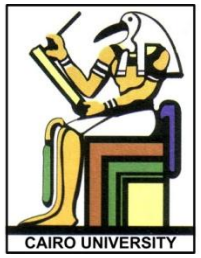


MICROCONTROLLERS AND INTERFACING USING ARM PROCESSOR

Lecture 1

Prof. Yasser Mostafa Kadah

Course Goals

- At the end of this course, you should be able to:
  - Understand the architecture of one of the most popular microcontroller (MCU) families
  - Use an integrated development environment (IDE) to program and debug an MCU
  - Program an MCU using Assembly and C languages
  - Understand and use peripherals integrated into an MCU
  - Interface an MCU to simple external components
  - Use timers in various modes
  - Understand and use interrupts
  - Understand and use analog to digital converters (ADC) and digital to analog converters (DAC)
  - Communicate using a serial interface (if time allows)

Course Prerequisites

- A course in Electric Circuits that includes understanding basic electronic components such as resistors, capacitors, diodes and transistors
- A course in basic digital logic design that includes logic gates and Boolean arithmetic
- Ability to program in a high-level programming language such as C or C++

Microprocessors and Microcontrollers

- Microprocessor: general-purpose CPU
  - Emphasis is on flexibility and performance
  - Generic user-interface such as keyboard, mouse, etc.
  - Used in a PC, PDA, cell phone, etc.

- Microcontroller: microprocessor + memory on a single chip
  - Emphasis is on size and cost reduction
  - The user interface is tailored to the application, such as the buttons on a TV remote control
  - Used in a digital watch, TV remote control, car and many common day-to-day appliances

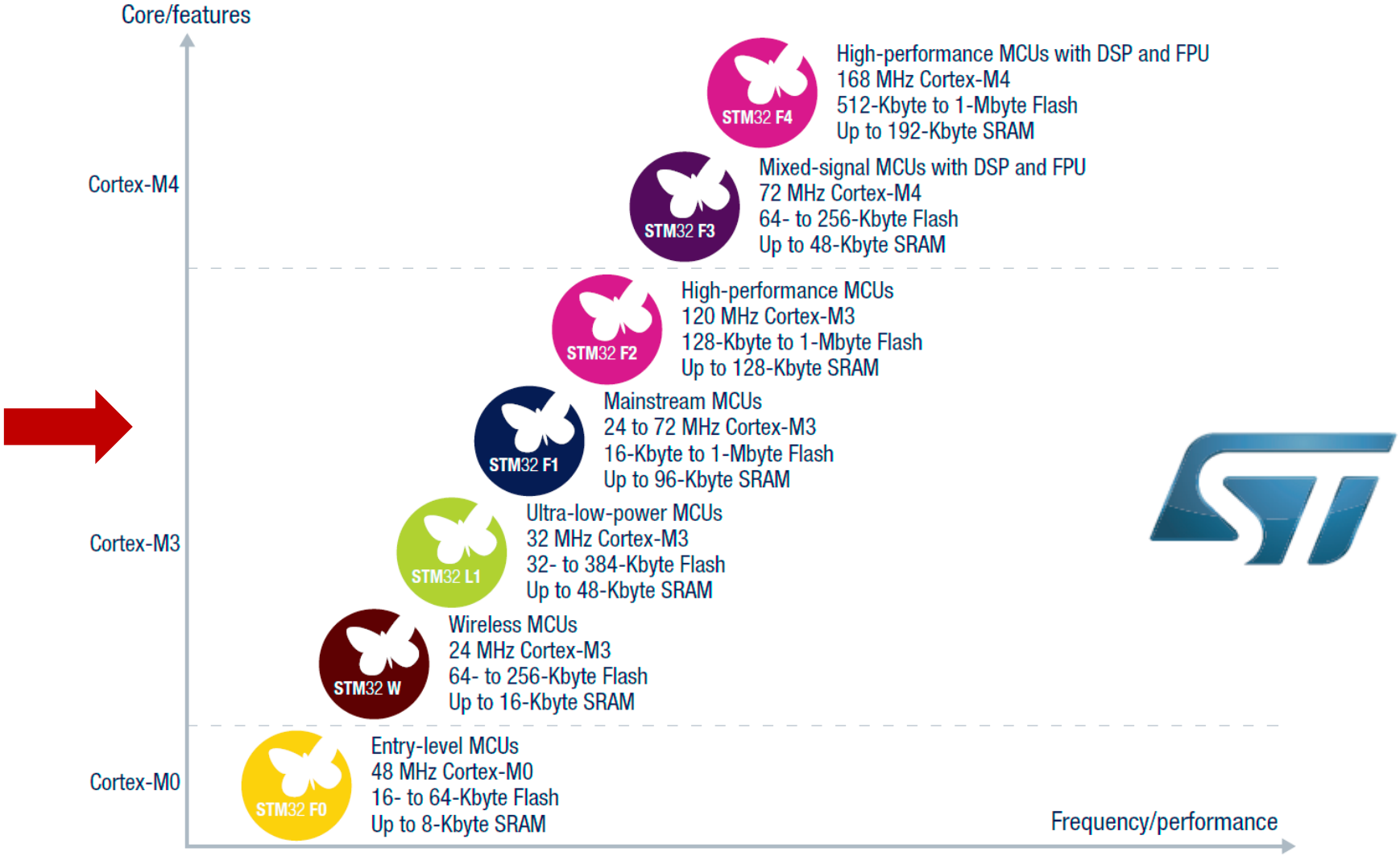
Microcontroller Architectures

- Microcontroller architecture refers to the internal hardware organization of a microcontroller
- Each hardware architecture has its own set of software instructions called assembly language that allows programming of the microcontroller
- Some of the popular microcontroller architectures
  - Intel 8051
  - Zilog Z80
  - Atmel AVR
  - Microchip PIC
  - ARM

ARM Processors

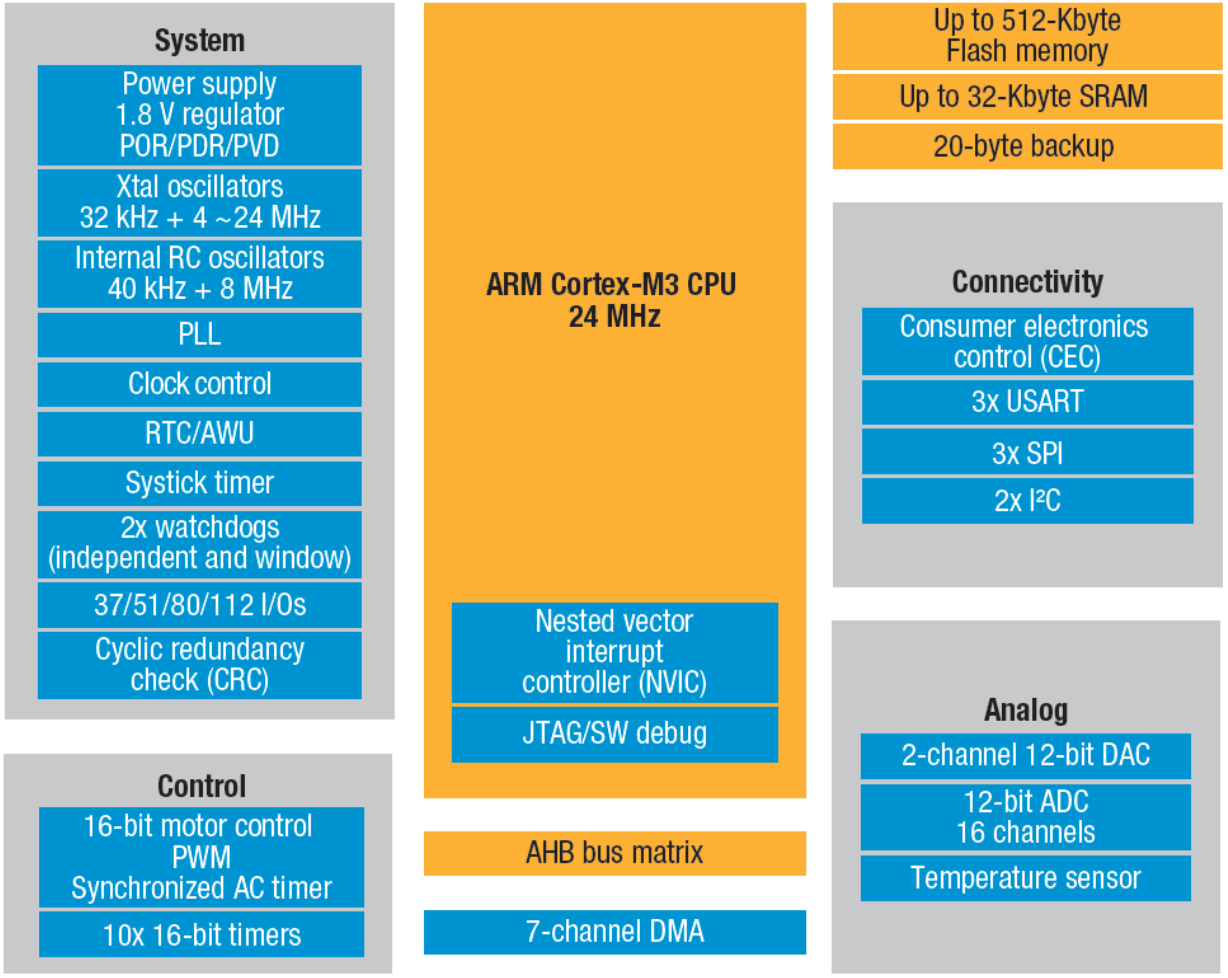
- ARM: Advanced RISC Machines
  - Previously "Acorn RISC Machines"
  - ARM1: 1985
  - Now: ARM11, ARM Cortex A, ARM Cortex R, ARM Cortex M

STM32 Platform



STM32 Product Lines

STM32 Value Line Block Diagram



Abbreviations:

AWU:	Auto wake up from halt	PDR:	Power-down reset	RTC:	Real-time clock
BOR:	Brown-out reset	POR:	Power-on reset	SPI:	Serial peripheral interface
I²C:	Inter integrated circuit	PVD:	Programmable voltage detector	USART:	Universal sync/async receiver transmitter

STM32 Value Line Devices

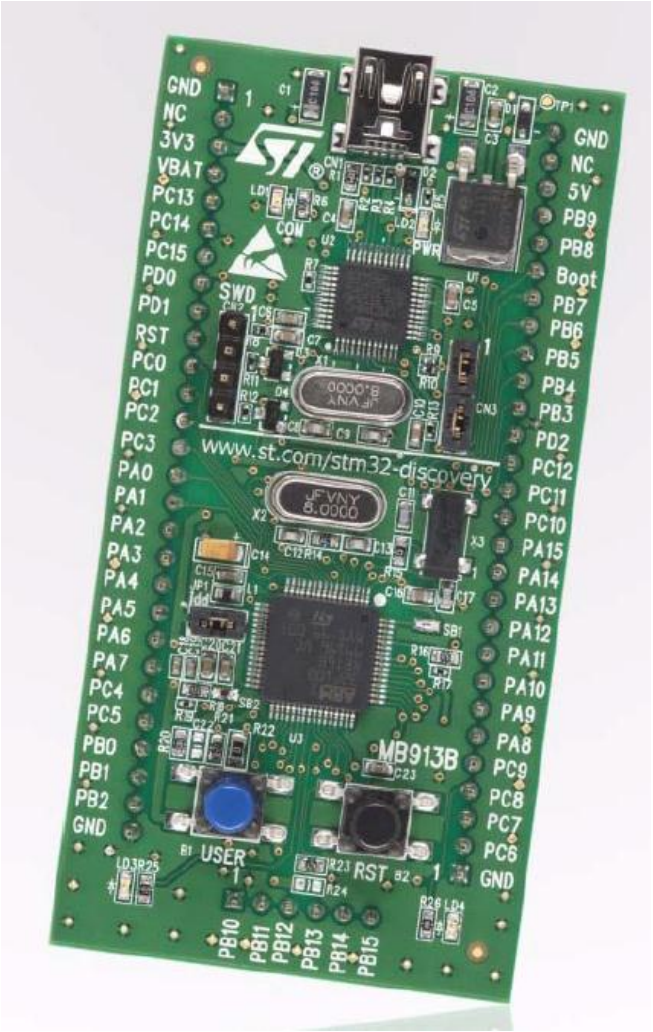
Part number	Package	Flash size (Kbytes)	Internal RAM size (Kbytes)	Timer functions		ADC	DAC	I/Os	Serial interface	Supply voltage (V)
				16-bit	Others					
STM32F100 Value line - 24 MHz CPU										
STM32F100C4	LQFP48 (7x7)	16	4	6x16-bit		10x12-bit	2x12-bit	37	1xSPI,1xI²C, CEC, 2xUSART (IrDa, ISO 7816)	2 to 3.6
STM32F100C6	LQFP48 (7x7)	32	4	6x16-bit		10x12-bit	2x12-bit	37		2 to 3.6
STM32F100C8	LQFP48 (7x7)	64	8	7x16-bit		10x12-bit	2x12-bit	37	2xSPI, 2xI²C, CEC, 3xUSART (IrDa, ISO 7816)	2 to 3.6
STM32F100CB	LQFP48 (7x7)	128	8	7x16-bit		10x12-bit	2x12-bit	37		2 to 3.6
STM32F100R4	LQFP64 (10x10), TFBGA6 (5x5)	16	4	6x16-bit		16x12-bit	2x12-bit	51	1xSPI, 1xI²C, CEC, 2xUSART (IrDa, ISO 7816)	2 to 3.6
STM32F100R6	LQFP64 (10x10), TFBGA64 (5x5), Unsawn wafer V.I. 100%	32	4	6x16-bit		16x12-bit	2x12-bit	51		2 to 3.6
STM32F100R8	LQFP64 (10x10), TFBGA64 (5x5)	64	8	7x16-bit		16x12-bit	2x12-bit	51	2xSPI, 2xI²C, CEC, 3xUSART (IrDa, ISO 7816)	2 to 3.6
STM32F100RB	LQFP64 (10x10), TFBGA64 (5x5)	128	8	7x16-bit	2xWDG, RTC, 24-bit down counter, 2x16-bit basic timers	16x12-bit	2x12-bit	51		2 to 3.6
STM32F100RC	LQFP64 (10x10)	256	24	11x16-bit		16x12-bit	2x12-bit	51	3xSPI, 2xI²C, CEC, 5xUSART (IrDa, ISO 7816)	2 to 3.6
STM32F100RD	LQFP64 (10x10)	384	32	11x16-bit		16x12-bit	2x12-bit	51		2 to 3.6
STM32F100RE	LQFP64 (10x10)	512	32	11x16-bit		16x12-bit	2x12-bit	51		2 to 3.6
STM32F100V8	LQFP100 (14x14)	64	8	7x16-bit		16x12-bit	2x12-bit	80	2xSPI, 2xI²C, CEC, 3xUSART (IrDa, ISO 7816)	2 to 3.6
STM32F100VB	LQFP100 (14x14)	128	8	7x16-bit		16x12-bit	2x12-bit	80		2 to 3.6
STM32F100VC	LQFP100 (14x14)	256	24	11x16-bit		16x12-bit	2x12-bit	80	3xSPI, 2xI²C, CEC, 5xUSART (IrDa, ISO 7816)	2 to 3.6
STM32F100VD	LQFP100 (14x14)	384	32	11x16-bit		16x12-bit	2x12-bit	80		2 to 3.6
STM32F100VE	LQFP100 (14x14)	512	32	11x16-bit		16x12-bit	2x12-bit	80		2 to 3.6
STM32F100ZC	LQFP144 (20x20)	256	24	11x16-bit		16x12-bit	2x12-bit	112		2 to 3.6
STM32F100ZD	LQFP144 (20x20)	384	32	11x16-bit		16x12-bit	2x12-bit	112		2 to 3.6
STM32F100ZE	LQFP144 (20x20)	512	32	11x16-bit		16x12-bit	2x12-bit	112		2 to 3.6

STM32 Value line Discovery Kit

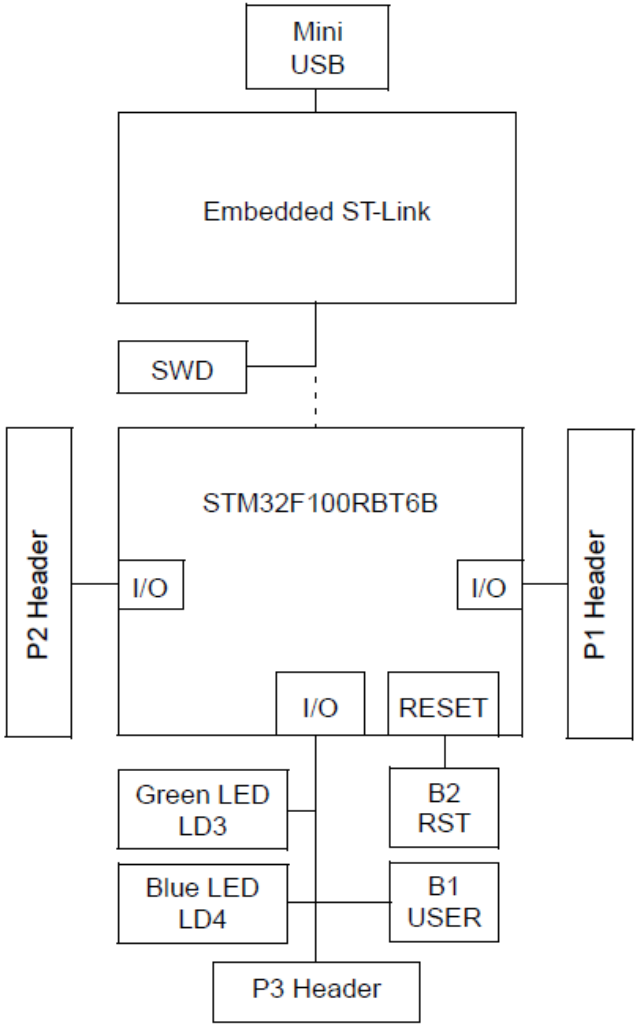
- The cheapest and quickest way to discover the STM32 Value line family
- Embedded ST-LINK or ST-LINK/V2 included to debug applications
- Works with several popular IDEs
  - MDK-ARM (Keil)
  - EWARM (IAR)

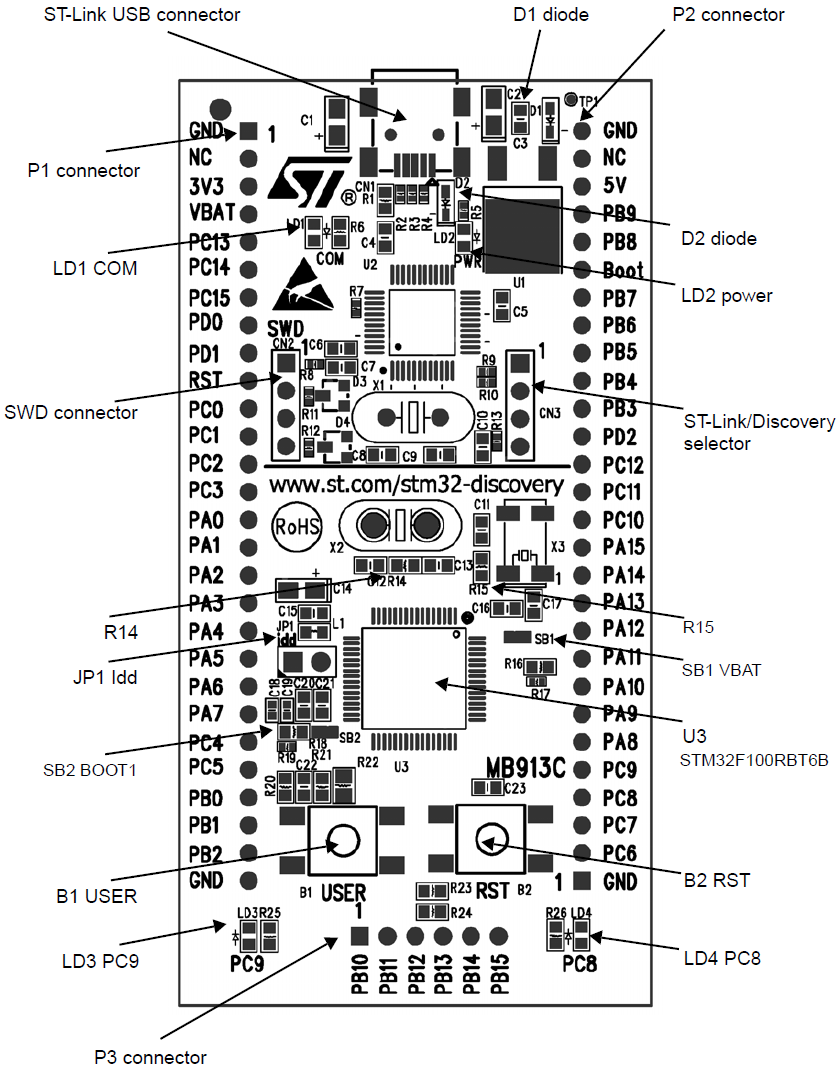
Kit Features

- STM32F100RBT6B microcontroller, 128 KB Flash, 8 KB RAM in 64-pin LQFP
- On-board ST-Link (for programming and debugging)
- Two red LEDs; LD1 for USB communication, LD2 for 3.3 volts power on
- Designed to be powered by USB or an external supply of 5 V or 3.3 V
- Can supply target application with 5 volts and 3 volts
- Two user LEDs, LD3 and LD4 (green and blue)
- Two push buttons (User and Reset)
- Extension header for all QFP64 I/Os for quick connection to prototyping board or easy probing

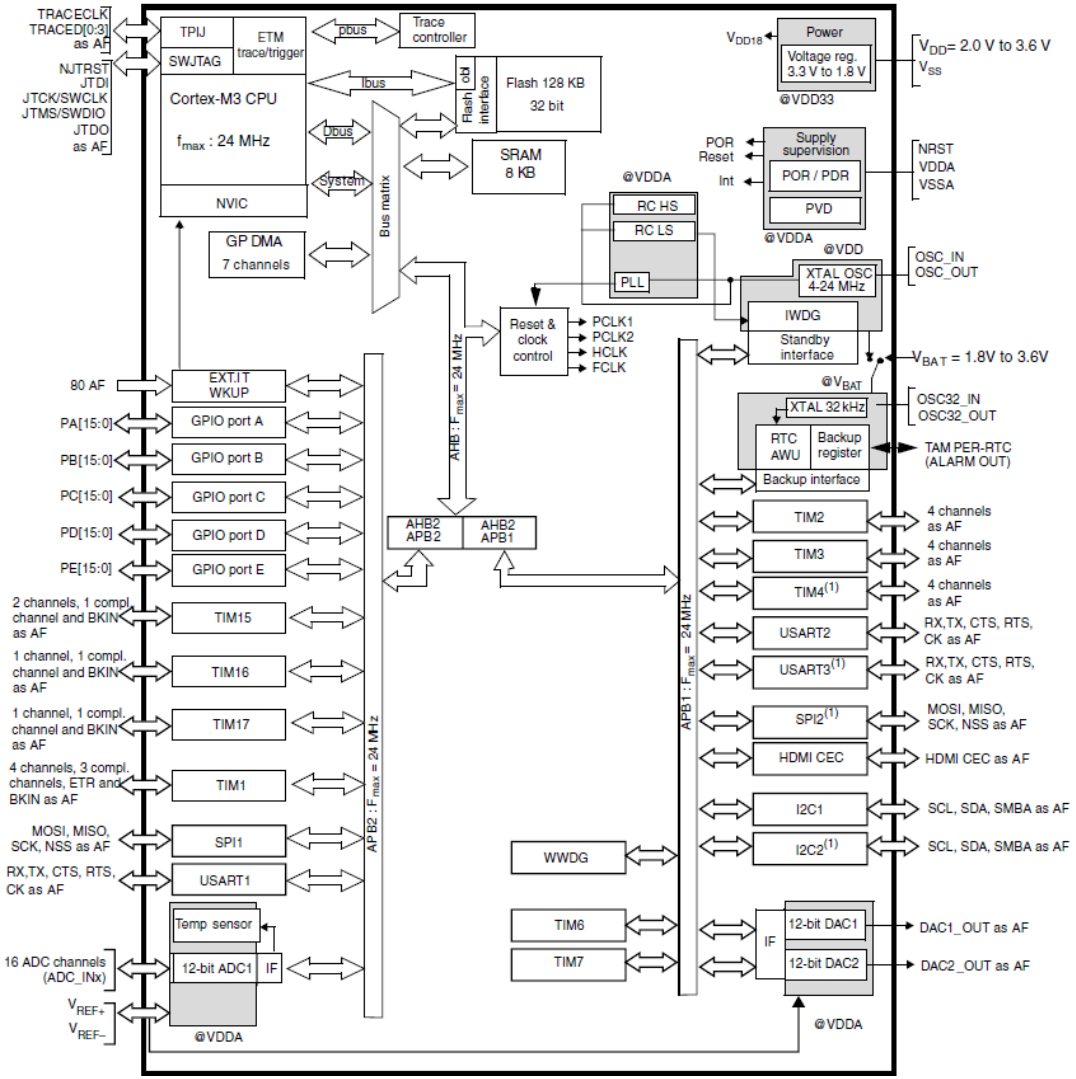


Kit Hardware Block Diagram





STM32F100RBT6B Microcontroller



Assignments

- Explain the differences between MIPS and DMIPS.
- Compare the major ARM processor families available today from the points of view of processing power, cost, applications. (Hint: focus on ARM11 and ARM Cortex series processors)
- Install the MDK-ARM Evaluation Version and compile and link the kit's demo software available from ST web site.
- Revision of C programming.

- Next week: Lab demonstration of kit and development tools