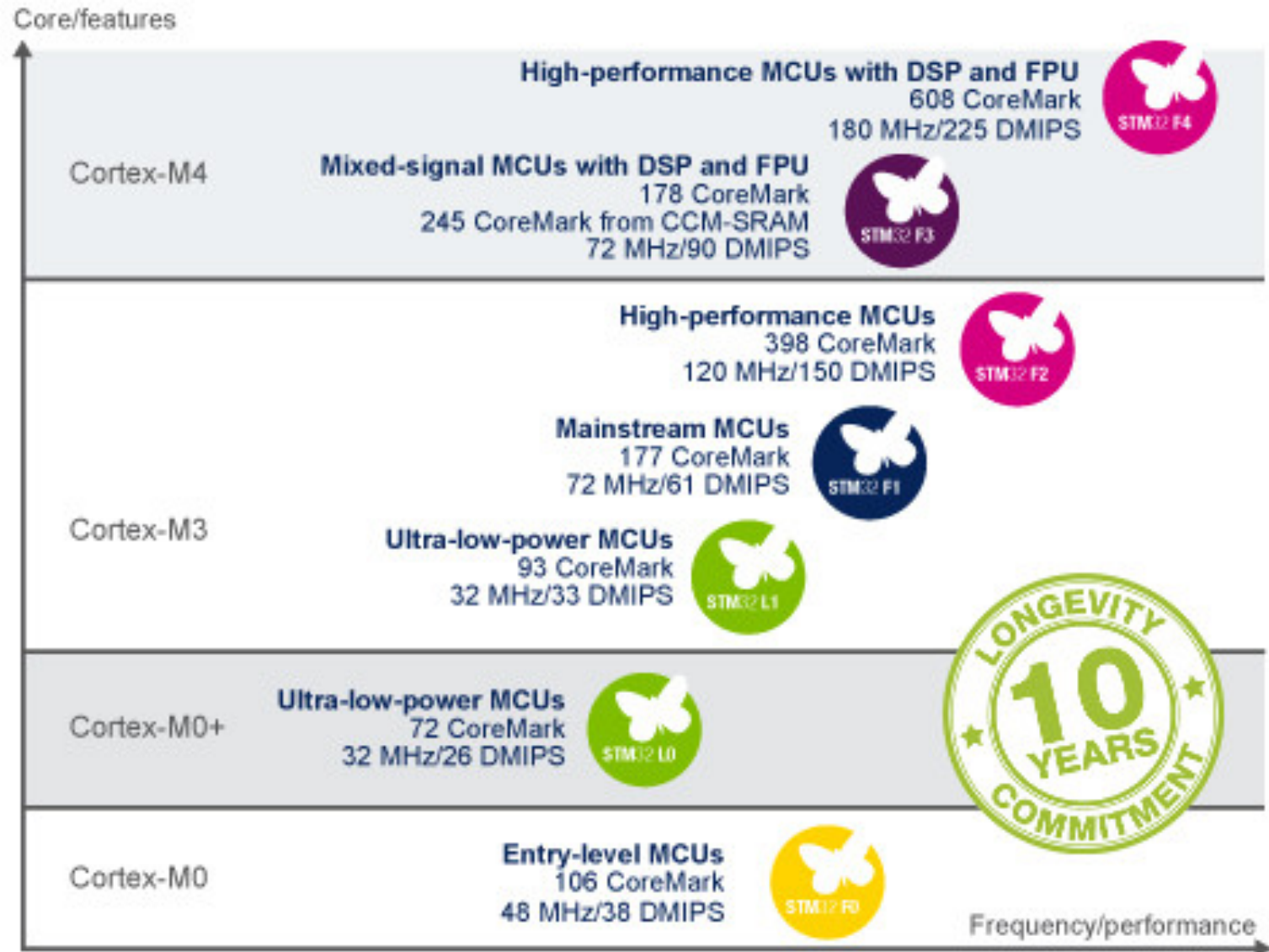# STM32
A Brief Introduction
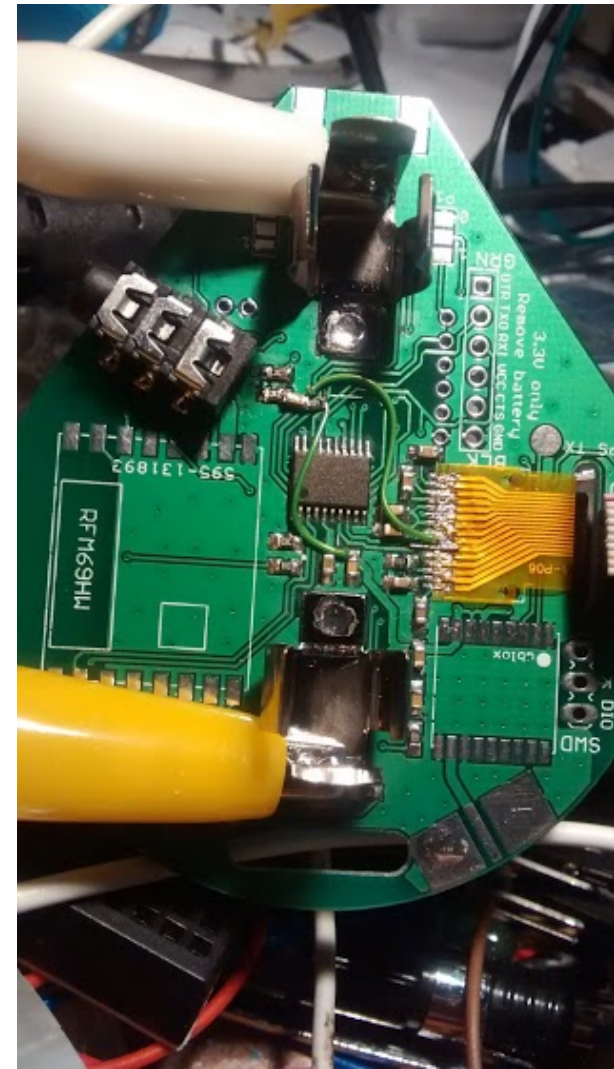
Jon Sowman

What are they?

- Modern microprocessors
- Core designed and licensed by ARM
- 3rd ARM family by ST
  - Add peripherals and manufacture the chip
  - Library support
- 32 bit RISC core (*thumb2*)
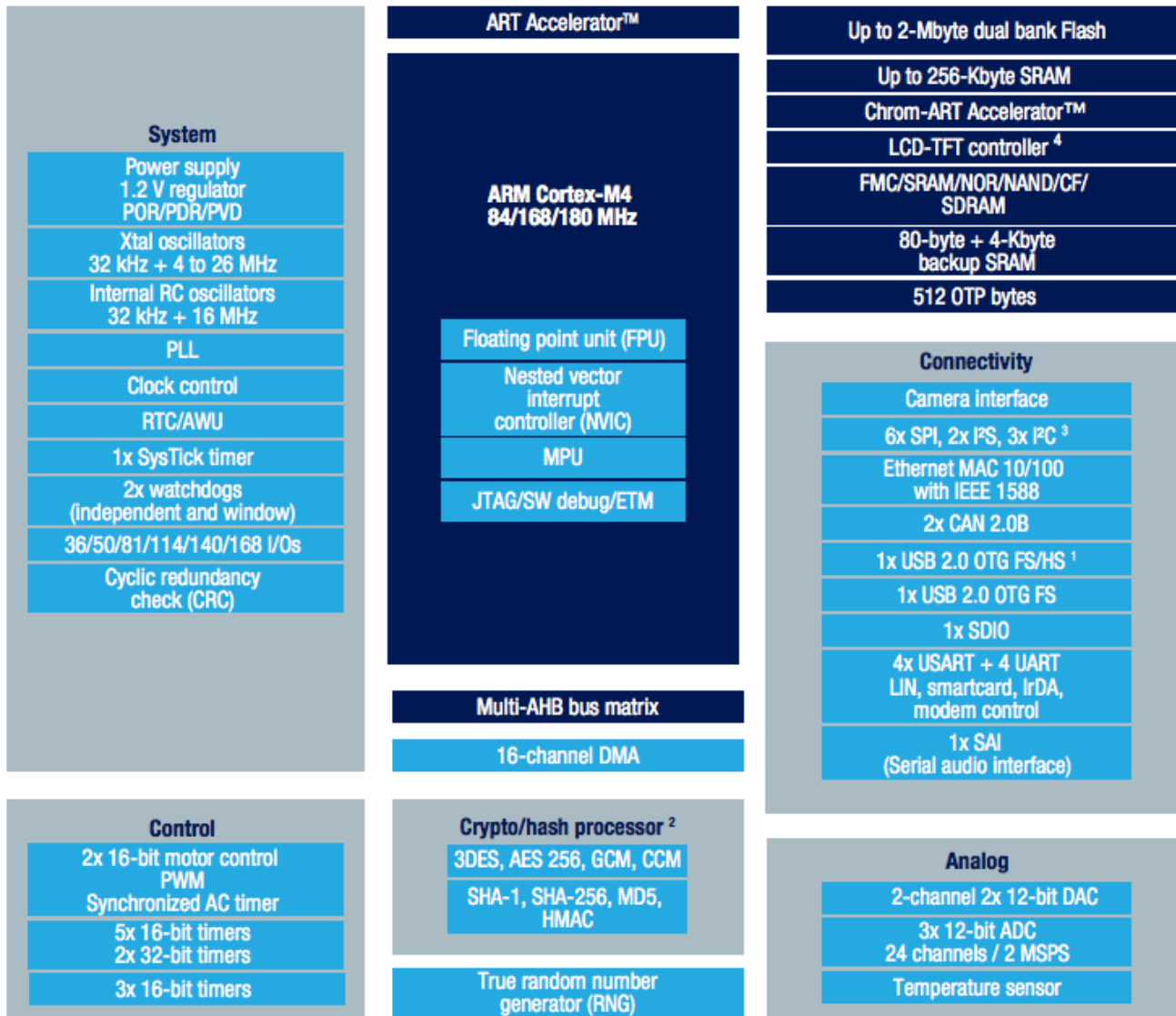- Gathering a hobbyist following

Range

Core/features

High-performance MCUs with DSP and FPU
608 CoreMark
180 MHz/225 DMIPS
STM32 F4

Cortex-M4

Mixed-signal MCUs with DSP and FPU
178 CoreMark
245 CoreMark from CCM-SRAM
72 MHz/90 DMIPS
STM32 F3

High-performance MCUs
398 CoreMark
120 MHz/150 DMIPS
STM32 F2

Mainstream MCUs
177 CoreMark
72 MHz/61 DMIPS
STM32 F1

Cortex-M3

Ultra-low-power MCUs
93 CoreMark
32 MHz/33 DMIPS
STM32 L1

Cortex-M0+

Ultra-low-power MCUs
72 CoreMark
32 MHz/26 DMIPS
STM32 L0

LONGEVITY
10
YEARS
COMMITMENT

Cortex-M0

Entry-level MCUs
106 CoreMark
48 MHz/38 DMIPS
STM32 F0

Frequency/performance

Why are they better?

- Peripherals
  - Higher density
  - Faster/bigger/better
    - NVIC particularly
- Cheap
  - Dev boards are £6-7
  - Badge CM0 is 77p!
- Excellent debugging
  - JTAG/SWD (Serial Wire Debug)
  - Run/halt; timer control; tracepoints/watchpoints
- Fast!
  - M0 48MHz, M3 72MHz, M4 180MHz!

Peripherals (CM4 example)

System
- Power supply 1.2 V regulator POR/PDR/PVD
- Xtal oscillators 32 kHz + 4 to 26 MHz
- Internal RC oscillators 32 kHz + 16 MHz
- PLL
- Clock control
- RTC/AWU
- 1x SysTick timer
- 2x watchdogs (independent and window)
- 36/50/81/114/140/168 I/Os
- Cyclic redundancy check (CRC)

ART Accelerator™

ARM Cortex-M4 84/168/180 MHz
- Floating point unit (FPU)
- Nested vector interrupt controller (NVIC)
- MPU
- JTAG/SW debug/ETM

Multi-AHB bus matrix
- 16-channel DMA

- Up to 2-Mbyte dual bank Flash
- Up to 256-Kbyte SRAM
- Chrom-ART Accelerator™
- LCD-TFT controller [4]
- FMC/SRAM/NOR/NAND/CF/ SDRAM
- 80-byte + 4-Kbyte backup SRAM
- 512 OTP bytes

Connectivity
- Camera interface
- 6x SPI, 2x I²S, 3x I²C [3]
- Ethernet MAC 10/100 with IEEE 1588
- 2x CAN 2.0B
- 1x USB 2.0 OTG FS/HS [1]
- 1x USB 2.0 OTG FS
- 1x SDIO
- 4x USART + 4 UART LIN, smartcard, IrDA, modem control
- 1x SAI (Serial audio interface)

Control
- 2x 16-bit motor control PWM Synchronized AC timer
- 5x 16-bit timers 2x 32-bit timers
- 3x 16-bit timers

Crypto/hash processor [2]
- 3DES, AES 256, GCM, CCM
- SHA-1, SHA-256, MD5, HMAC
- True random number generator (RNG)

Analog
- 2-channel 2x 12-bit DAC
- 3x 12-bit ADC 24 channels / 2 MSPS
- Temperature sensor

Peripheral Libraries

- Peripherals are flexible but complex!
  - e.g. Advanced Control Timers (20 x 32bit registers)
  - Use of a library eases development (c.f. AVR)

- Standard Peripheral Library
  - Developed by ST

- libopencm3
  - Open source, community developed project
 - Under development but very useable (examples)

Programmers/Debuggers

- Hardware to flash and debug
  - (Flash) Similar to AVRISP
  - (Debug) Control state of target
- ST-Link/V2 (~£30)
  - "Official" ST tool
  - Nucleo and Discovery boards have built in ST-Link
- Serial bootloader
  - STM32 only
- Blackmagic Probe (BMP) (~$60)
  - Open source project
  - Talks directly to gdb
  - No openocd or gdb stubs :)
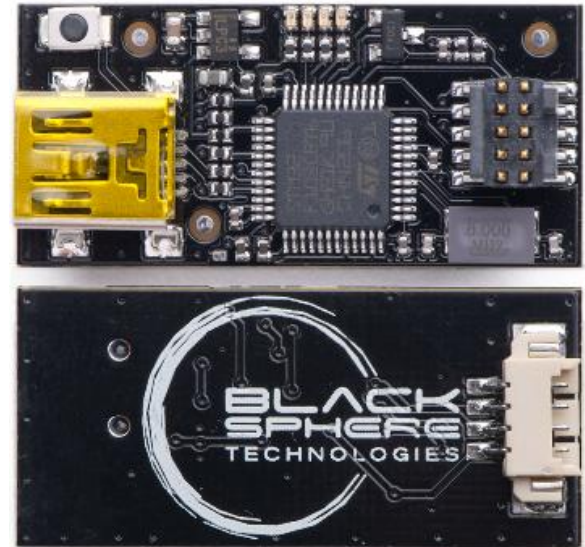
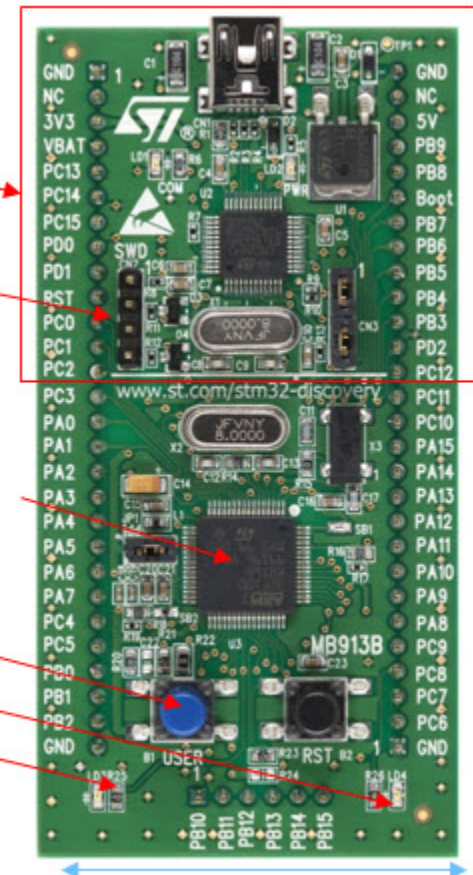# How can I play with one?

Discovery and nucleo (mbed compat) boards
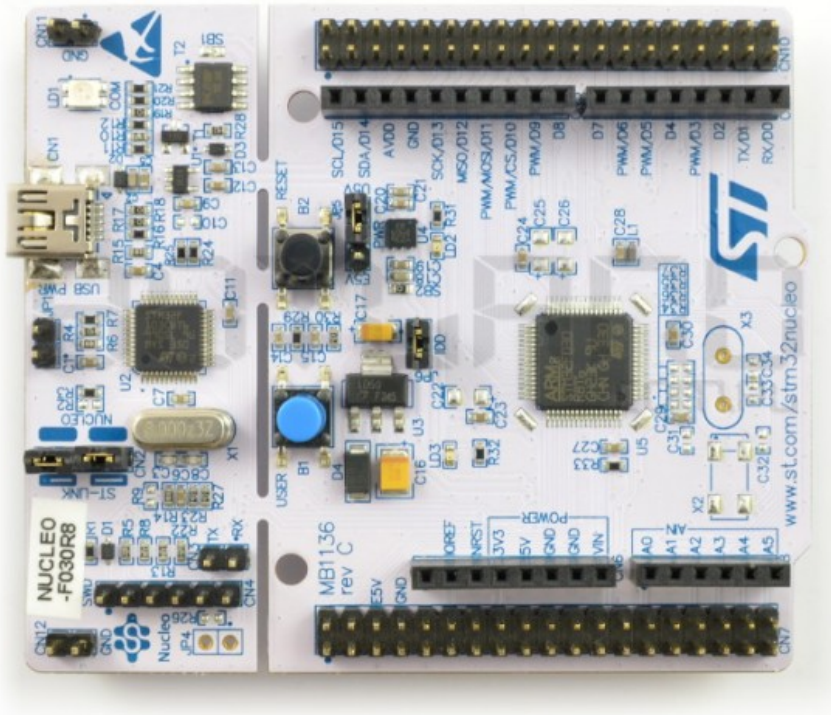
Documentation

- ST Datasheet
  - Device specific (e.g. F051)
  - Anything that is specific to single device in the range
- ST Reference manual
  - Family specific (e.g. F0x1)
  - Everything else!
- Firmware examples
  - ST package examples for the SPL
  - libopencm3 has examples demonstrating use of peripherals via its API
  - github.com/libopencm3/libopencm3-examples