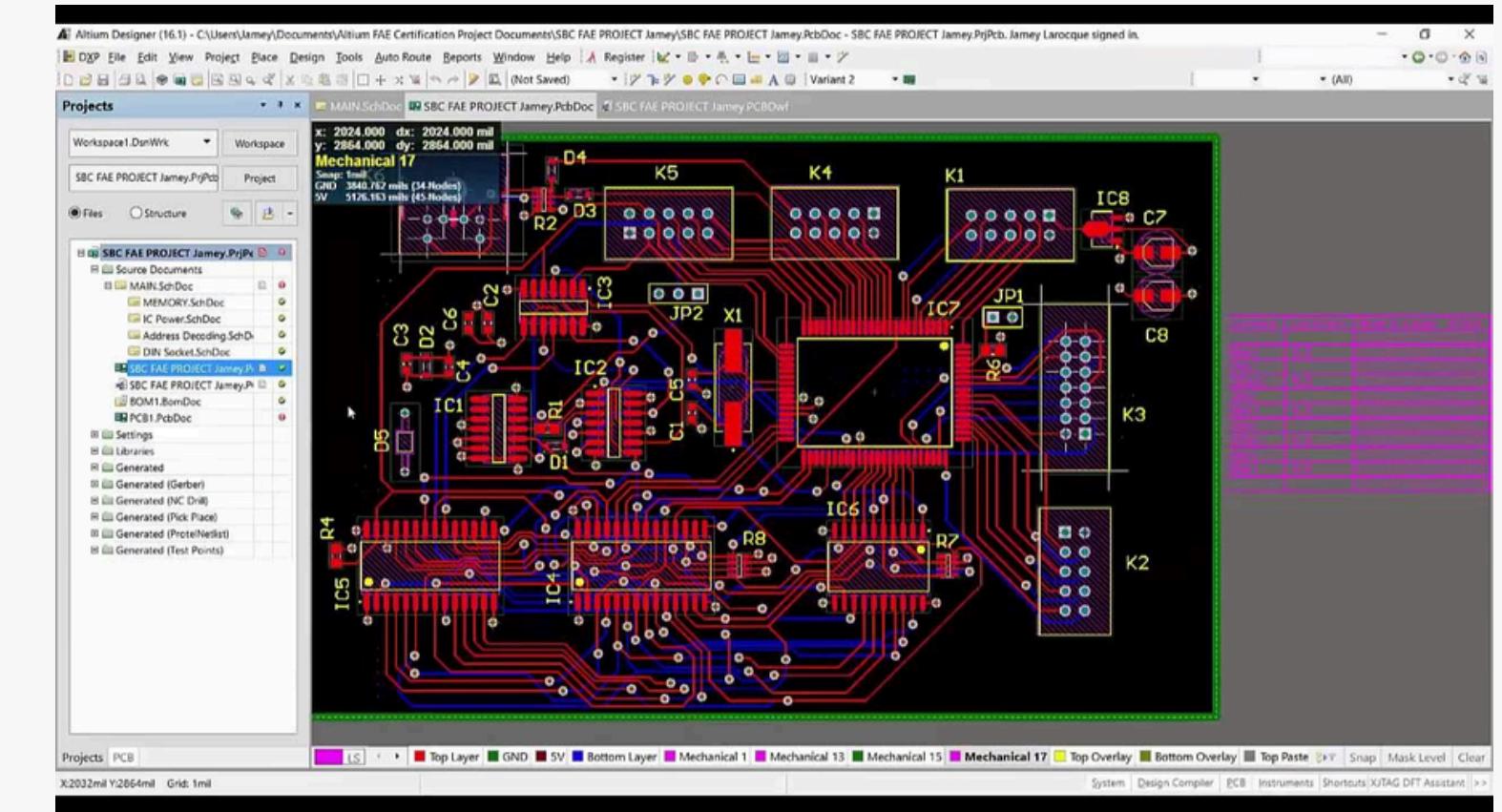


EasyEDA

SOFTWARE



cadence Allegro



Altium Designer

SOFTWARE

The screenshot shows the Arduino IDE 2.0.0-beta.2 interface. The main window displays the `Blink.ino` sketch. The code is as follows:

```
19 // This example code is in the public domain.
20 // http://www.arduino.cc/en/Tutorial/Blink
21 /*
22  * the setup function runs once when you press reset or power the board
23  */
24 void setup() {
25     // initialize digital pin LED_BUILTIN as an output.
26     pinMode(LED_BUILTIN, OUTPUT);
27     Serial.begin(9600);
28 }
29
30 // the loop function runs over and over again forever
31 void loop() {
32     digitalWrite(LED_BUILTIN, HIGH);    // turn the LED on (HIGH is the voltage level)
33     delay(1000);                      // wait for a second
34     digitalWrite(LED_BUILTIN, LOW);   // turn the LED off by making the voltage LOW
35     delay(1000);                      // wait for a second
36     Serial.println("LED is blinking");
37 }
38
39
40
```

The Serial Monitor window shows the output: "LED is blinking" repeated every second.

Arduino IDE

The screenshot shows the Atmel Studio interface. The main window displays the `main.c` file. The code is as follows:

```
/*
 * ATtiny85_Blink.c
 *
 * * Created: 25.06.2019 18:30:08
 * * Author : Ewald
 */
#include <avr/io.h>
#include <util/delay.h>

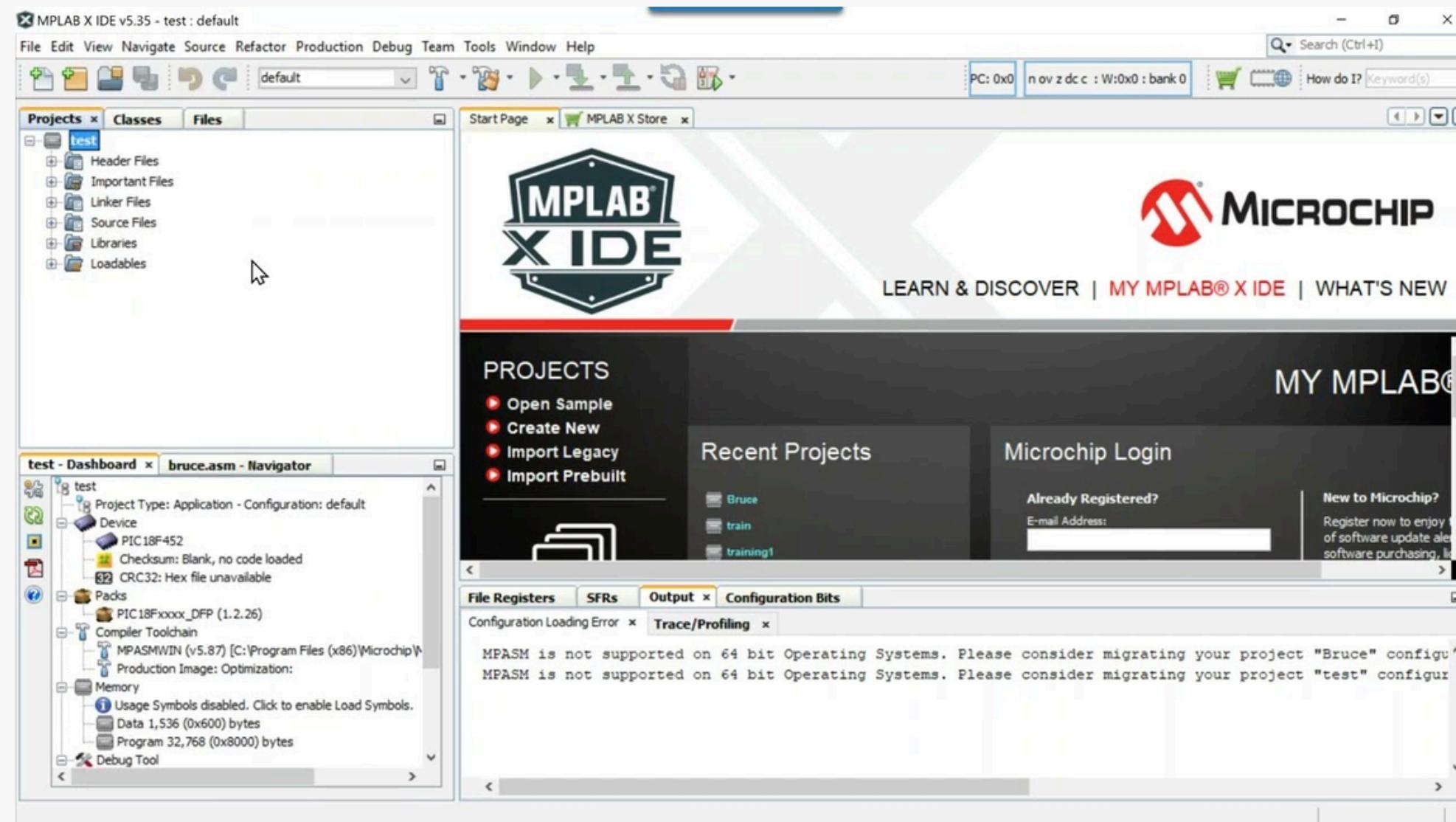
int main(void)
{
    DDRB |= (1<<PB4);

    while (1)
    {
        PORTB |= (1<<PB4);
        // _delay_ms(1000);
        PORTB &= ~(1<<PB4);
        // _delay_ms(1000);
    }
}
```

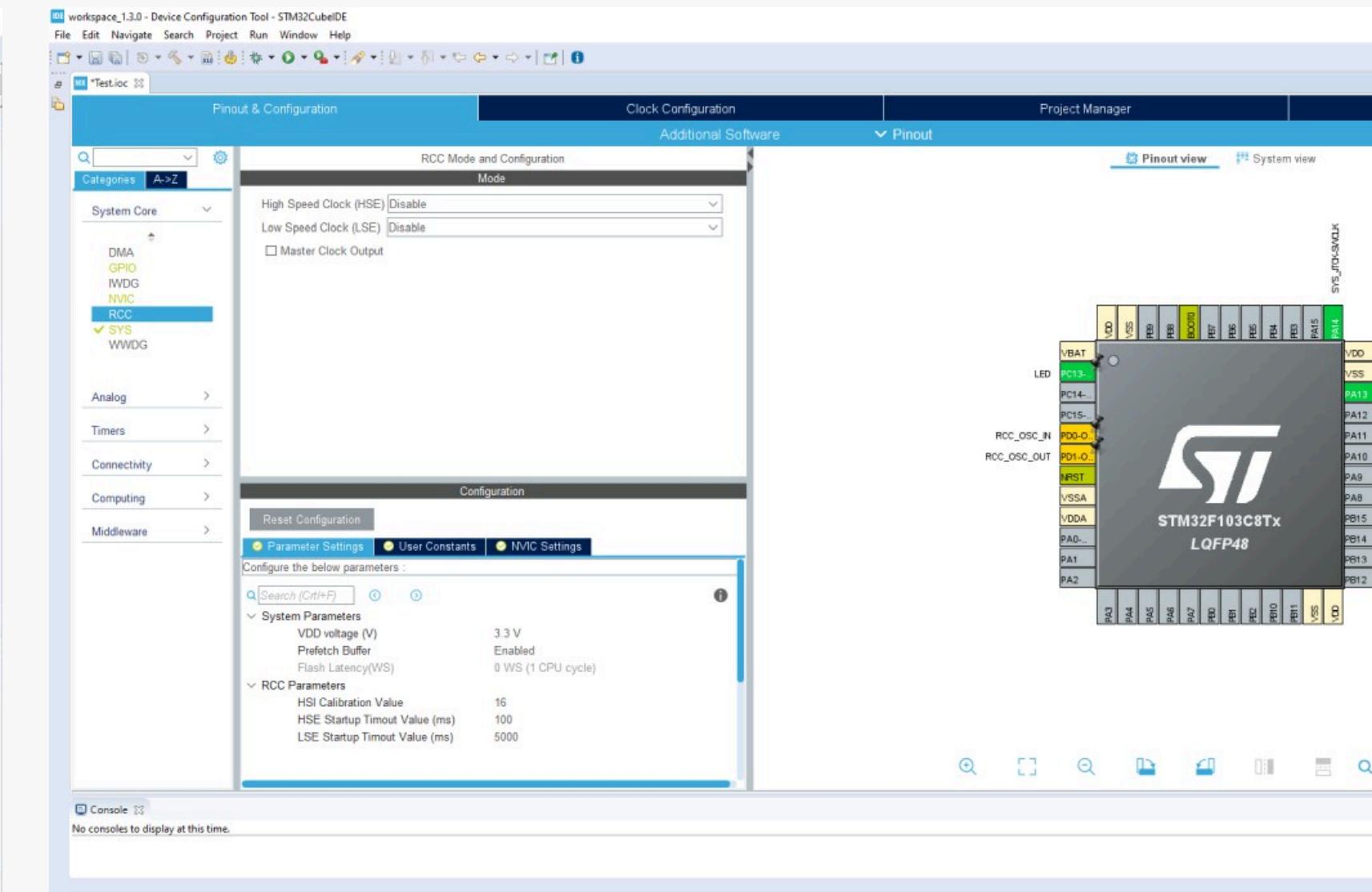
The toolbar at the top has a question mark icon highlighted with a red circle. The status bar at the bottom indicates: "UTF-8 ■ Arduino Nano 33 BLE on /dev/cu.usbmodem143201 4 □".

ATMEL Studio

SOFTWARE

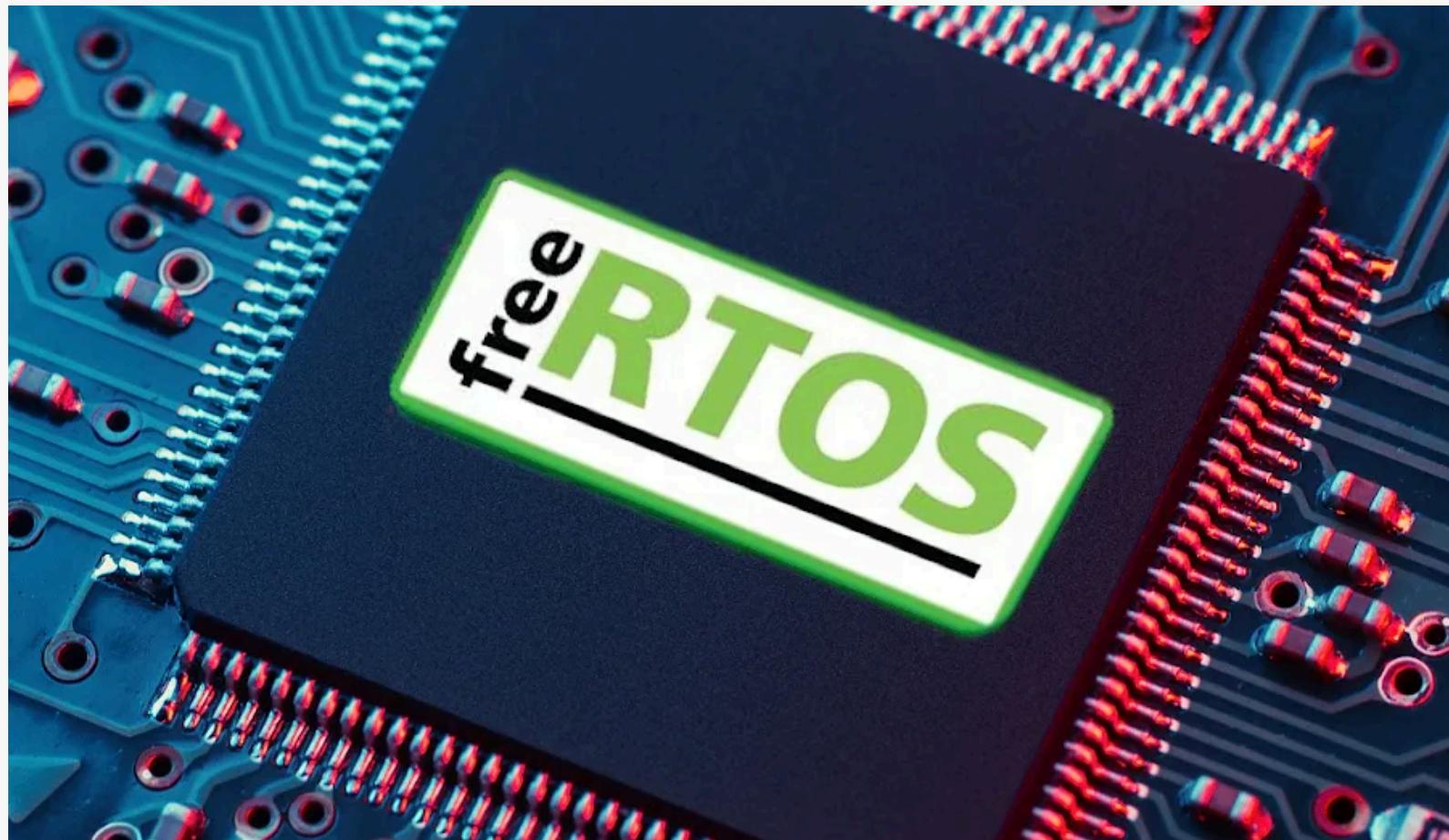


MPLAB X

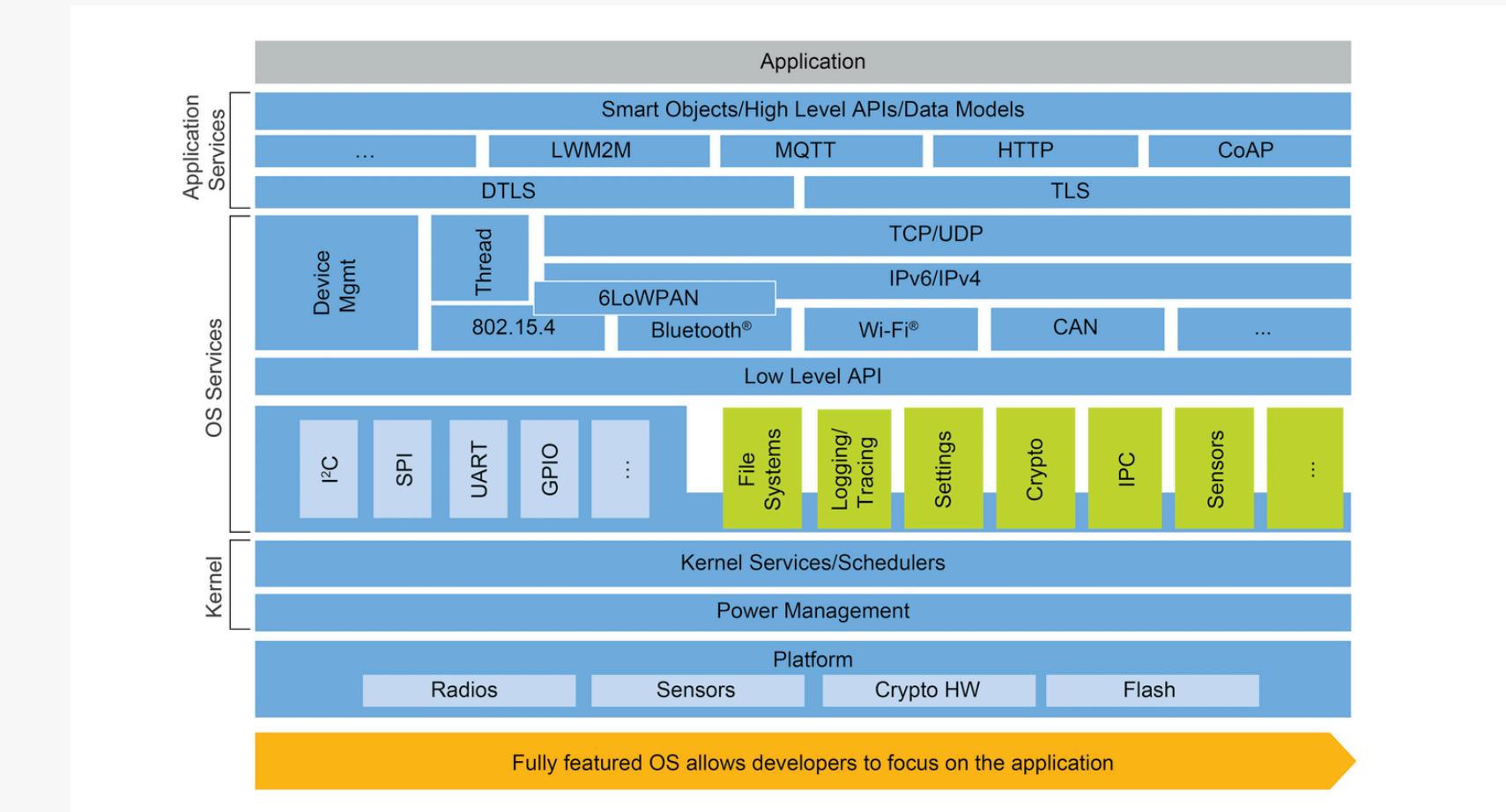


STM32CUBE IDE

RTOS



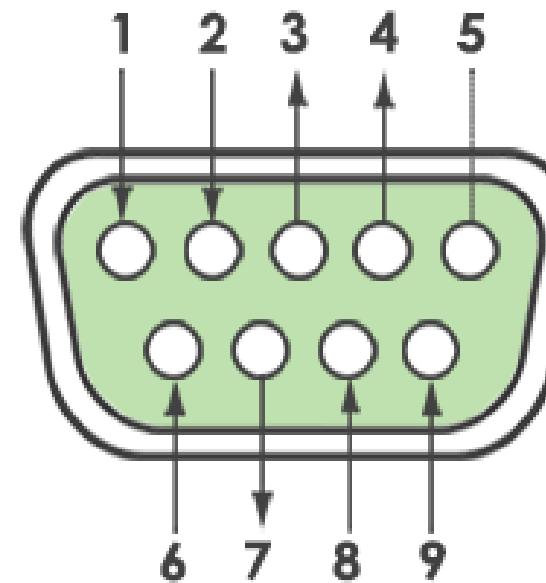
FREE RTOS



ZEPHYR OS

STANDARD PROTOCOLS RS232

DB9 Male

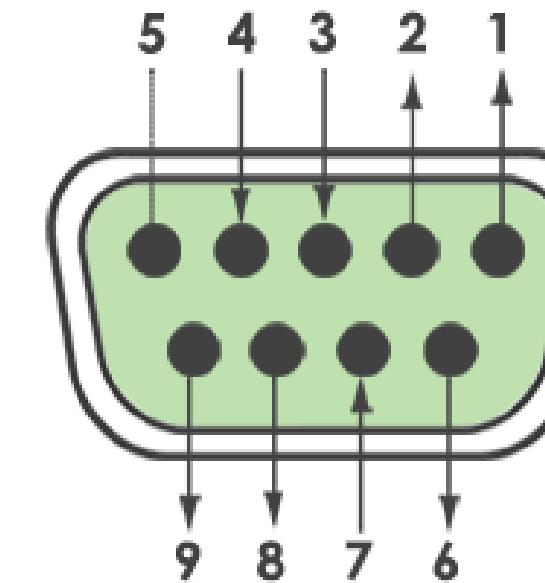


Pin	Signal Direction	Signal Name	Signal Function
1	←	CD	Carrier Detect
2	←	RxD	Receive Data
3	→	TxD	Transmit Data
4	→	DTR	Data Terminal Ready
5	—	GND	Ground
6	←	DSR	Data Set Ready
7	→	RTS	Request To Send
8	←	CTS	Clear To Send
9	←	RI	Ring Indicator

→ Transmitted from DTE Device

← Received by DTE Device

DB9 Female

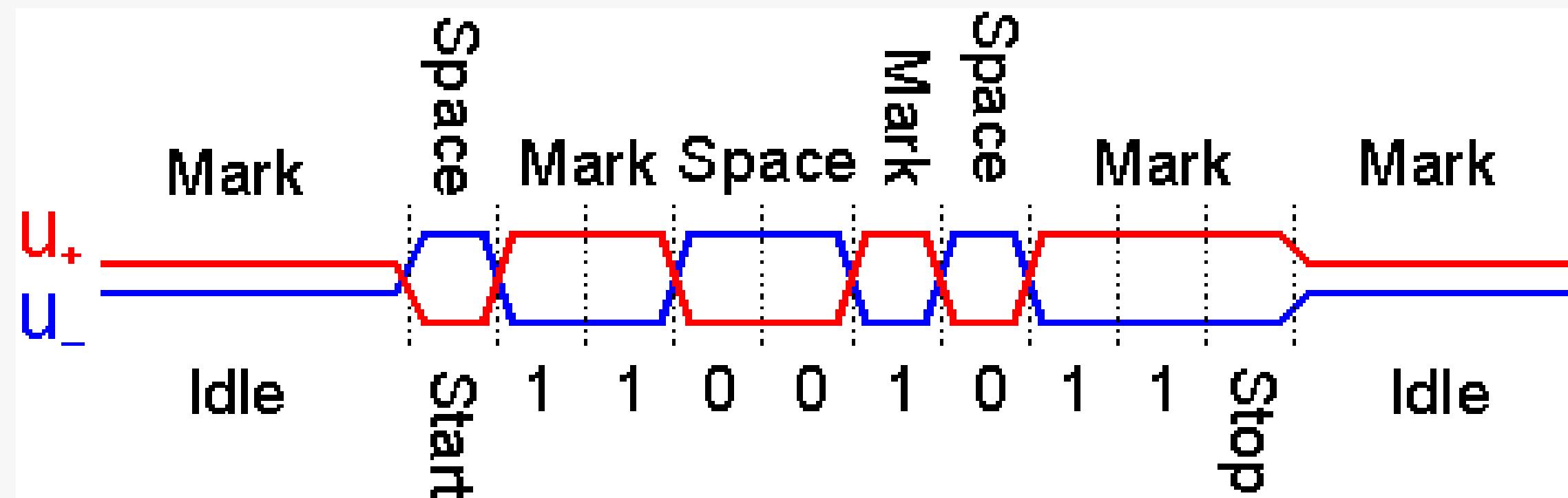
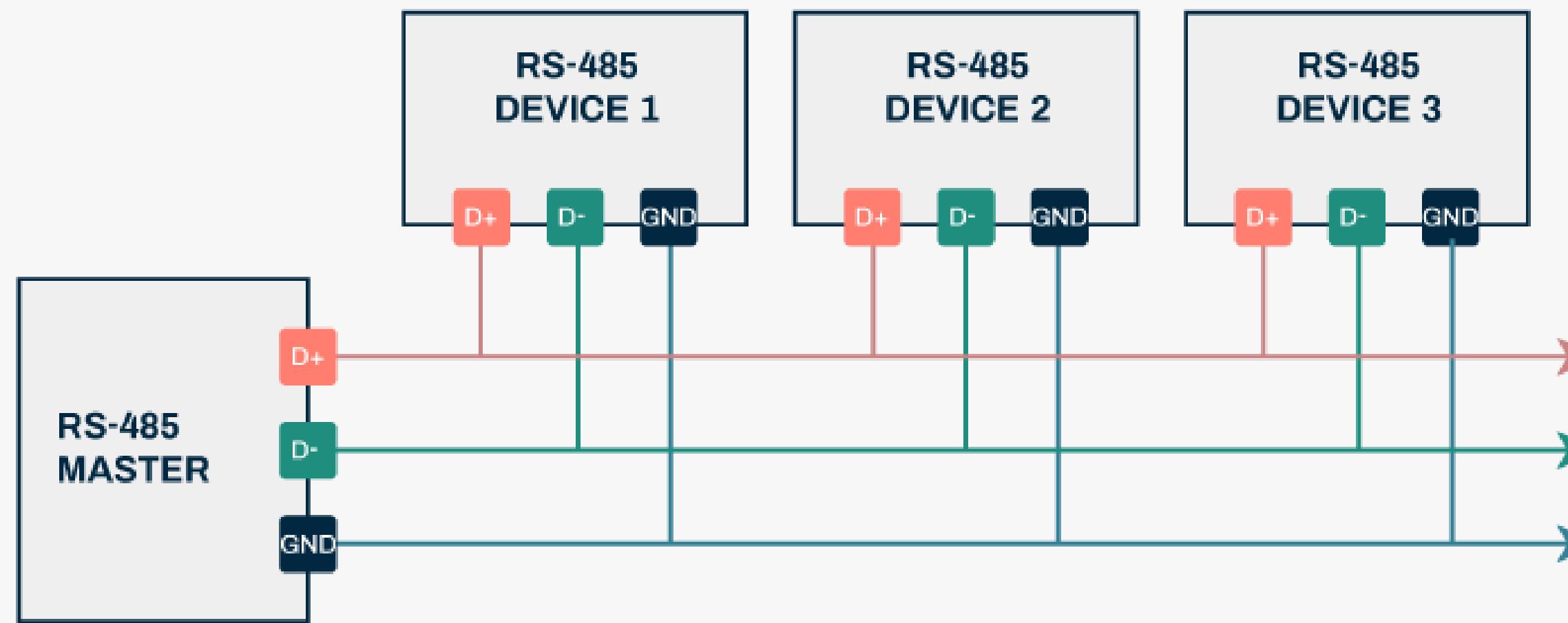


Pin	Signal Direction	Signal Name	Signal Function
1	→	CD	Carrier Detect
2	→	TxD	Transmit Data
3	←	RxD	Receive Data
4	←	DTR	Data Terminal Ready
5	—	GND	Ground
6	→	DSR	Data Set Ready
7	→	CTS	Clear To Send
8	→	RTS	Request To Send
9	→	RI	Ring Indicator

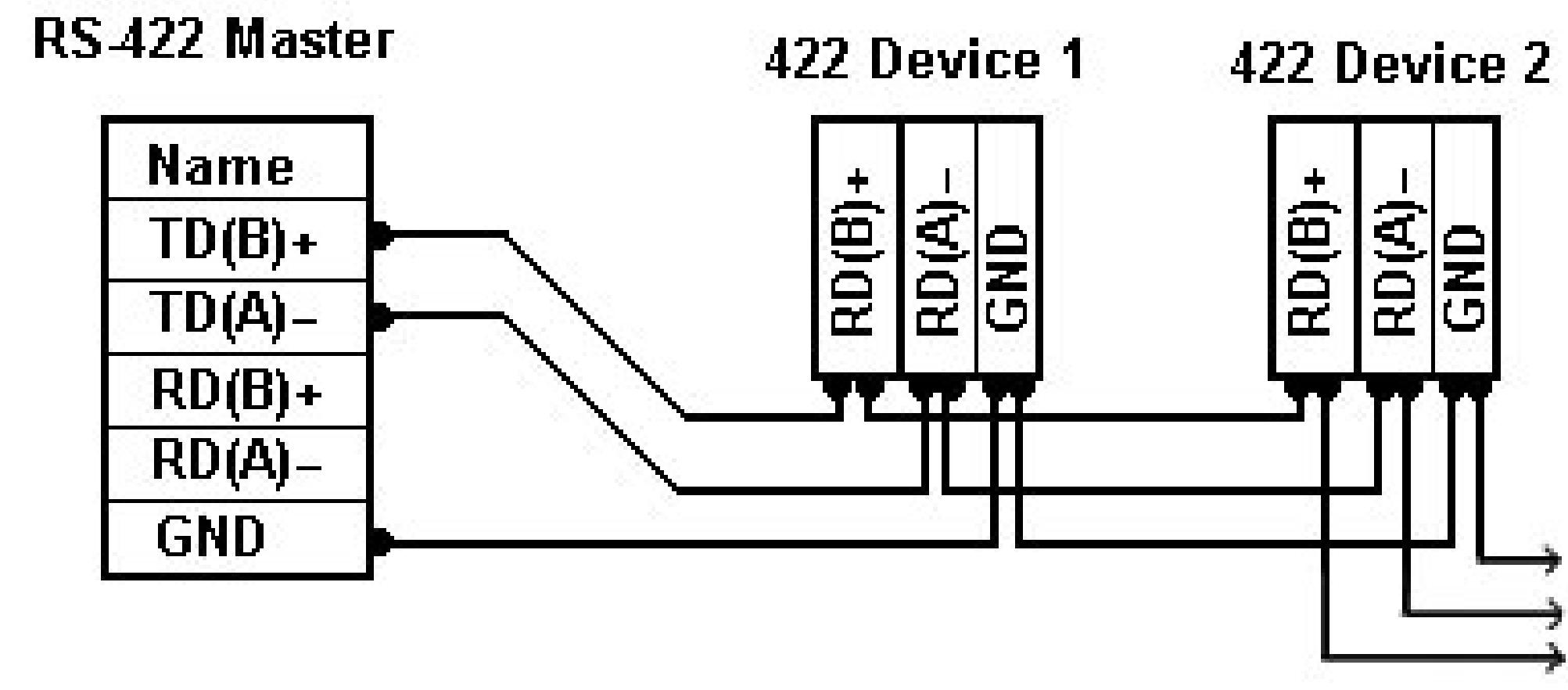
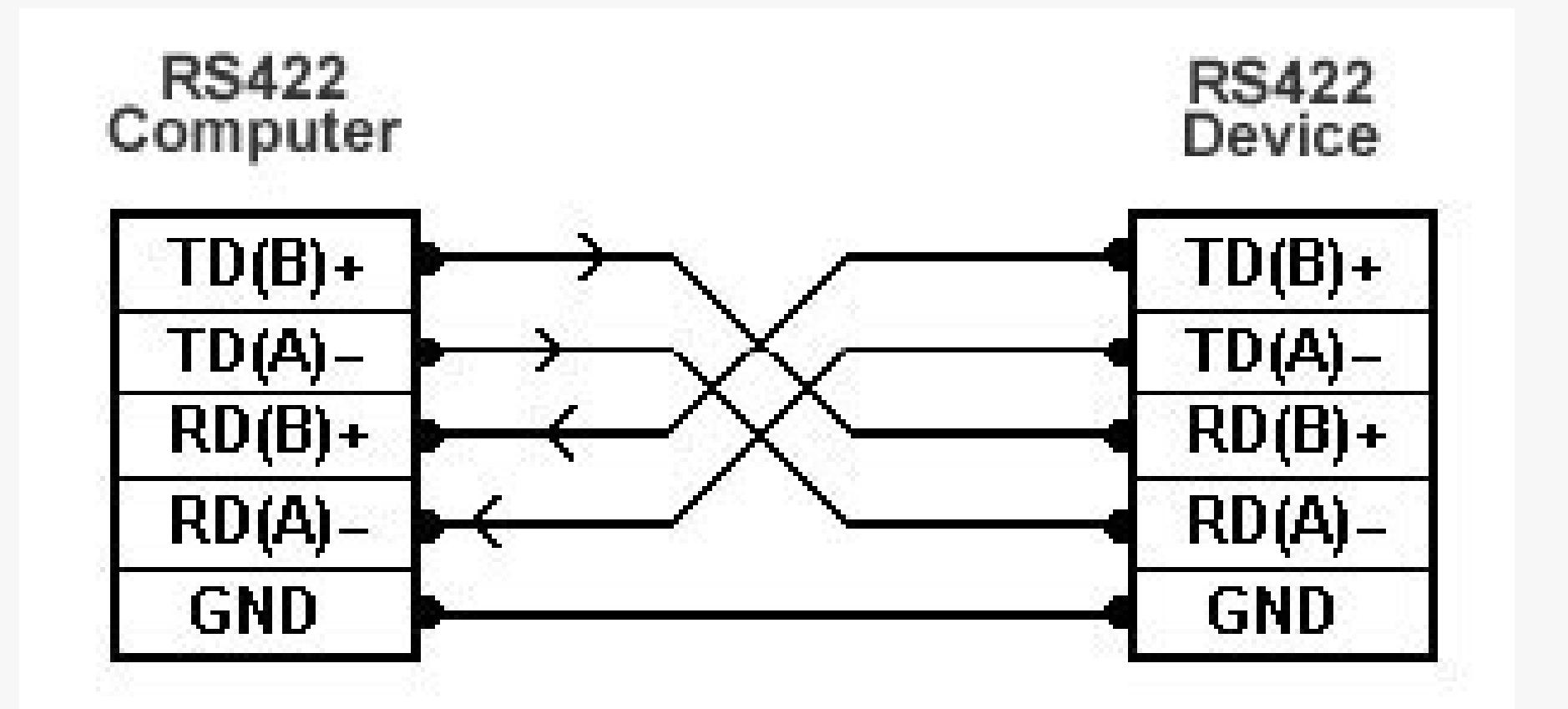
→ Transmitted from DCE Device

← Received by DCE Device

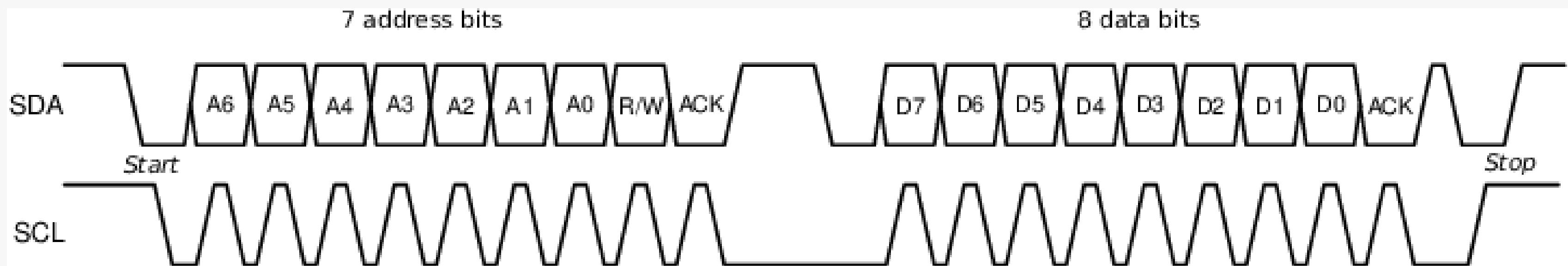
STANDARD PROTOCOLS RS485



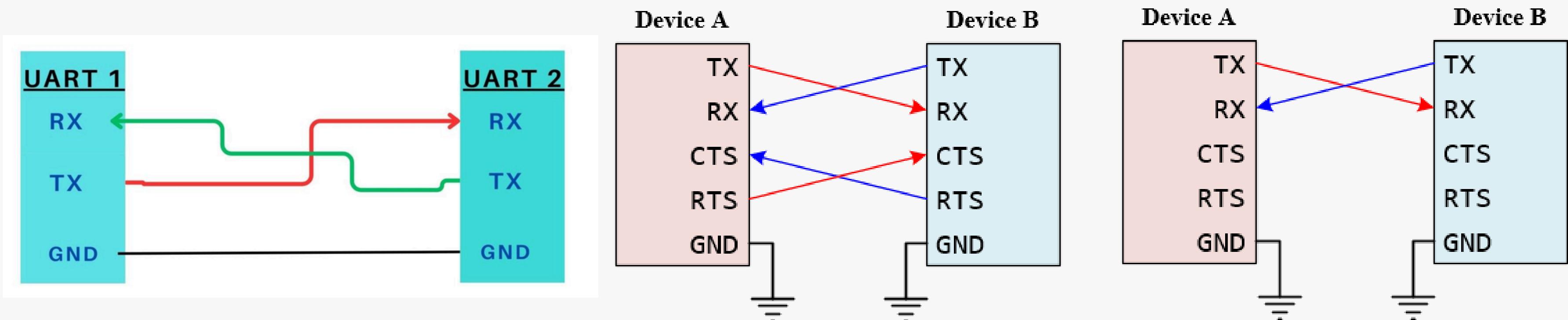
STANDARD PROTOCOLS RS422



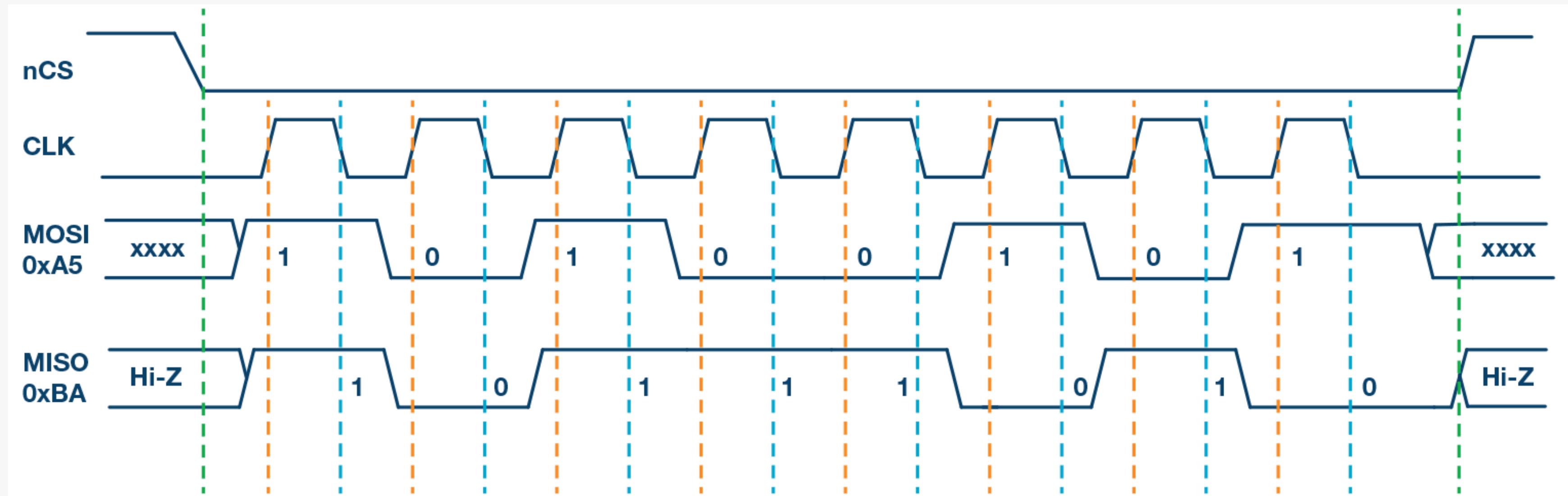
STANDARD PROTOCOLS I₂C



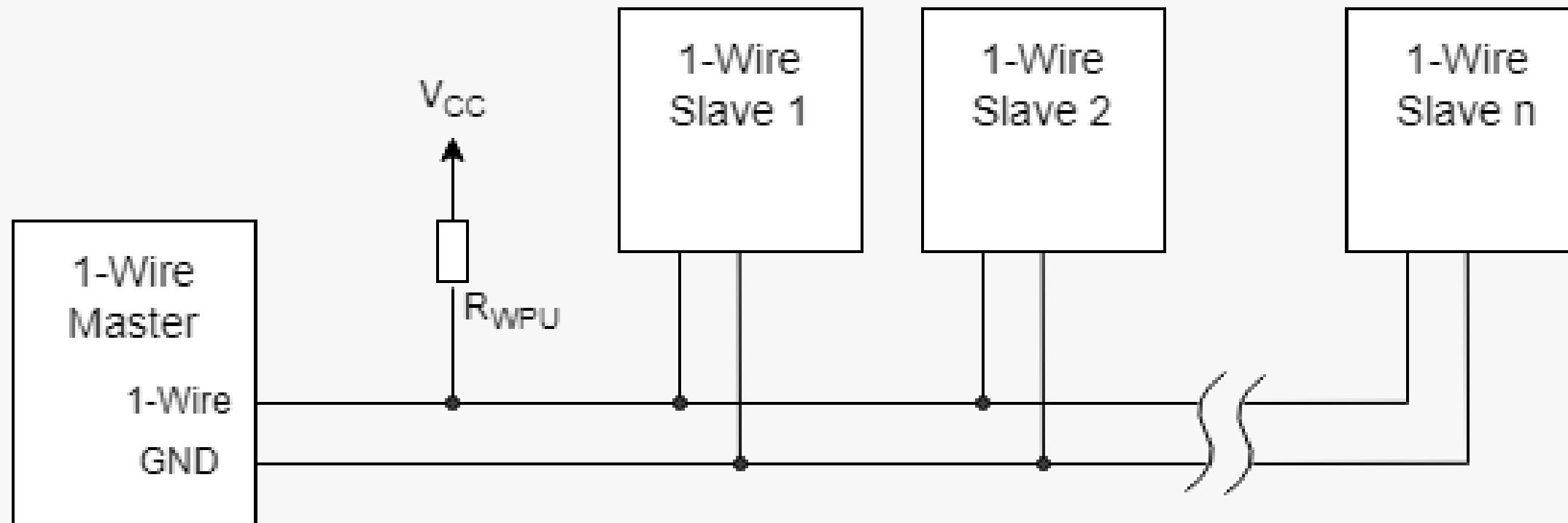
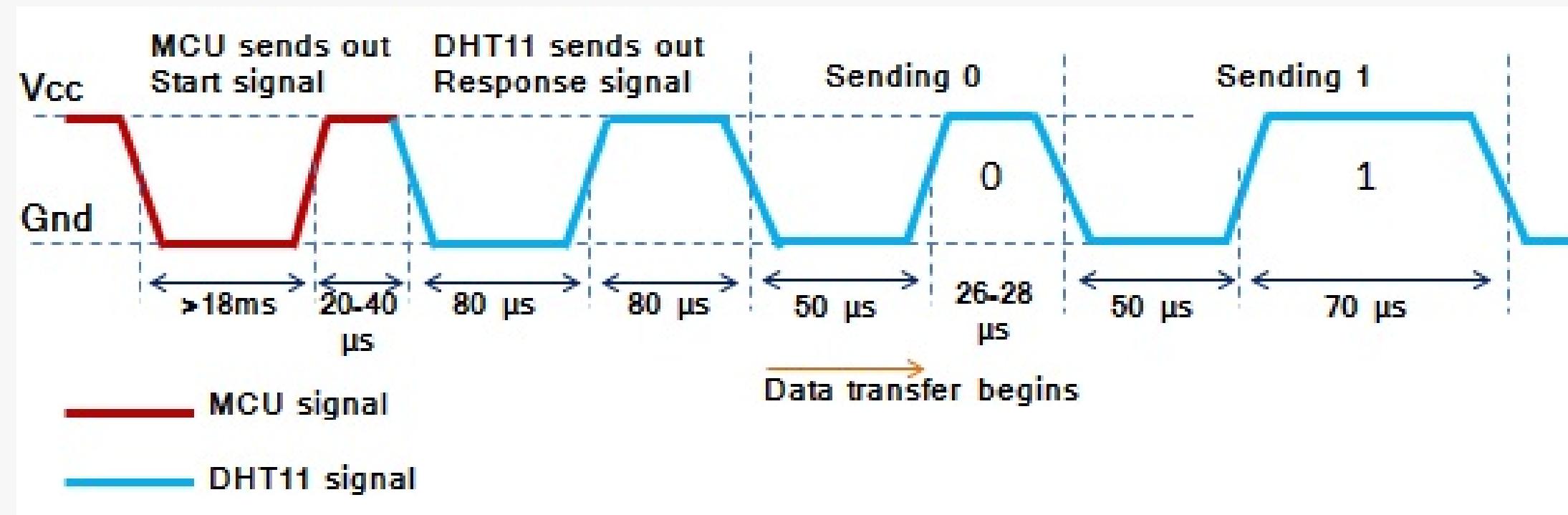
STANDARD PROTOCOLS UART



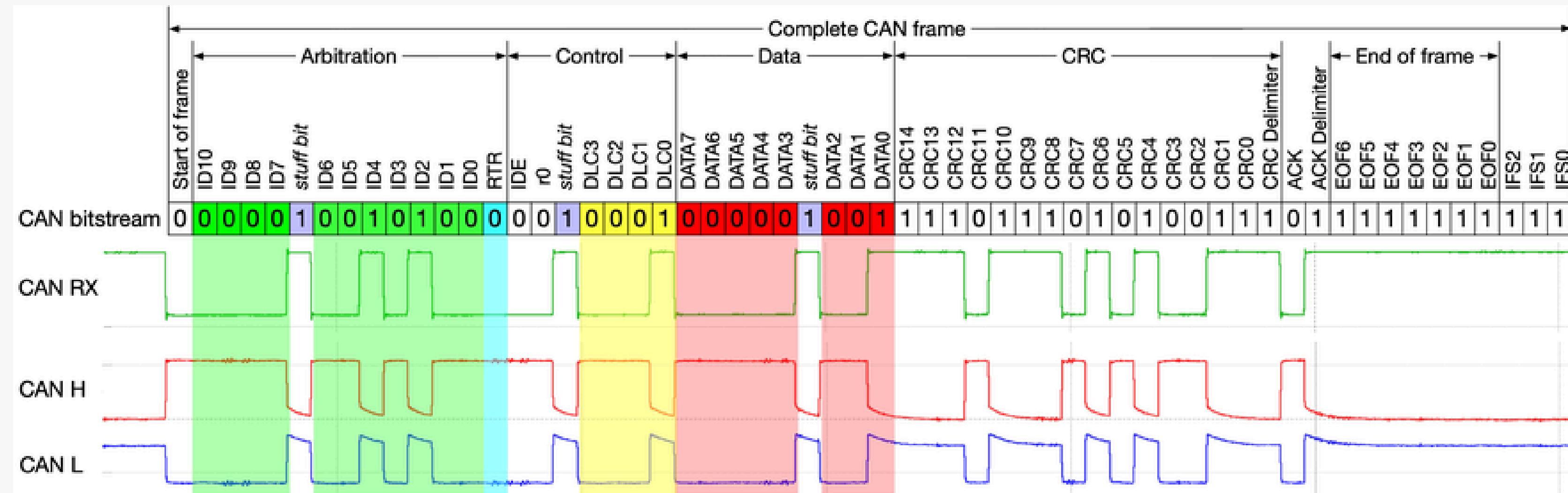
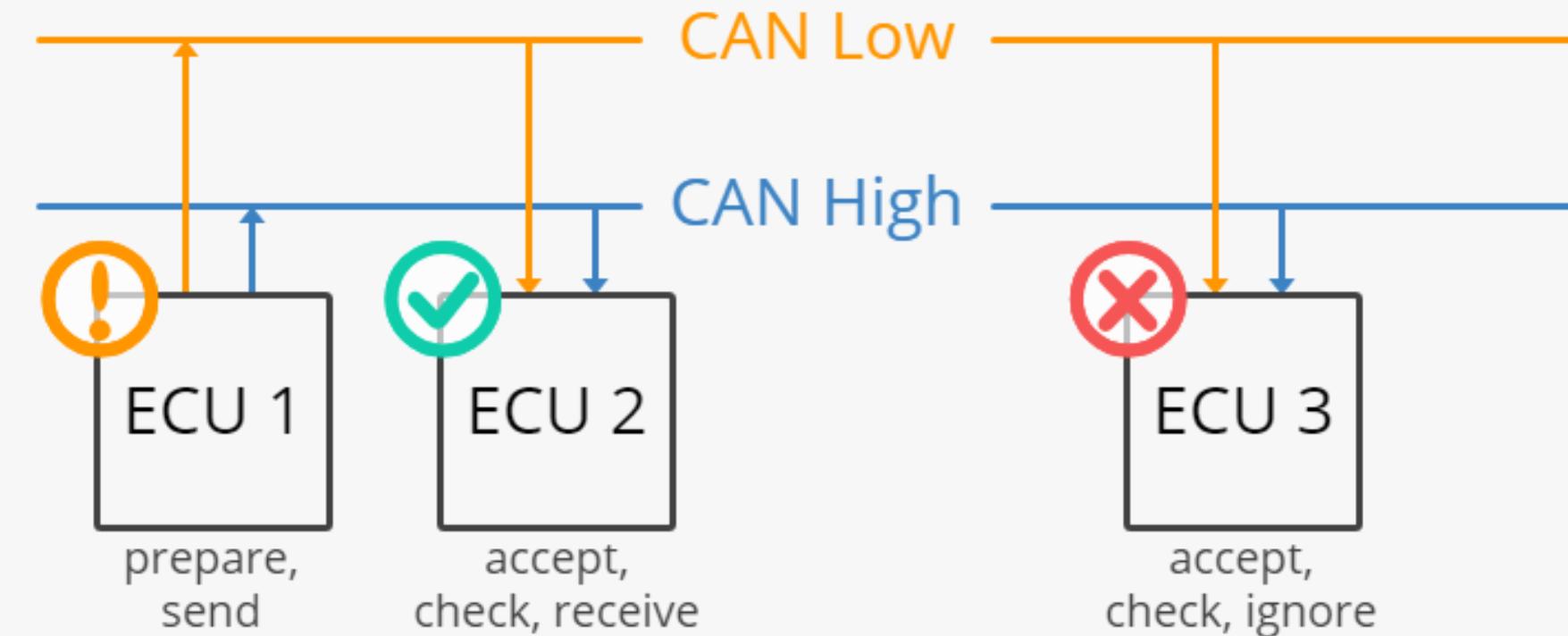
STANDARD PROTOCOLS SPI



STANDARD PROTOCOLS ONEWIRE

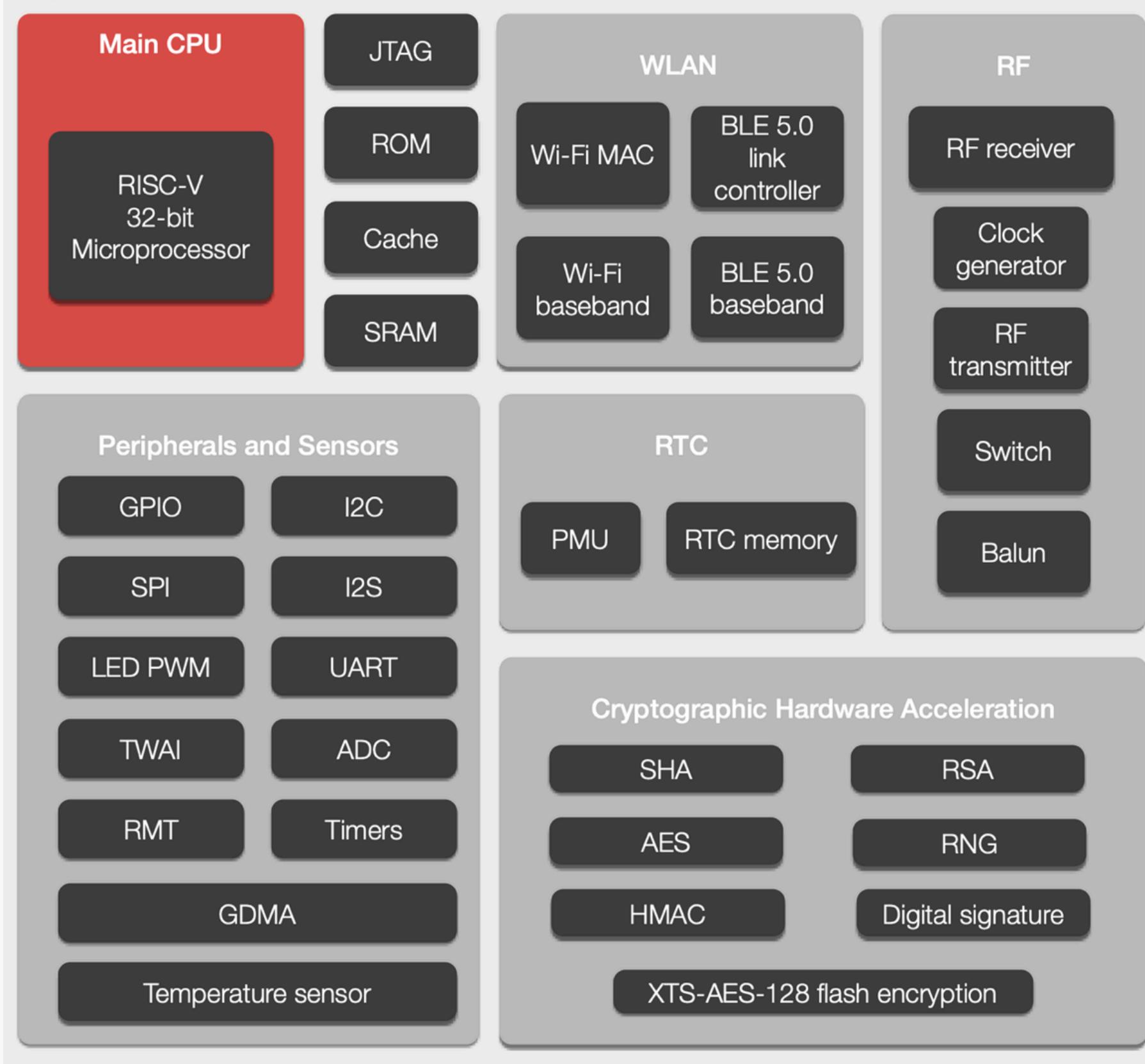


STANDARD PROTOCOLS TWAI(CAN BUS)



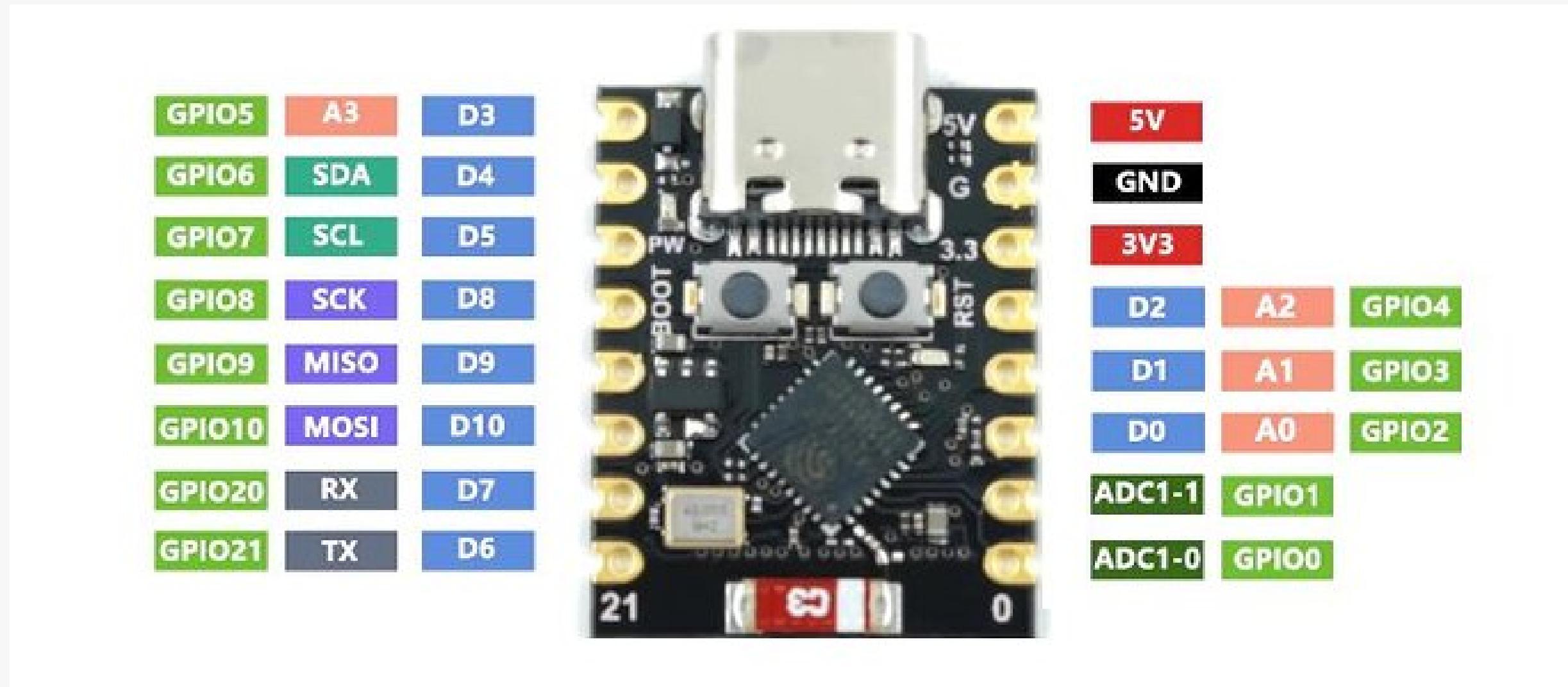
CHIP ARCHITECTURE

Espressif's ESP32-C3 Wi-Fi + BLE SoC



Block Diagram of ESP32-C3

PINOUT



THANK YOU

FOR COMING