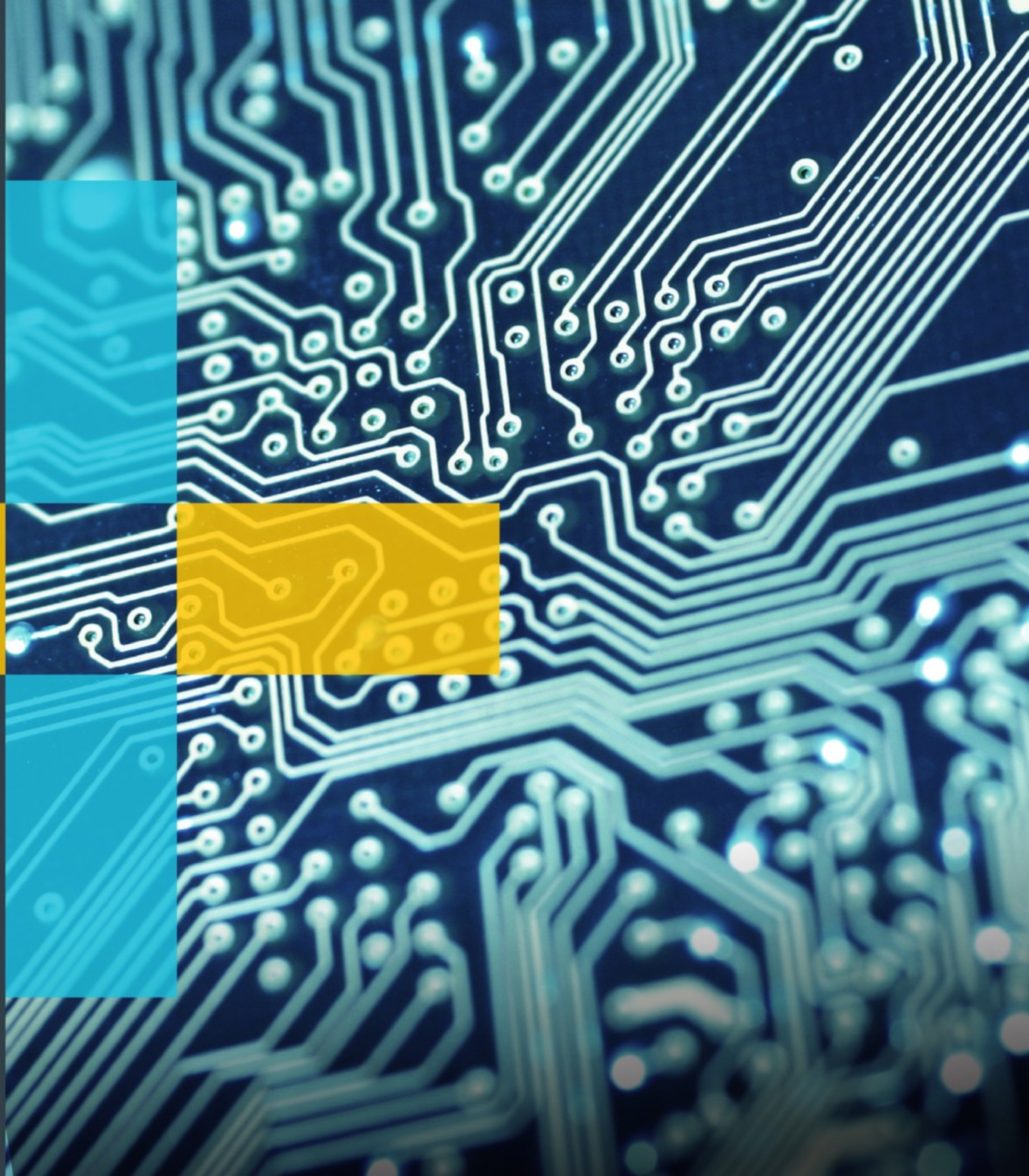
Simulating microcontrollers



Jan Jongboom

Principal Developer Evangelist

Arm

Copyright © 2018 Arm, All rights reserved.

Jan Jongboom

Principal Developer Evangelist

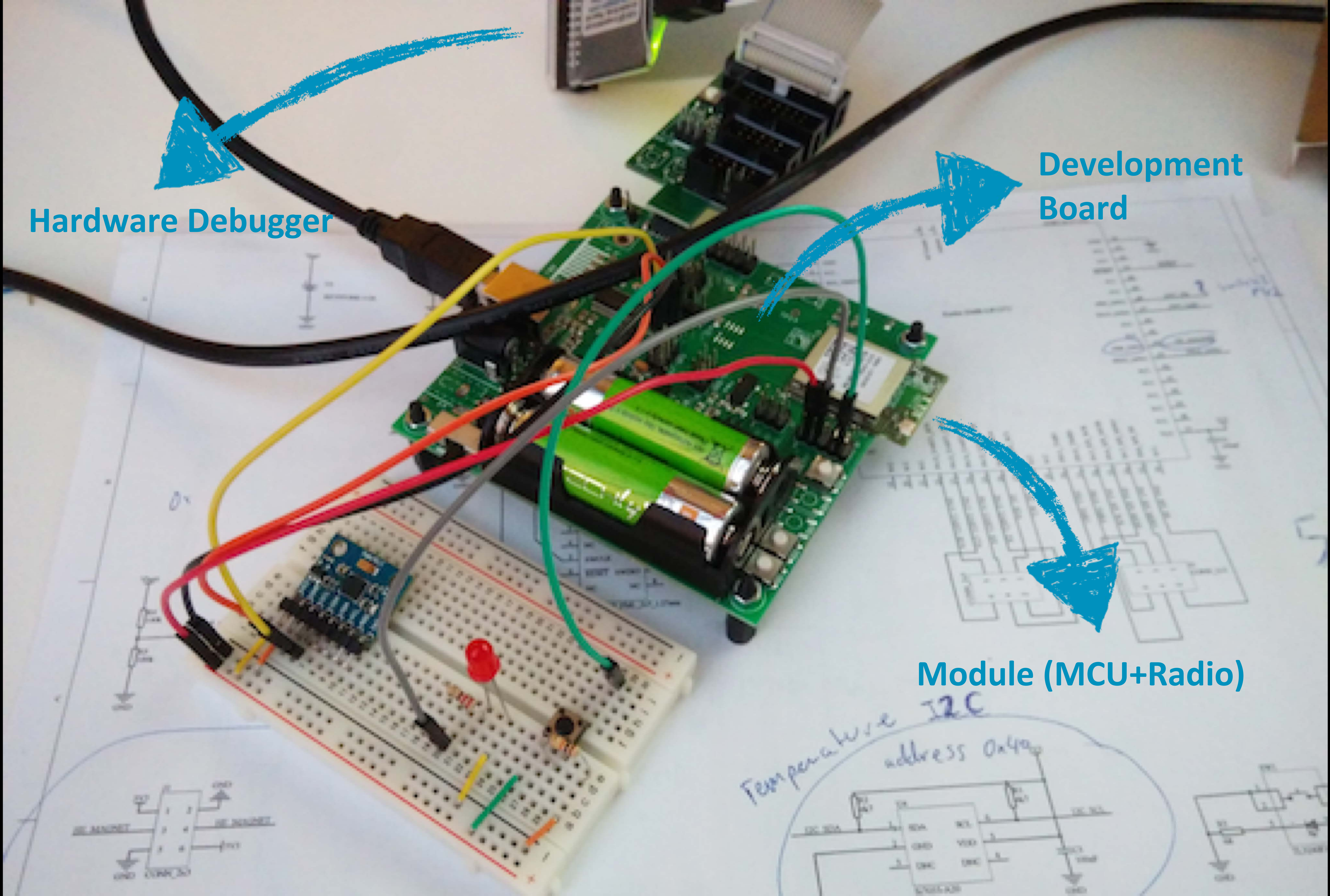
jan.jongboom@arm.com

Web development

💚

MCU development

😩



Hardware Debugger

Development Board

Module (MCU+Radio)

Embedded development is stuck in the 90s

Everything ASM / C / C++

Development tools only run on Windows

Paid debugger with limited number of breakpoints

Very slow feedback loop

https://www.flickr.com/photos/65290859@N05/5995262928

Copyright © 2018 Arm, All rights reserved.

BBC micro:bit

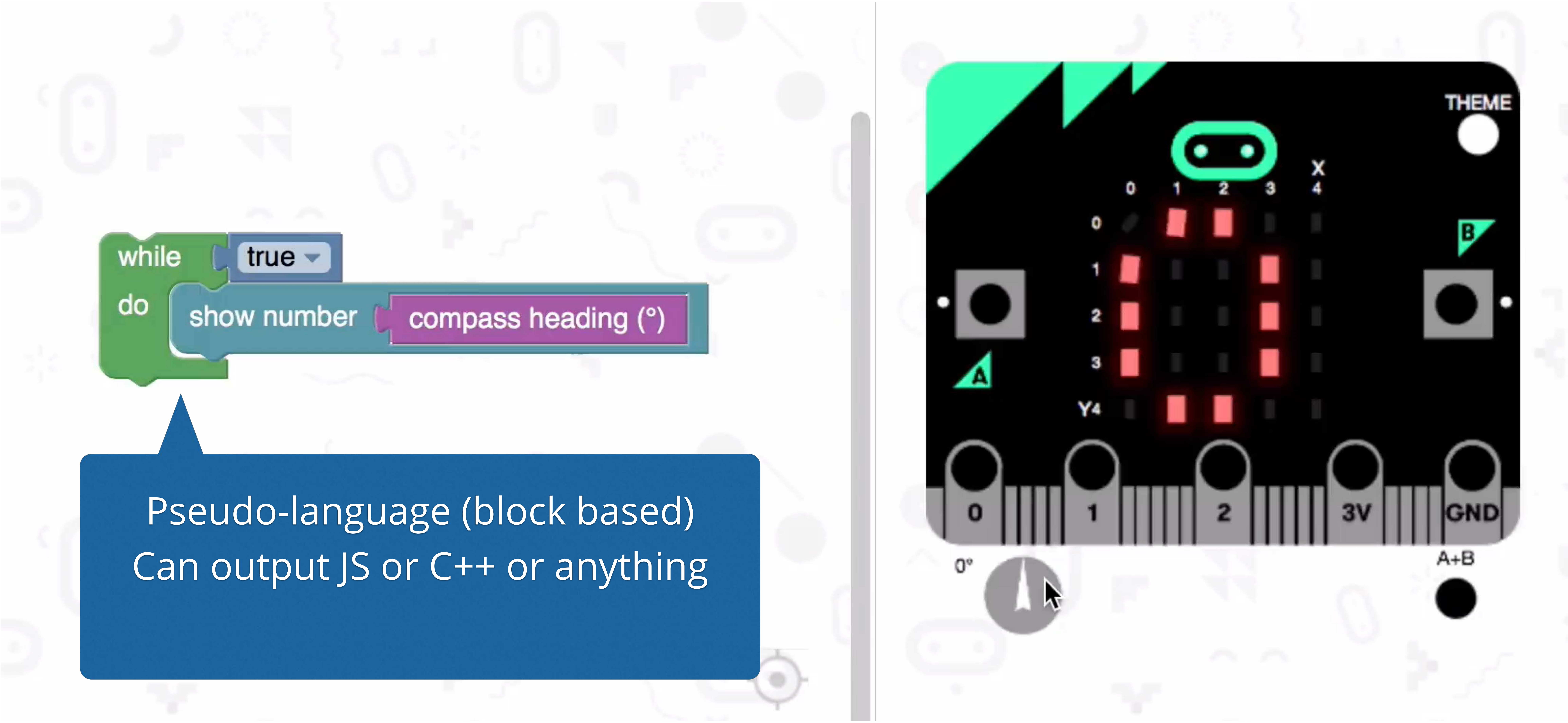
Given to 1 mio kids in UK

True microcontroller (16K of RAM, runs Mbed)

Unique developer experience

https://pxhere.com/en/photo/1396499

BBC micro:bit - online editor

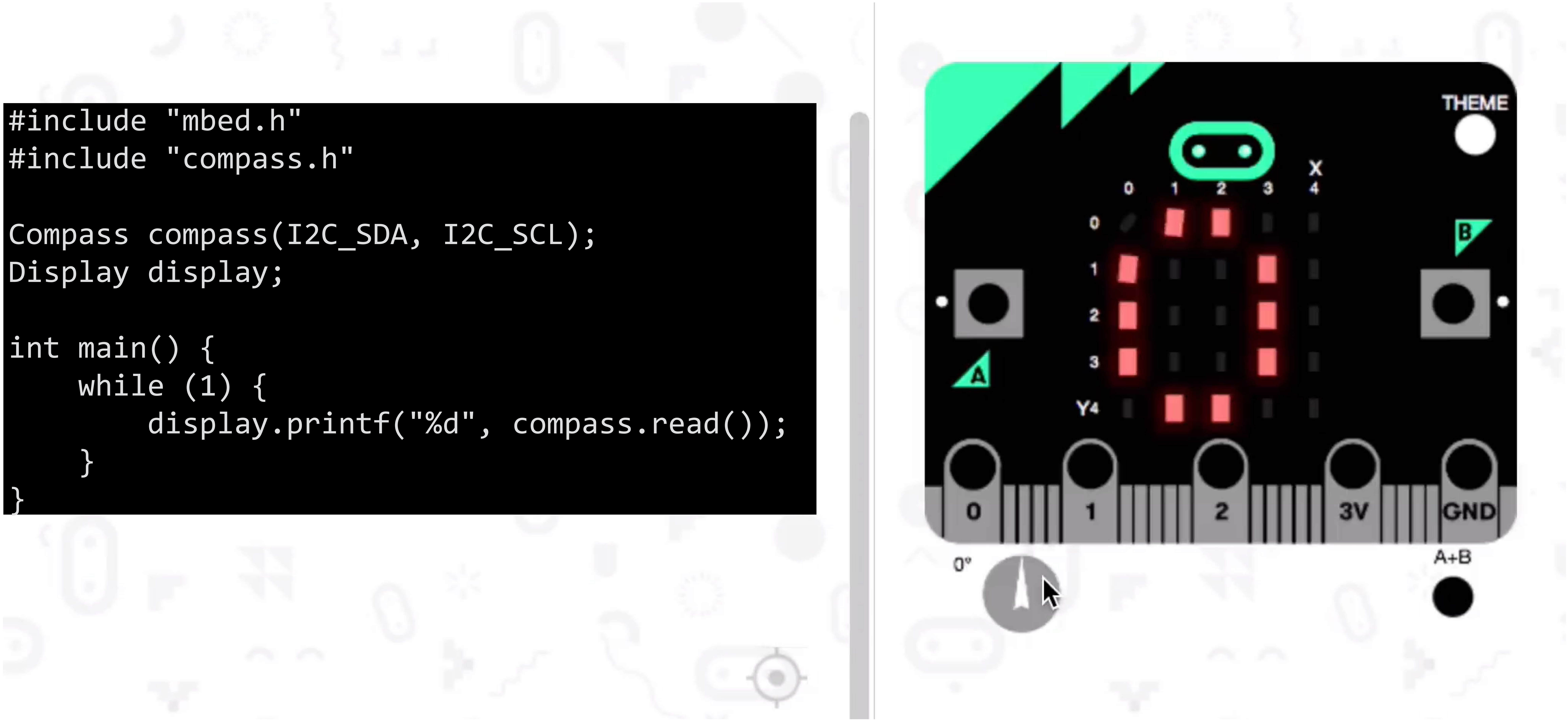


```
while true
do
    show number compass heading (°)
```

Pseudo-language (block based)
Can output JS or C++ or anything

Copyright © 2018 Arm, All rights reserved.

I want this... But in normal C++



```
#include "mbed.h"
#include "compass.h"

Compass compass(I2C_SDA, I2C_SCL);
Display display;

int main() {
    while (1) {
        display.printf("%d", compass.read());
    }
}
```

Copyright © 2018 Arm, All rights reserved.

Emscripten

LLVM to asm.js / WebAssembly

Runs C/C++ in the browser

Supported by Mozilla

unity

UNREAL ENGINE

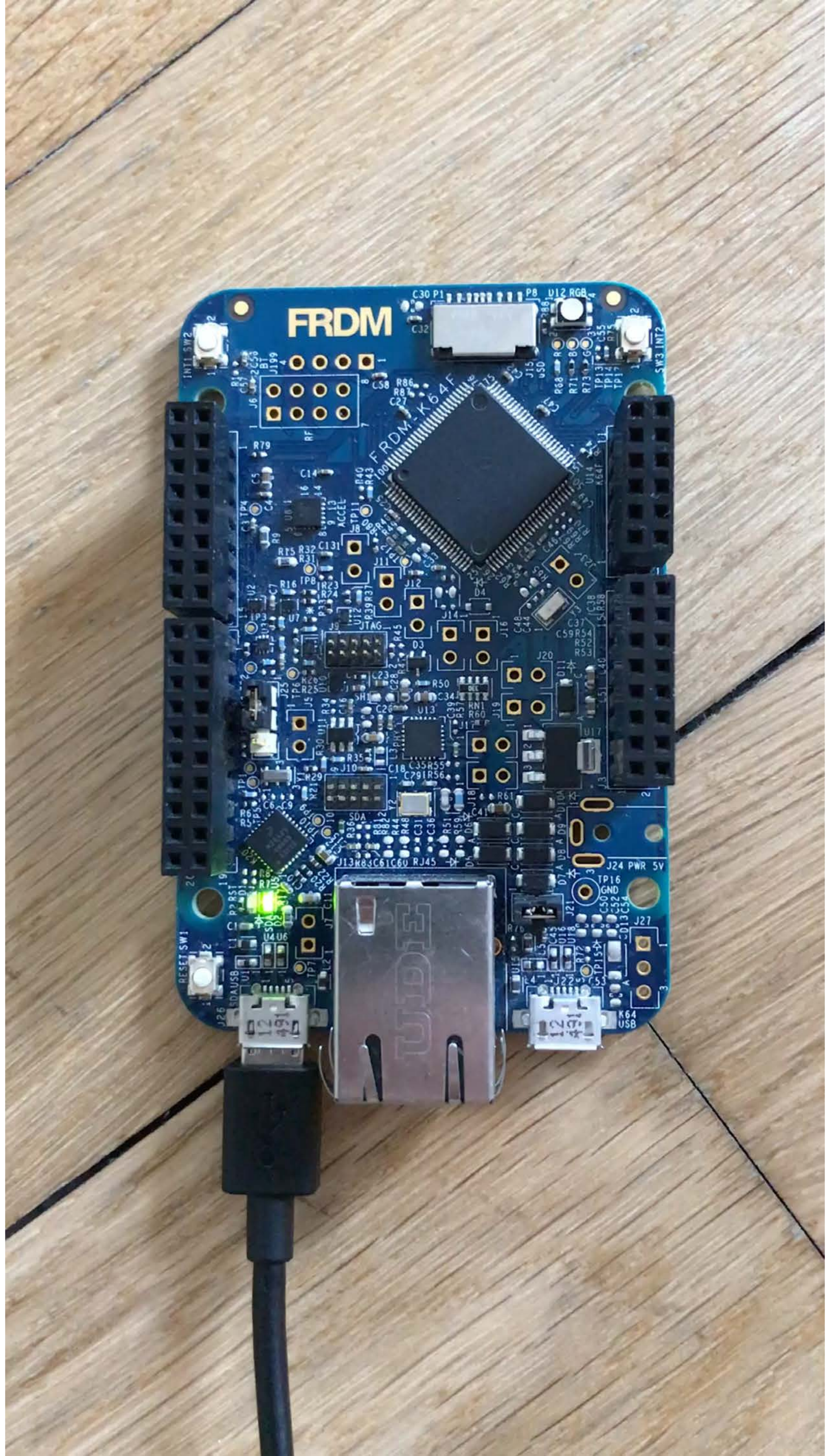
SQLite

AUTODESK® FORMIT®

Copyright © 2018 Arm, All rights reserved.

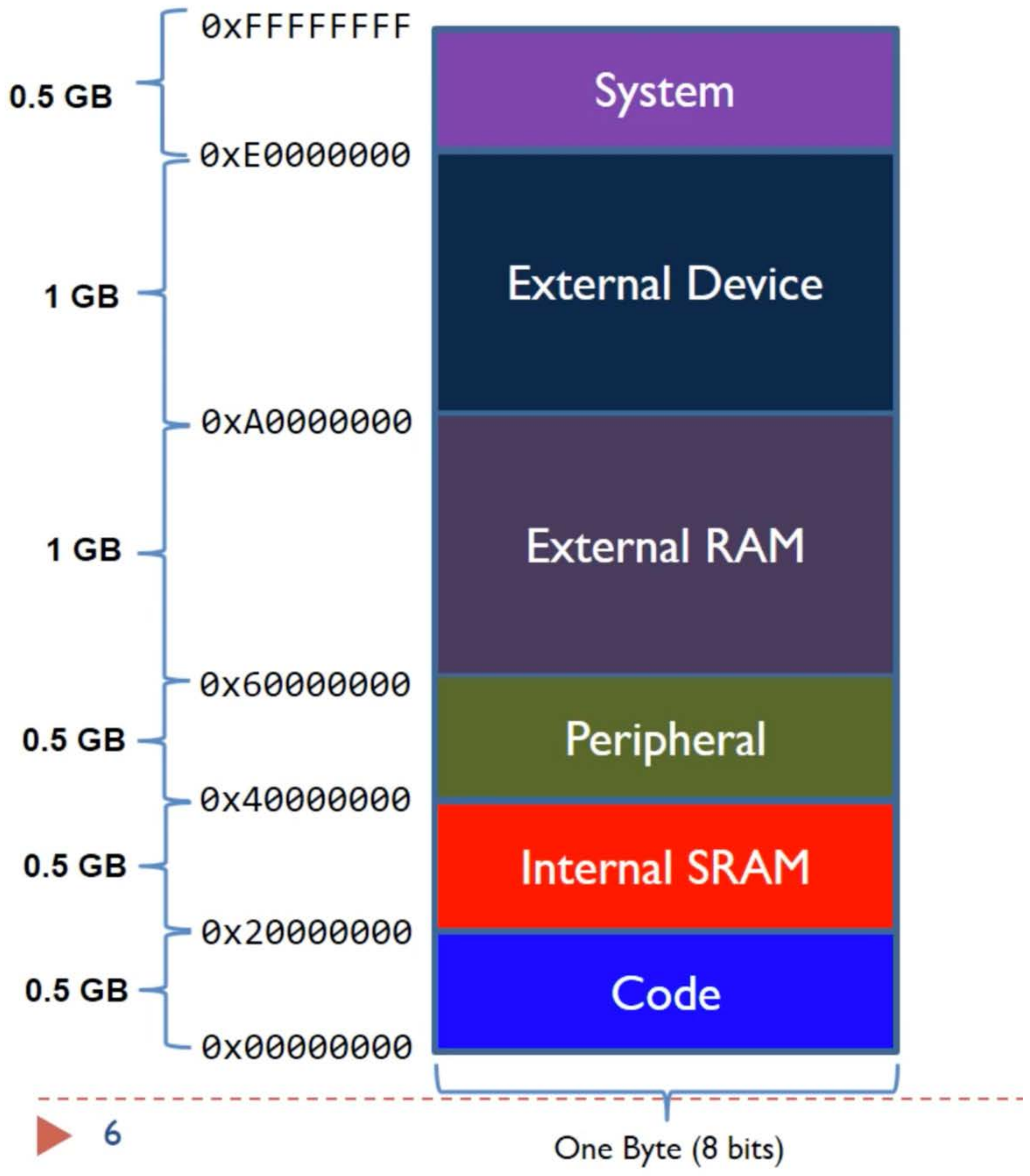
Mbed OS Blinky - User API

```
#include "mbed.h”

DigitalOut led(LED1);

int main() {
    while (1) {
        led = !led;
        wait_ms(500);
    }
}
```



Copyright © 2018 Arm, All rights reserved.

Mbed OS Blinky - under the covers

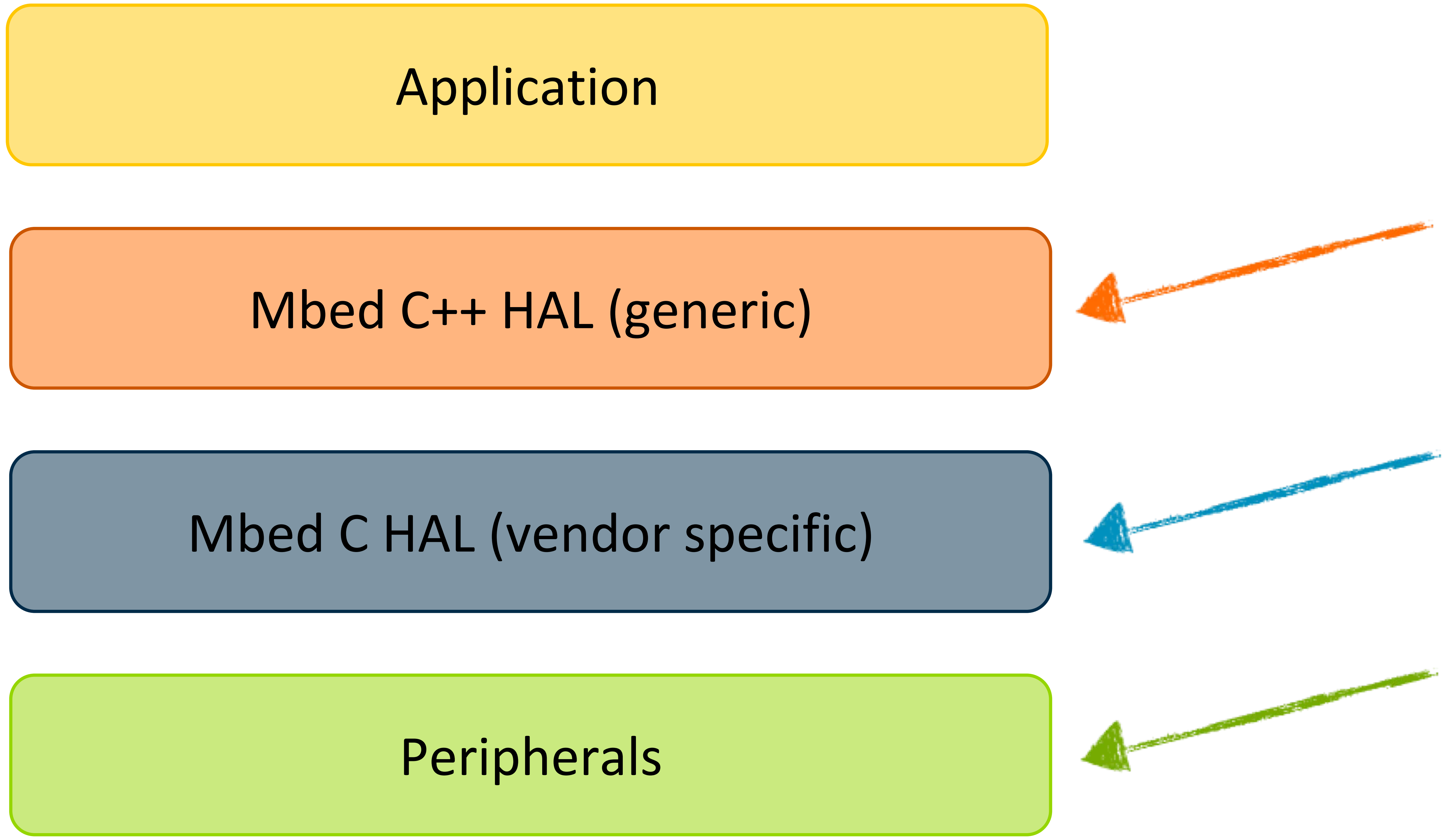


```
// Mbed C++ HAL
DigitalOut::write(int value) {
    gpio_write(&gpio, value);
}

// Mbed C HAL (implemented by vendor)
void gpio_write(gpio_t *gpio, int value) {
    if (value == 1) {
        0x40200000 = 1U << gpio->pin;
    }
    else {
        0x40200004 = 1U << gpio->pin;
    }
}
```

Cortex-M boot sequence explanation: https://www.youtube.com/watch?v=3brOzLJmeek

Copyright © 2018 Arm, All rights reserved.

Places for simulation



Application

Mbed C++ HAL (generic)

Mbed C HAL (vendor specific)

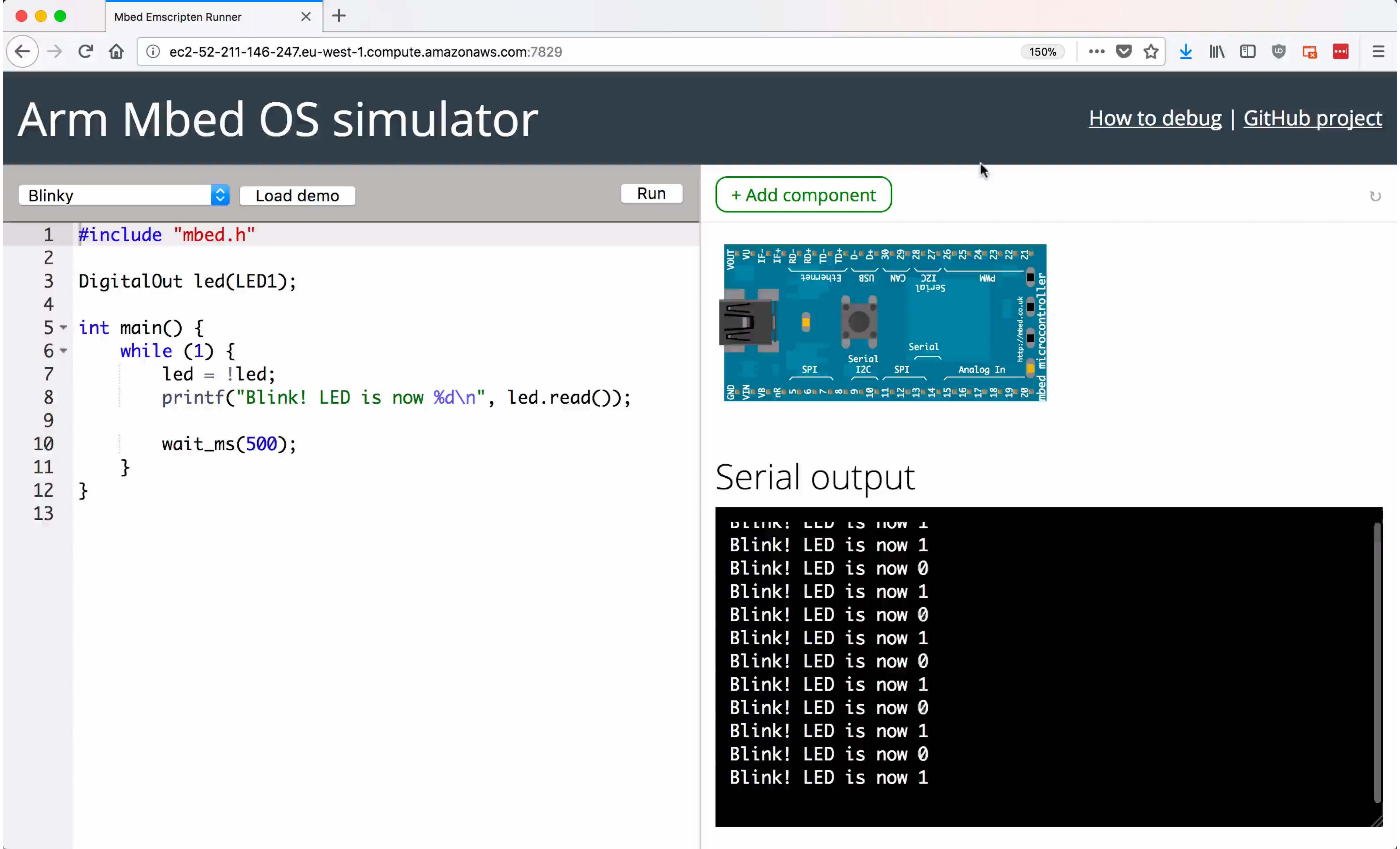
Peripherals

When DigitalOut::write is called
call JavaScript

When gpio_write() is called
call JavaScript 🎉

When write to 0x40200000 comes in
call JavaScript

Copyright © 2018 Arm, All rights reserved.


Mbed Emscripten Runner
ec2-52-211-146-247.eu-west-1.compute.amazonaws.com:7829
150%
Arm Mbed OS simulator
How to debug | GitHub project
Blinky
Load demo
Run
+ Add component
VOUT VU IF- IF+ RD- RD+ TD- TD+ D- D+ 30 29 28 27 26 25 24 23 22 21
Ethernet USB CAN I2C Serial PWM
Serial I2C SPI SPI Serial Analog In
http://mbed.co.uk
mbed microcontroller
GND VIN VB nR 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

```
#include "mbed.h"

DigitalOut led(LED1);

int main() {
    while (1) {
        led = !led;
        printf("Blink! LED is now %d\n", led.read());

        wait_ms(500);
    }
}
```

Serial output

```
Blink! LED is now 1
Blink! LED is now 1
Blink! LED is now 0
Blink! LED is now 1
Blink! LED is now 0
Blink! LED is now 1
Blink! LED is now 0
Blink! LED is now 1
Blink! LED is now 0
Blink! LED is now 1
Blink! LED is now 0
Blink! LED is now 1
```

Challenges

Microcontroller is always busy

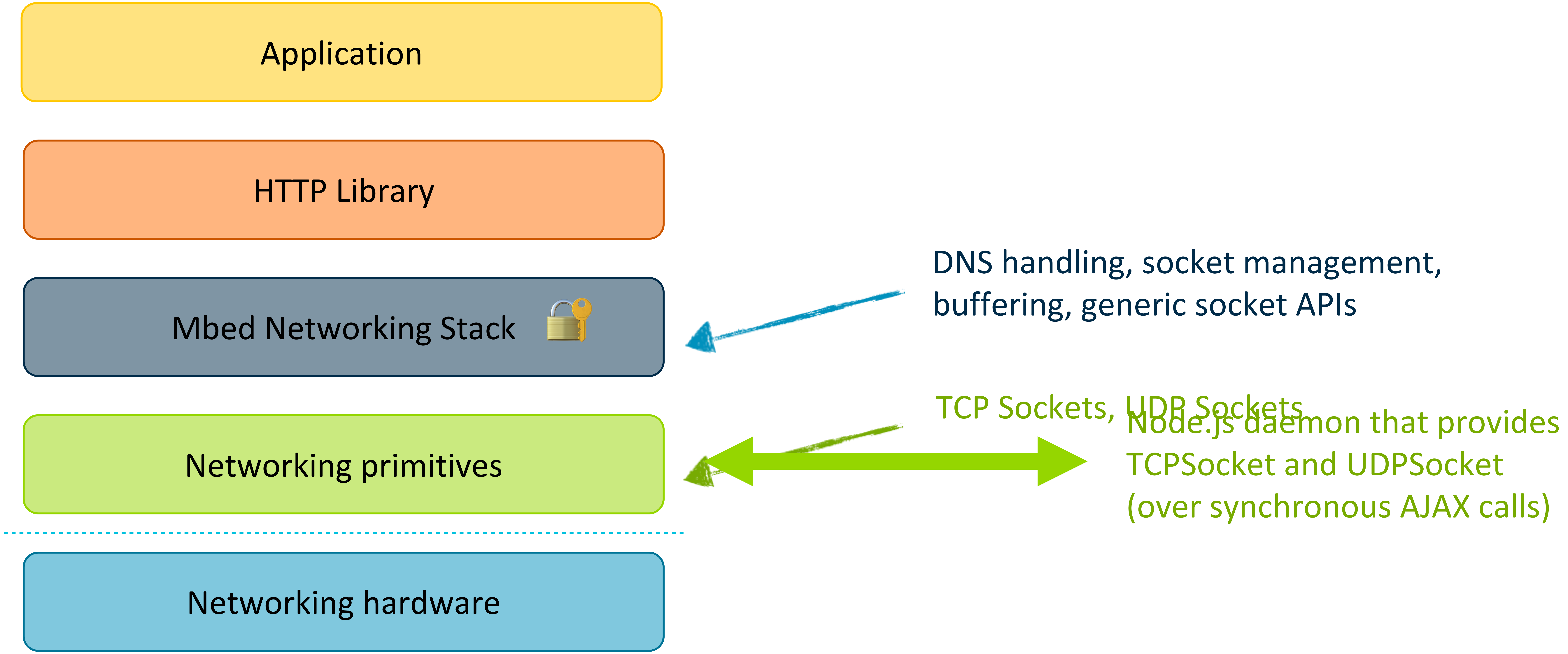
Interrupt handling

Single-threaded

Networking

Copyright © 2018 Arm, All rights reserved.

IP Networking



Application

HTTP Library

Mbed Networking Stack

Networking primitives

Networking hardware

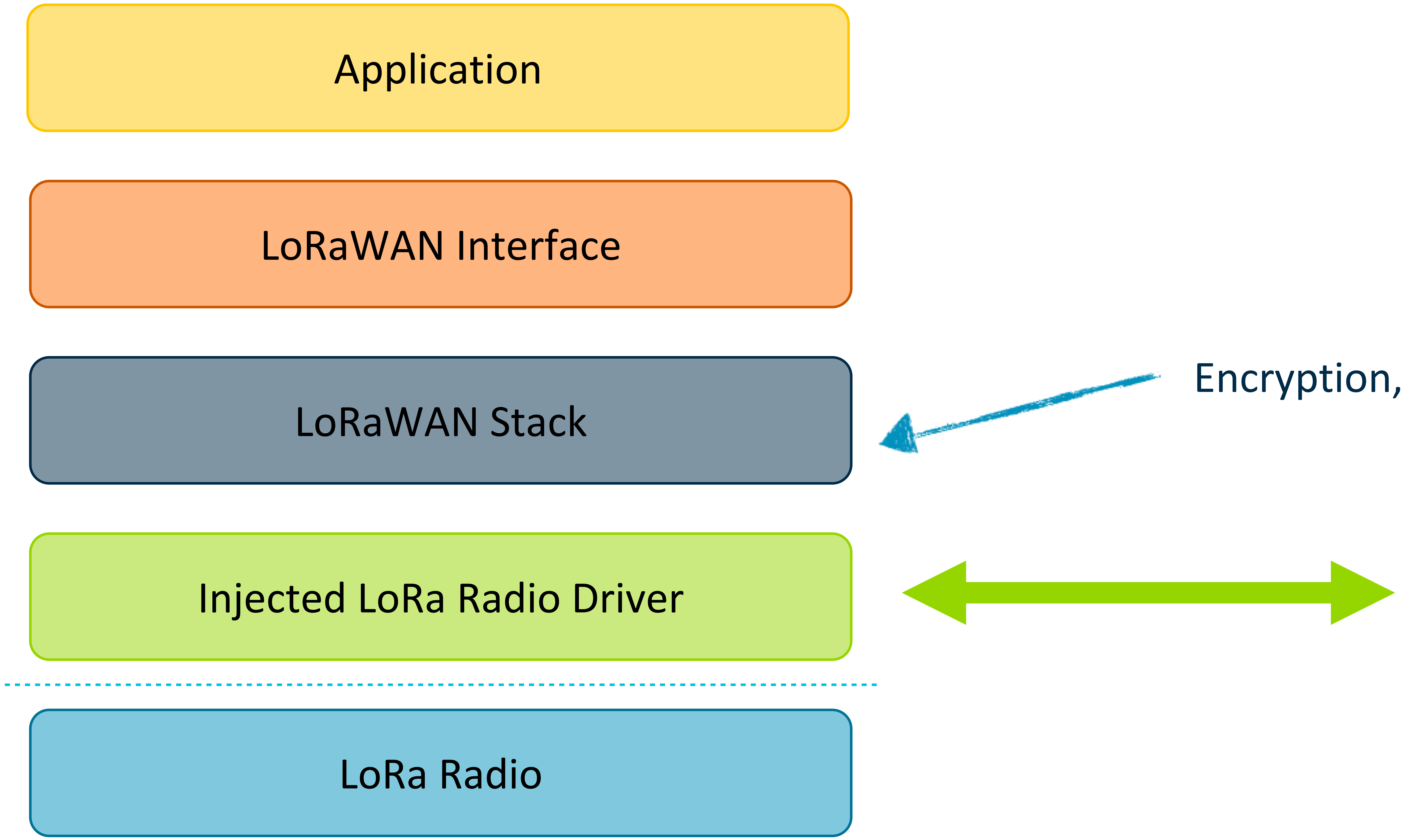
DNS handling, socket management, buffering, generic socket APIs

TCP Sockets, UDP Sockets

Node.js daemon that provides TCPSocket and UDPSocket (over synchronous AJAX calls)

Copyright © 2018 Arm, All rights reserved.

LPWAN networking



Application

LoRaWAN Interface

LoRaWAN Stack

Injected LoRa Radio Driver

LoRa Radio

Encryption, join procedure, frequency hopping

Node.js daemon implementing
LoRaWAN gateway protocol
(Network sees real device / real gateway)

Copyright © 2018 Arm, All rights reserved.

Arm Mbed OS simulator
How to debug | GitHub project
LoRaWAN
Load demo
Run
+ Add component
SHT31 (p28, p27)
20.50°C
30.00%
THE THINGS NETWORK
CONSOLE COMMUNITY EDITION
Applications
Gateways
Support
armmbed
Applications › mbed-simulator › Devices › simulator-1 › Data
Overview
Data
Settings
APPLICATION DATA
pause
clear
Filters
uplink
downlink
activation
ack
error
time
counter
port

```
#include "mbed.h"
#include "mbed_trace.h"
#include "mbed_events.h"
#include "LoRaWANInterface.h"
#include "Sht31.h"
#include "SX1276_LoRaRadio.h"

// Device credentials, register device as OTAA in The Things Network and c
static uint8_t DEV_EUI[] = { 0x00, 0xA9, 0x9D, 0x49, 0x21, 0xB2, 0x6D, 0x7
static uint8_t APP_EUI[] = { 0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x00, 0xC1, 0x8
static uint8_t APP_KEY[] = { 0xE1, 0x13, 0x6D, 0x7E, 0xB6, 0x91, 0x7F, 0xC


// The port we're sending and receiving on
#define MBED_CONF_LORA_APP_PORT     15

// Peripherals (LoRa radio, temperature sensor and button)
SX1276_LoRaRadio radio(D11, D12, D13, D10, A0, D2, D3, D4, D5, D8, D9, NC,
Sht31 sht31(I2C_SDA, I2C_SCL);
InterruptIn btn(BUTTON1);

// EventQueue is required to dispatch events around
static EventQueue ev_queue;

// Constructing Mbed LoRaWANInterface and passing it down the radio object
static LoRaWANInterface lorawan(radio);

// Application specific callbacks
static lorawan_app_callbacks_t callbacks;

// LoRaWAN stack event handler
static void lora_event_handler(lorawan_event_t event);

// Send a message over LoRaWAN
static void send_message() {
    uint8_t tx_buffer[50] = { 0 };

    // Sending strings over LoRaWAN is not recommended
    sprintf((char*) tx_buffer, "Temperature = %3.1f",
                                       sht31.readTemperature());

    int packet_len = strlen((char*) tx_buffer);

    printf("Sending %d bytes: \"%s\"\n", packet_len, tx_buffer);

    int16_t retcode = lorawan.send(MBED_CONF_LORA_APP_PORT, tx_buffer, pac

    // for some reason send() returns -1... I cannot find out why, the sta
    if (retcode < 0) {
        retcode == LORAWAN_STATUS_WOULD_BLOCK ? printf("send - duty cycle
                    : printf("send() - Error code %d\n", retcode);
        return;
    }

    printf("%d bytes scheduled for transmission\n", retcode);
}

int main() {
```

Serial output

```
Press BUTTON1 to send the current value of the temperature sens
[DBG ][LSTK]: Initializing MAC layer
[DBG ][LSTK]: Initiating OTAA
[DBG ][LSTK]: Sending Join Request ...
[DBG ][LMAC]: Frame prepared to send at port 0
[DBG ][LMAC]: TX: Channel=2, DR=5
[DBG ][LRAD]: transmit channel=868500000 power=13 bandwidth=7 d
Connection - In Progress ...
[DBG ][LSTK]: Transmission completed
[DBG ][LRAD]:  ][LSTK]: Transmission completed
[DBG ][LMAC]: Opening RX1 Window
[DBG ][LSTK]: OTAA Connection OK!
Connection - Successful
```

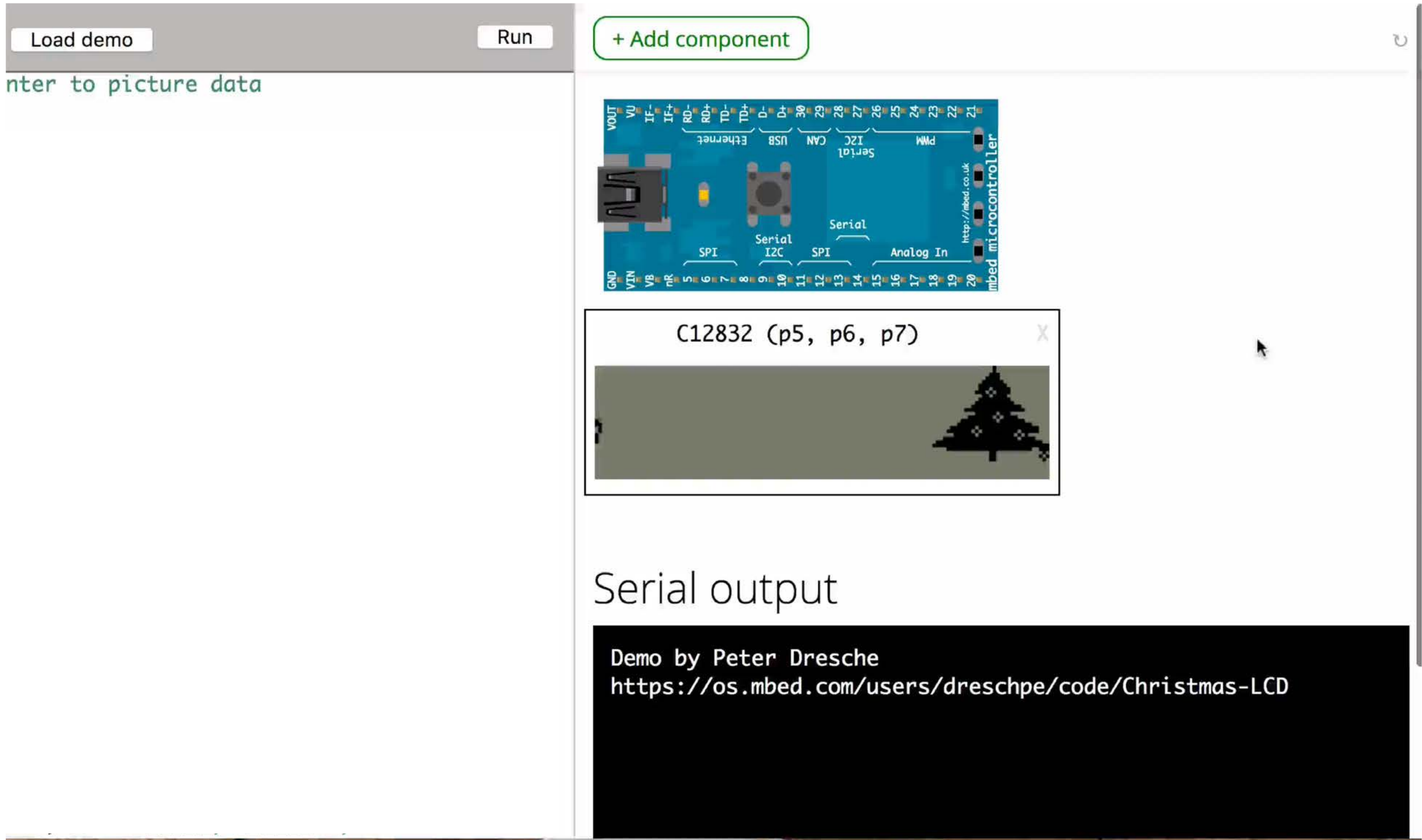
Bringing in new components

Take C++ library

Add in a few EM_ASM calls in strategic places
(#ifdef TARGET_SIMULATOR)

F.e. when flushing frame buffer over SPI for a display

Implement JS HAL and JS UI
(e.g. draw canvas and handle frame buffer updates)

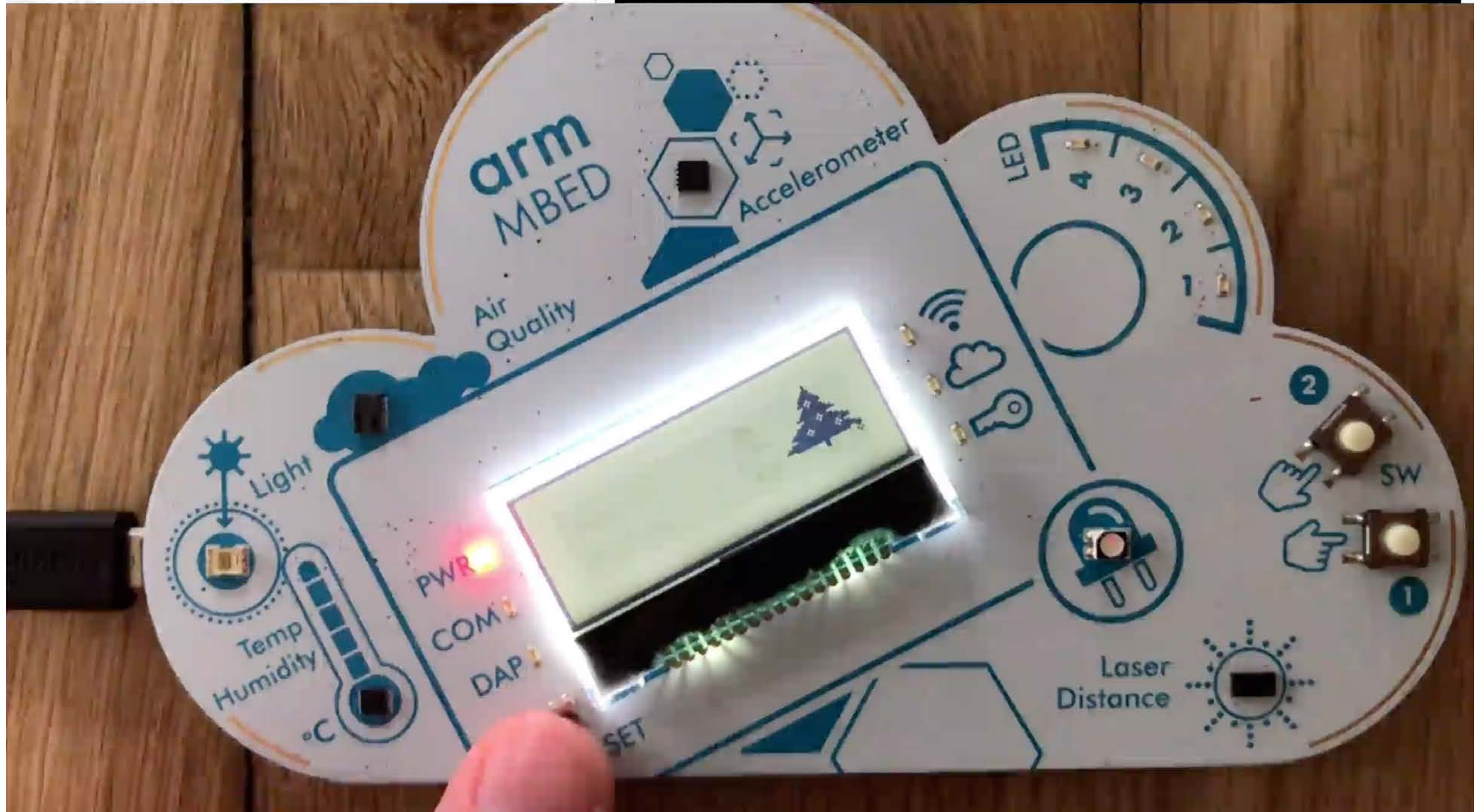


```
nter to picture data
```

C12832 (p5, p6, p7)

Serial output

```
Demo by Peter Dresche
https://os.mbed.com/users/dreschpe/code/Christmas-LCD
```



Copyright © 2018 Arm, All rights reserved.

Two ways of using the simulator

Online: https://labs.mbed.com/simulator

- Great for teaching
- Online compiler
- Share code by copy pasting the URL

Offline (requires Emscripten SDK):

```
$ npm install mbed-simulator -g
$ mbed-simulator .
```

Copyright © 2018 Arm, All rights reserved.

main.cpp — mbed-os-example-blinky
EXPLORER
OPEN EDITORS
main.cpp
MBED-OS-EXAMPLE-BLIN...
.vscode
BUILD
mbed-os
.gitignore
.mbed
main.cpp
mbed_app.json
mbed_settings.py
mbed_settings.p...
mbed-os.lib
README.md
main.cpp
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
1: bash

```
#include "mbed.h"

DigitalOut led1(LED1);

int main() {
    while (true) {
        led1 = !led1;
        printf("Blink! LED is now %d\n", led1.read());
        wait(0.5);
    }
}
```

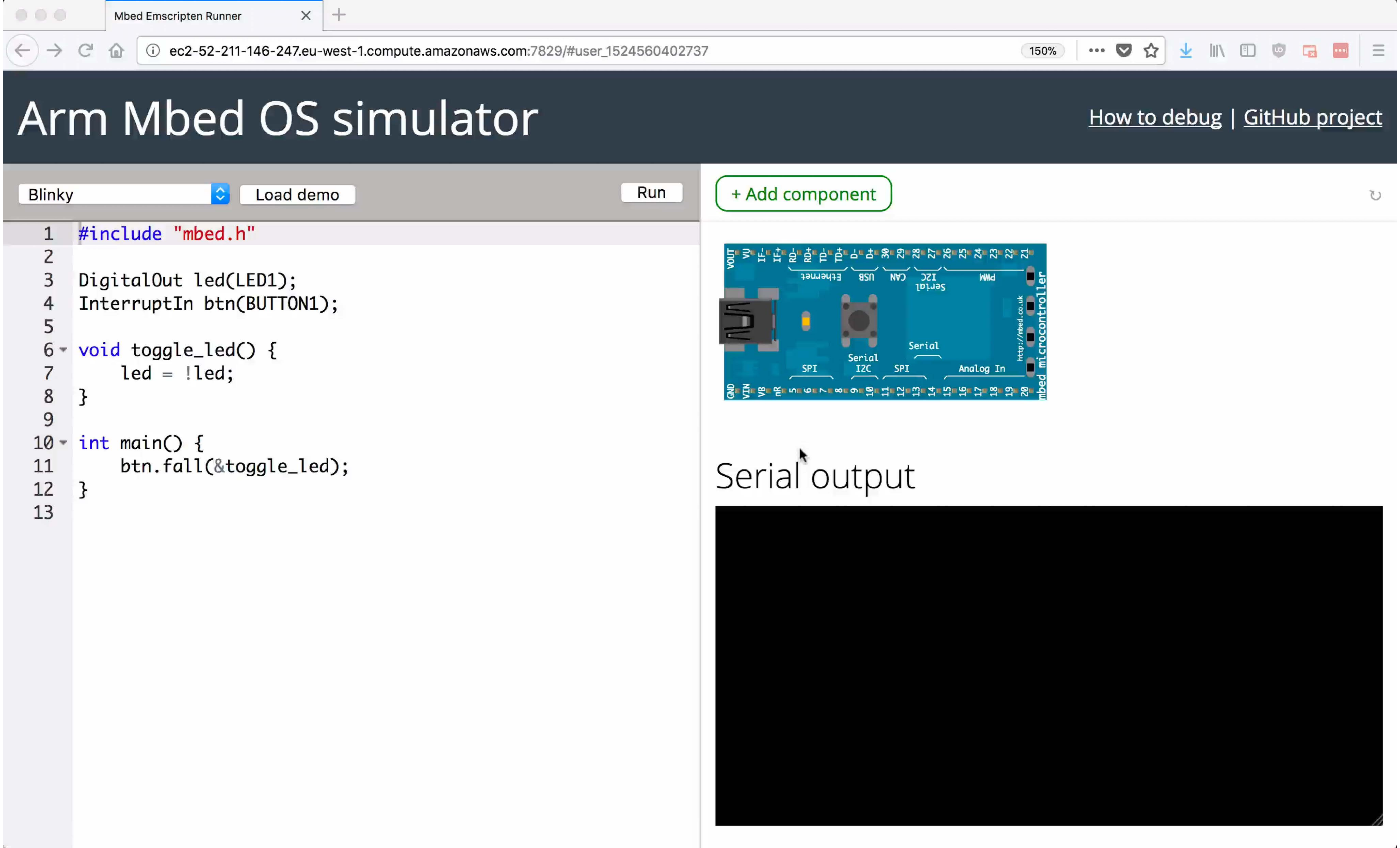
```
janjon01:~/repos/mbed-os-example-blinky (master) $ 
```

Debugging

Source maps can be generated by Emscripten

Debugging through normal browser JS debugger

Copyright © 2018 Arm, All rights reserved.


Mbed Emscripten Runner
ec2-52-211-146-247.eu-west-1.compute.amazonaws.com:7829/#user_1524560402737
150%
How to debug | GitHub project
Blinky
Load demo
Run
+ Add component
VOUT VU IF- IF+ RD- RD+ TD- TD+ D- D+ 30 29 28 27 26 25 24 23 22 21
Ethernet USB CAN I2C Serial PWM
http://mbed.co.uk mbed microcontroller
Serial
SPI Serial I2C SPI Analog In
GND VIN VB nR 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Arm Mbed OS simulator

```
#include "mbed.h"

DigitalOut led(LED1);
InterruptIn btn(BUTTON1);

void toggle_led() {
    led = !led;
}

int main() {
    btn.fall(&toggle_led);
}

```

Serial output

Recap

Embedded development is a PITA

The web has brought so many cool tools, let's use them

Cross-compile your embedded app => ??? => PROFIT

Open source @ https://github.com/janjongboom/mbed-simulator

Live demo @ https://labs.mbed.com

https://www.pexels.com/photo/arduino-breadboard-circuit-diodes-370518/

Trademark and copyright statement
The trademarks featured in this presentation are registered and/or unregistered trademarks of Arm (or its subsidiaries) in the EU and/or elsewhere. All rights reserved. All other marks featured may be trademarks of their respective owners.

Copyright © 2018

Thank You!

Copyright © 2018 Arm, All rights reserved.