Microprocessor systems

EMISY

Architecture of microcontrollers Lecture 2

Semester 20L – Summer 2020
© Maciej Urbanski, MSc

email: M.Urbanski@elka.pw.edu.pl

WUT

ISE

Important remark

This material is intended to be used by the students during the Microprocessor Systems course for educational purposes only. The course is conducted in the Faculty of Electronics, Warsaw University of Technology.

The use of this material in any other purpose than education is prohibited.

This material has been prepared based on many sources, considered by the author as valueable, however it is possible that the material contains errors and misstatements.

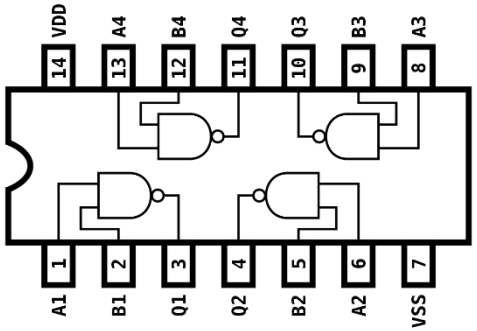
The author takes no responsibility for the usage of this material and any potential losses this usage can lead to. Furthermore the author will be very grateful for pointing out any errors found and also for any other useful remarks on the course material and potential upgrades.

Logic gates – basic types

NAND gate (Not AND)

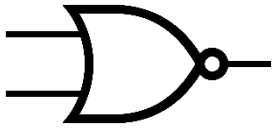
The output is logic zero If both inputs are logic ones, otherwise the output is logic one

A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

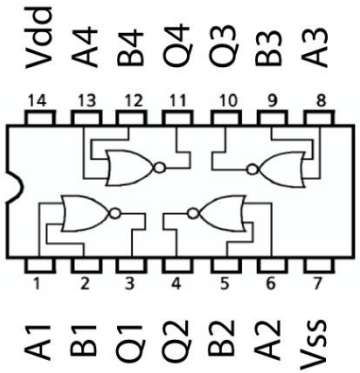


NOR gate (Not OR)

The output is true if both inputs are false, Otherwise the output is false

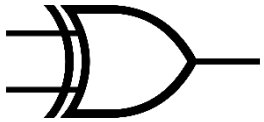


A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0



XOR gate (Exclusive OR)

The output is true if the inputs are not alike, otherwise the output is false



A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

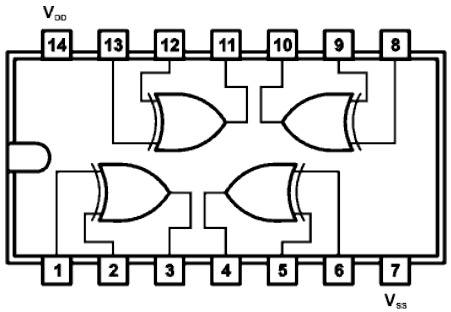
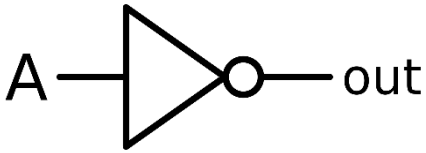


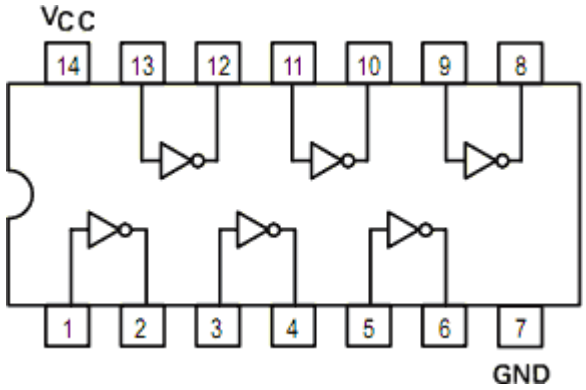
Image source: wikipedia.org

Logic gates – basic types

NOT gate

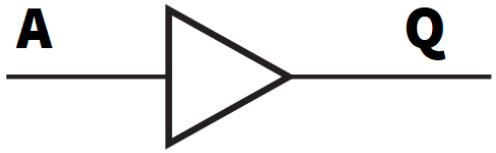
The output is opposite to the input.





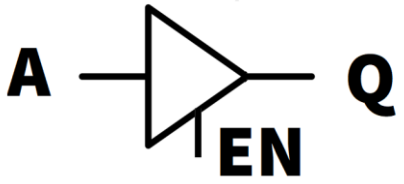
Buffer (Not OR)

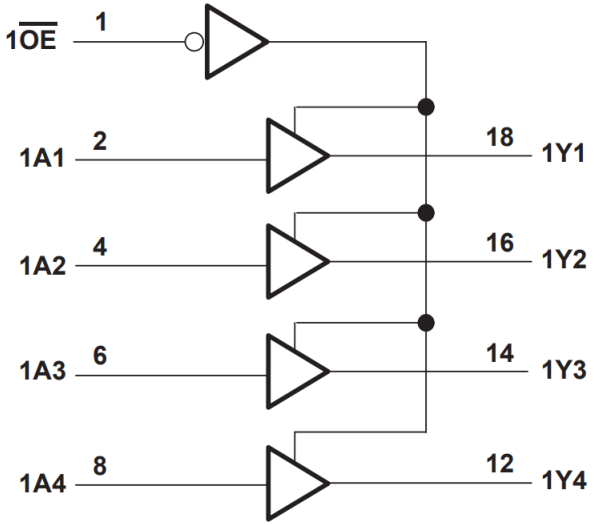
The output is true if both inputs are false,
Otherwise the output is false



Tristate gate (High Z)

If gate is enabled the output is equal to
The input, otherwise the output is
In high impedance (High Z) mode.



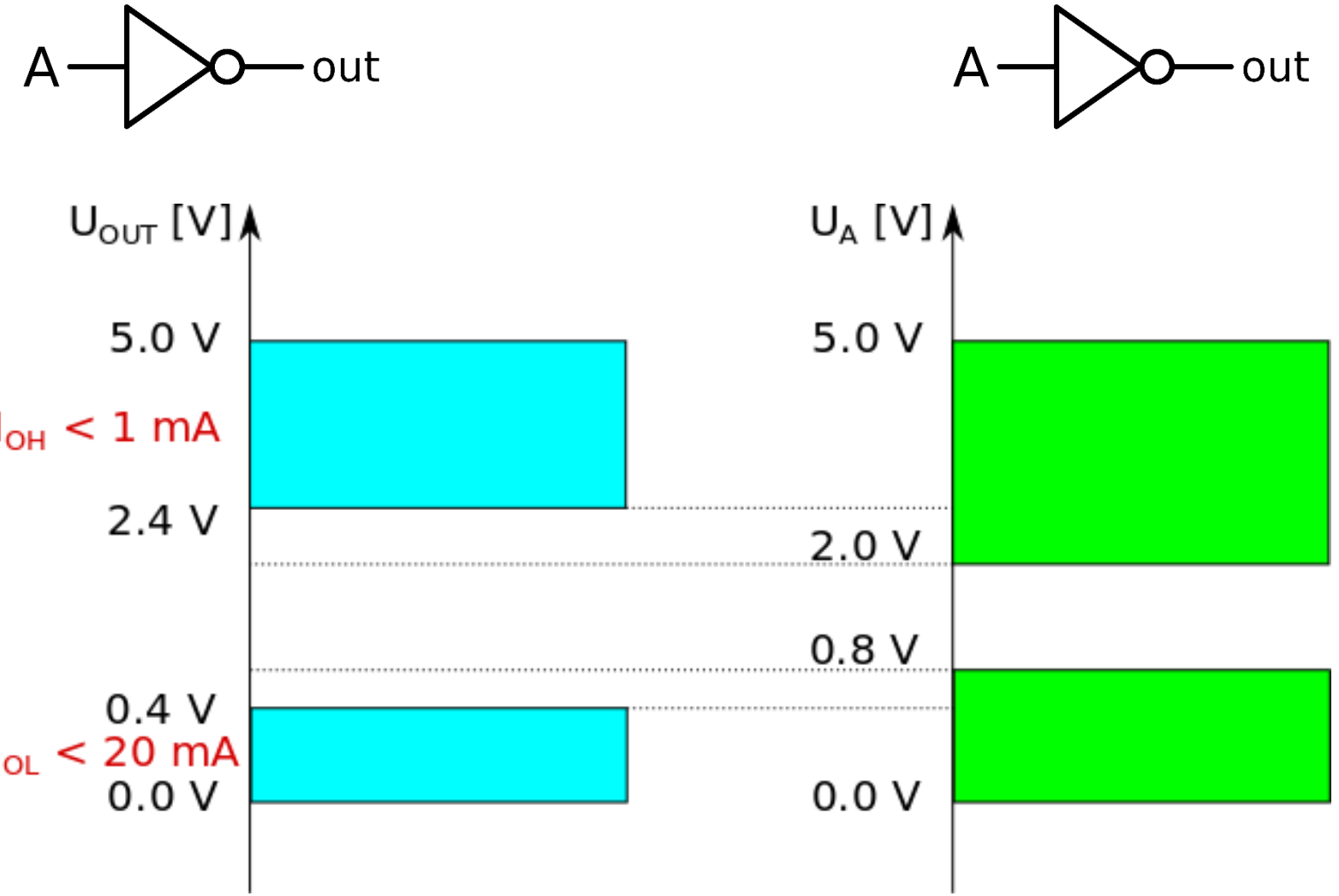


Basic synchronous circuits – flip flops, latches, registers – will be described during Lecture 3.

Image sources: wikipedia.org, ti.com

Logic levels for various types of logic circuits

TTL – Transistor Transistor Logic, 5.0V



These values apply only to TTL chips!

Image source: wikipedia.org

CMOS (AHC) 74AHC04 voltage levels

Symbol	Parameter	Conditions	25 °C			–40 °C to +85 °C		–40 °C to +125 °C		Unit
			Min	Typ	Max	Min	Max	Min	Max	
74AHC04										
V_{IH}	HIGH-level input voltage	V_{CC} = 2.0 V	1.5	-	-	1.5	-	1.5	-	V
		V_{CC} = 3.0 V	2.1	-	-	2.1	-	2.1	-	V
		V_{CC} = 5.5 V	3.85	-	-	3.85	-	3.85	-	V
V_{IL}	LOW-level input voltage	V_{CC} = 2.0 V	-	-	0.5	-	0.5	-	0.5	V
		V_{CC} = 3.0 V	-	-	0.9	-	0.9	-	0.9	V
		V_{CC} = 5.5 V	-	-	1.65	-	1.65	-	1.65	V
V_{OH}	HIGH-level output voltage	$V_I = V_{IH}$ or V_{IL}								
		I_O = –50 μA; V_{CC} = 2.0 V	1.9	2.0	-	1.9	-	1.9	-	V
		I_O = –50 μA; V_{CC} = 3.0 V	2.9	3.0	-	2.9	-	2.9	-	V
		I_O = –50 μA; V_{CC} = 4.5 V	4.4	4.5	-	4.4	-	4.4	-	V
		I_O = –4.0 mA; V_{CC} = 3.0 V	2.58	-	-	2.48	-	2.40	-	V
		I_O = –8.0 mA; V_{CC} = 4.5 V	3.94	-	-	3.80	-	3.70	-	V
V_{OL}	LOW-level output voltage	$V_I = V_{IH}$ or V_{IL}								
		I_O = 50 μA; V_{CC} = 2.0 V	-	0	0.1	-	0.1	-	0.1	V
		I_O = 50 μA; V_{CC} = 3.0 V	-	0	0.1	-	0.1	-	0.1	V
		I_O = 50 μA; V_{CC} = 4.5 V	-	0	0.1	-	0.1	-	0.1	V
		I_O = 4.0 mA; V_{CC} = 3.0 V	-	-	0.36	-	0.44	-	0.55	V
		I_O = 8.0 mA; V_{CC} = 4.5 V	-	-	0.36	-	0.44	-	0.55	V

Image source: wikipedia.org

Logic levels for various types of logic circuits

TTL – Transistor Transistor Logic, 5.0V and 3.3V

Recommended Operating Conditions

Symbol	Parameter	Min	Nom	Max	Units
V_{CC}	Supply Voltage	4.75	5	5.25	V
V_{IH}	HIGH Level Input Voltage	2			V
V_{IL}	LOW Level Input Voltage			0.8	V
I_{OH}	HIGH Level Output Current			–1	mA
I_{OL}	LOW Level Output Current			20	mA
T_A	Free Air Operating Temperature	0		70	°C

Conclusions:

- Care must be taken when driving multiple digital gates (current load, max. number of outputs, buffering)
- Logic levels must be compatible
  - It is not always possible to connect 5.0V and 3.3V logic directly. It may even lead to permanent circuit destruction!!

Image source: wikipedia.org

Basic functional diagram of the microcontroller

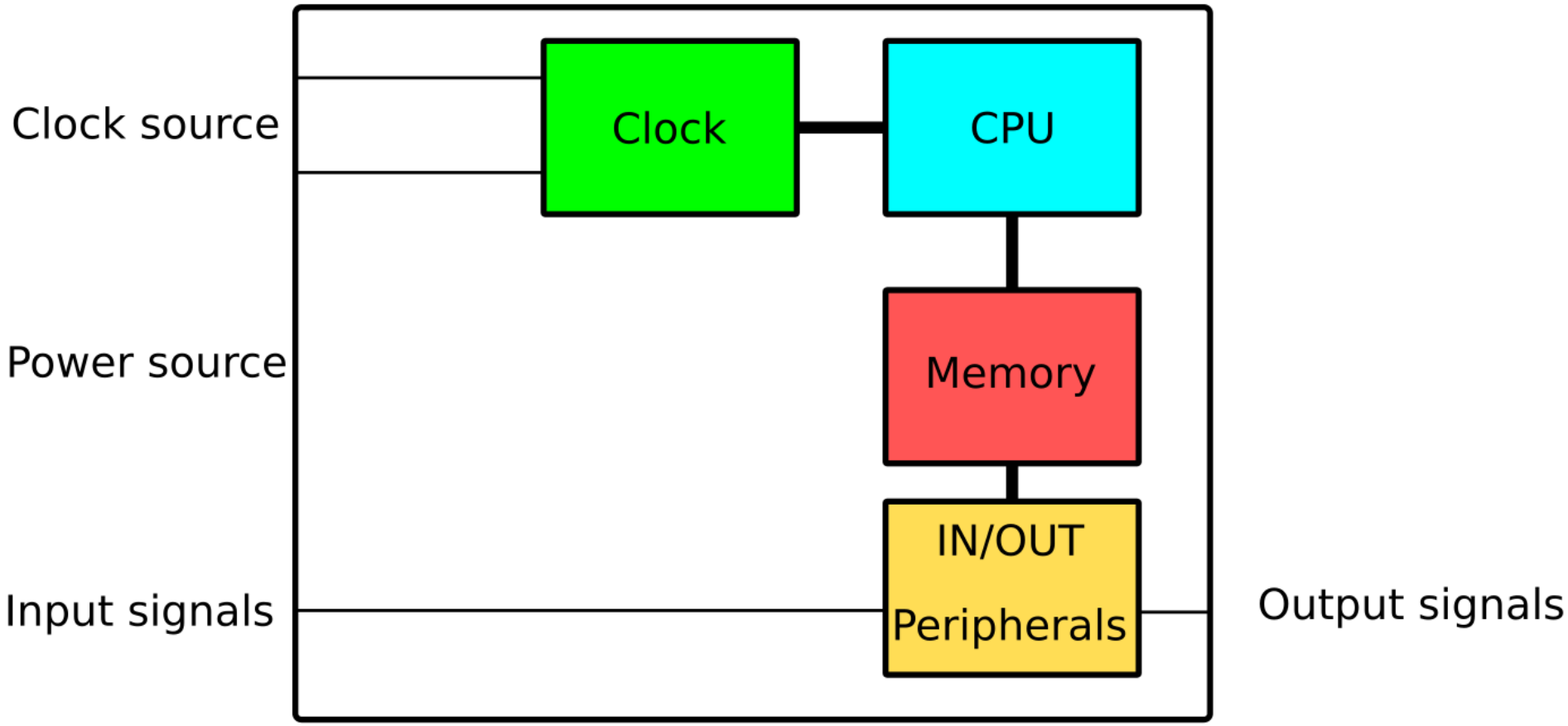
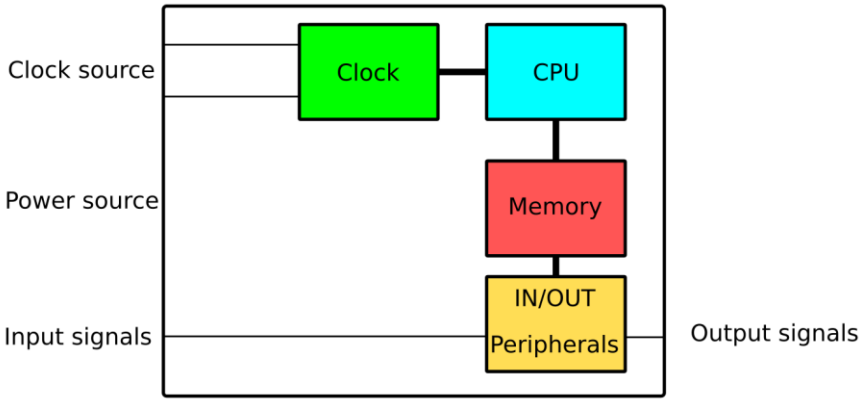
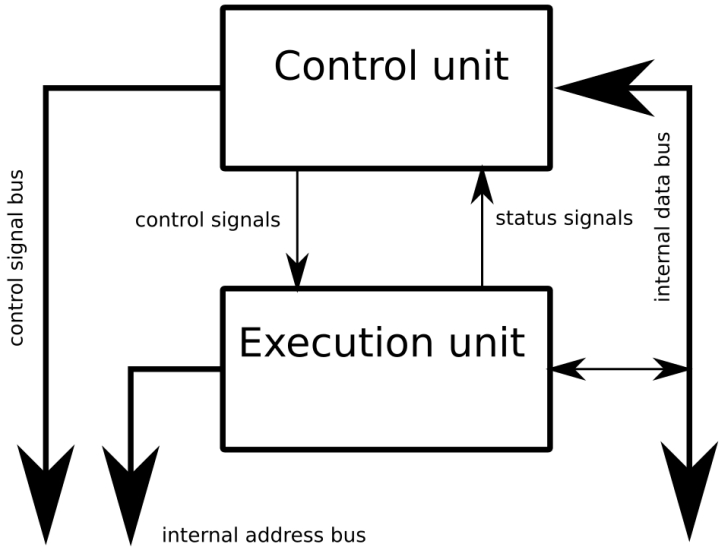


Image source: wikipedia.org

Basic functional diagram of the microcontroller





CPU (Central Processing Unit) performs the set of operations, according to the program stored in the memory.
CPU is synchronous sequential circuit.

CPU consists of two functional blocks:
- Control block
- Execution block

Control unit coordinates functional blocks of microcontroller, by proper routing of data on the buses.

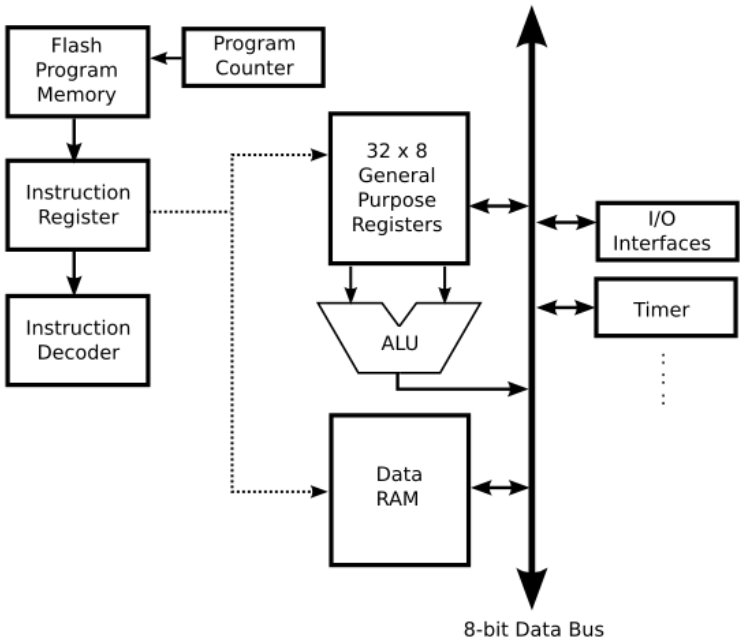
Execution unit performs operations, basing on instructions from the memory and signals from the control unit.

Execution module consists of:
- ALU (arithmetic-logic unit) – performs math and logic functions
- Accumulator
- Registers (more on than in next lecture)

All the modules and blocks are connected via buses:
- Data bus
- Control bus
- Address bus

Image source: wikipedia.org

CPU architecture



Simplified microcontroller Block diagram with single data bus

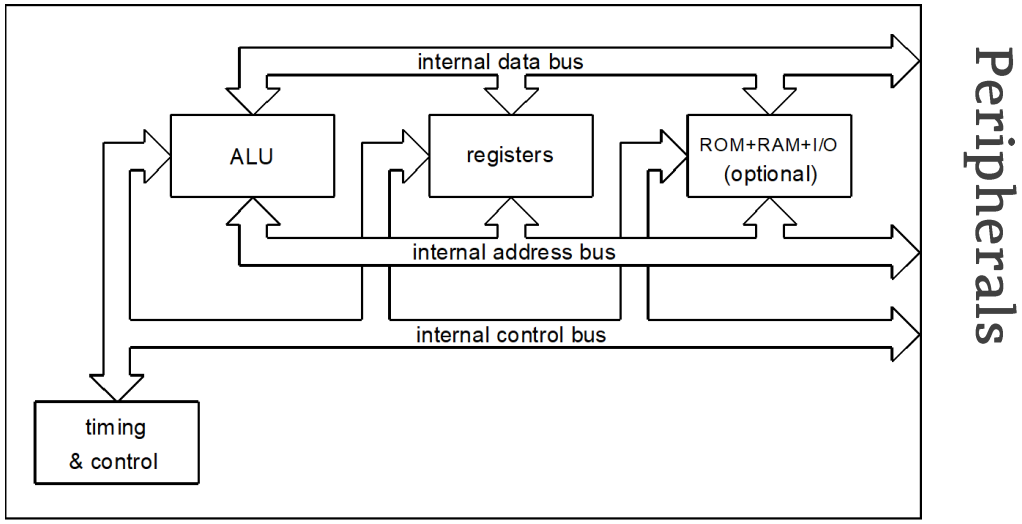
All the modules connected to the bus.

ALU performs mathematical and logical operations, based on arguments stored in the registers. One of the registers is **accumulator**. Operations done by the ALU are limited by the instruction set, given for each type of the microcontroller.

Operation results are stored in the data RAM, temporary results may be stored in general purpose registers.

Image source: https://ccrma.stanford.edu/wiki/Microcontroller_Architecture

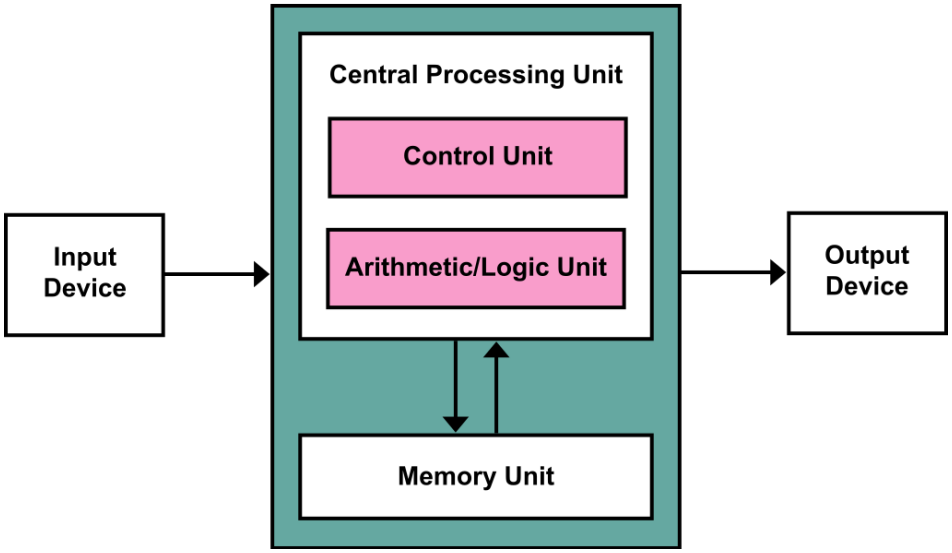
Basic block diagram of three bus architecture of a microcontroller



Data bus	-	it is used to transfer data – operation results, instruction codes. This bus is bi-directional, which means that each device connected can be either transmitter or receiver.
Address bus	-	it transfers addresses in only one direction and is used to transfer addresses of the registers / peripherals that CPU wants to control
Control bus	-	set of control lines, used to drive internal peripherals and mark microcontroller status – it transfers reset signals, interrupt bits, etc.

Image source: wikipedia.org

John von Neumann architecture



- CPU contains ALU and registers
- One bus that is used for both data and instruction transfer
  - Simple, but potential bottleneck
- One memory for data and instructions
  - Stored program concept
  - Linearly addressed memory
  - Memory organization to be decided by the designer
  - Risk of instruction rewrite
- Impossible to perform parallel execution

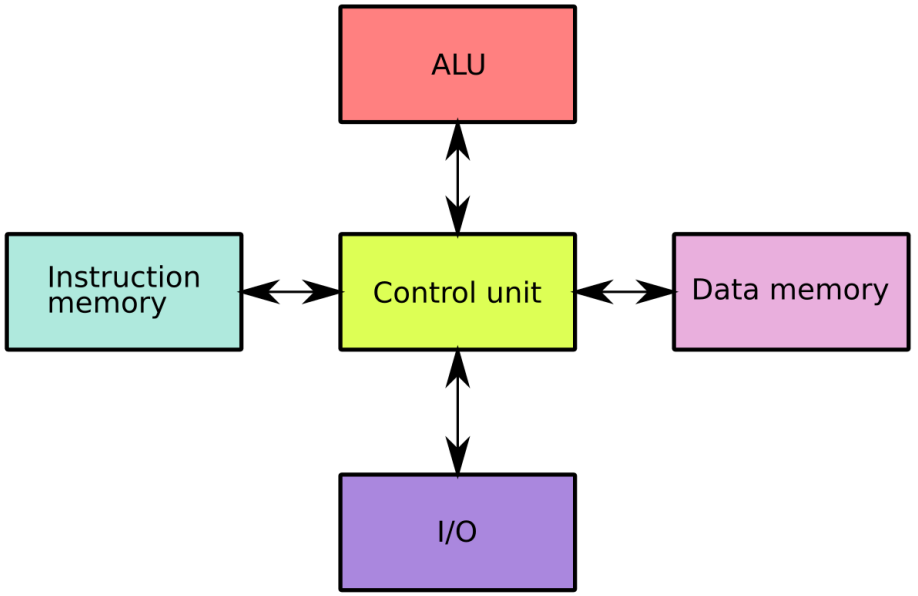
John von Neumann (1903 – 1957)

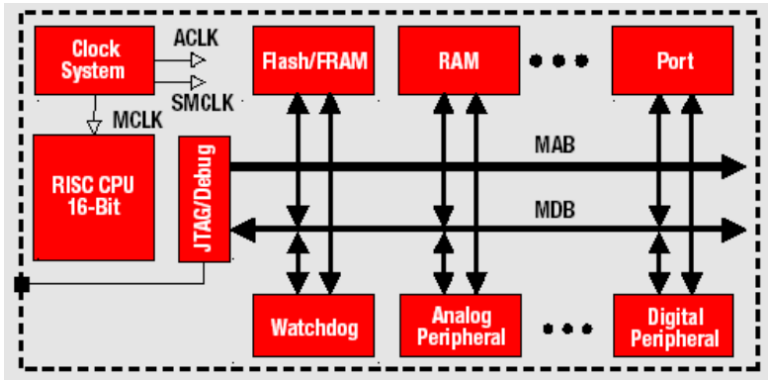
- Hungarian mathematician, engineer, chemist, physicist and IT expert
- Game theory autorh
- Participated in the Manhattan Project
- Helped to improve numerical weather forecasts

Image source: wikipedia.org

Harvard architecture

- Data memory and instruction memory separated
  - Parallel access to instructions and data
- Free data memory cannot be used for instructions
- Free instruction memory cannot be used for data
- It is impossible to accidentally rewrite instructions with data
- Most microcontrollers are based on Harvard architecture





TI MSP430 internal block diagram

Image source: wikipedia.org

Bus organization and how are they done?

Buses are used to transfer information (instructions, data) between microcontroller registers.

Numbers or registers used during these transfers are called register addresses.

A bus is a set of parallel transmission lines (connections), controlled via specialized digital circuitry. They allow to transfer data in parallel mode, between many internal registers.

It is important to emphasize that one transfer can occur only between two selected registers. It is not possible to address many registers during one transfer operation.

Transmitter and receiver (source register and destination register) are connected via tri-state buffers. This allows to select only one active receiving register at a time.

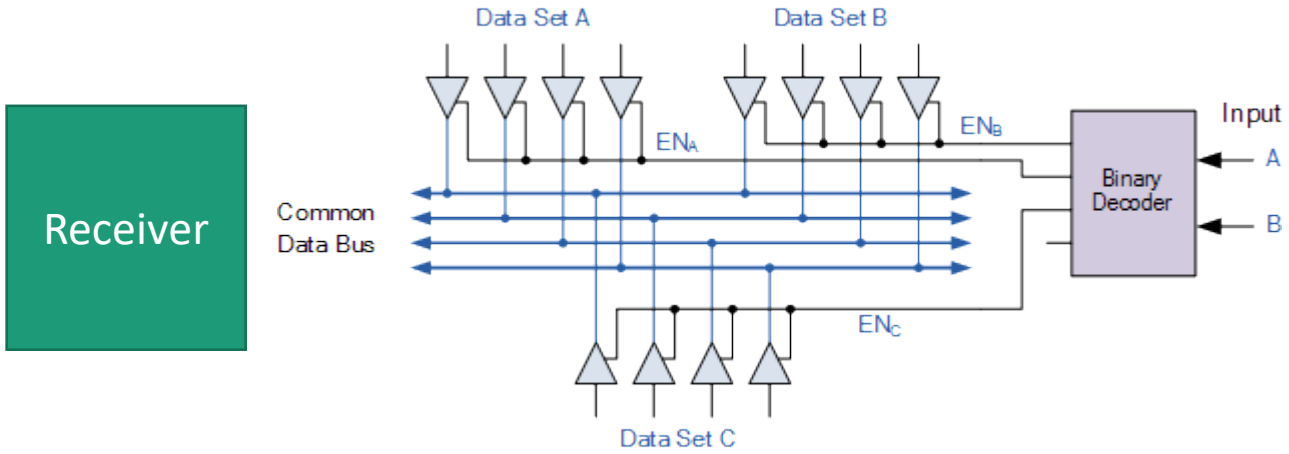
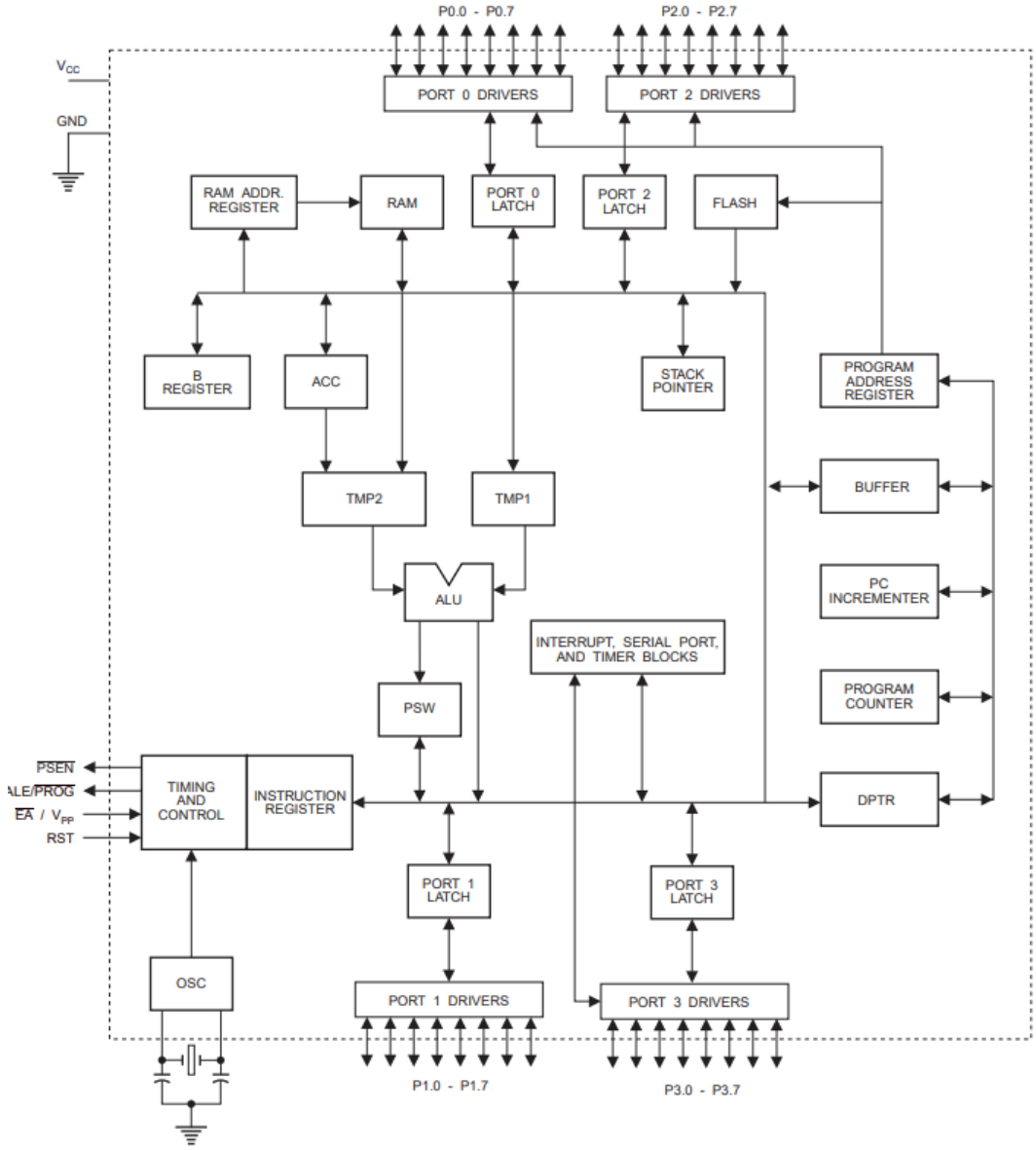
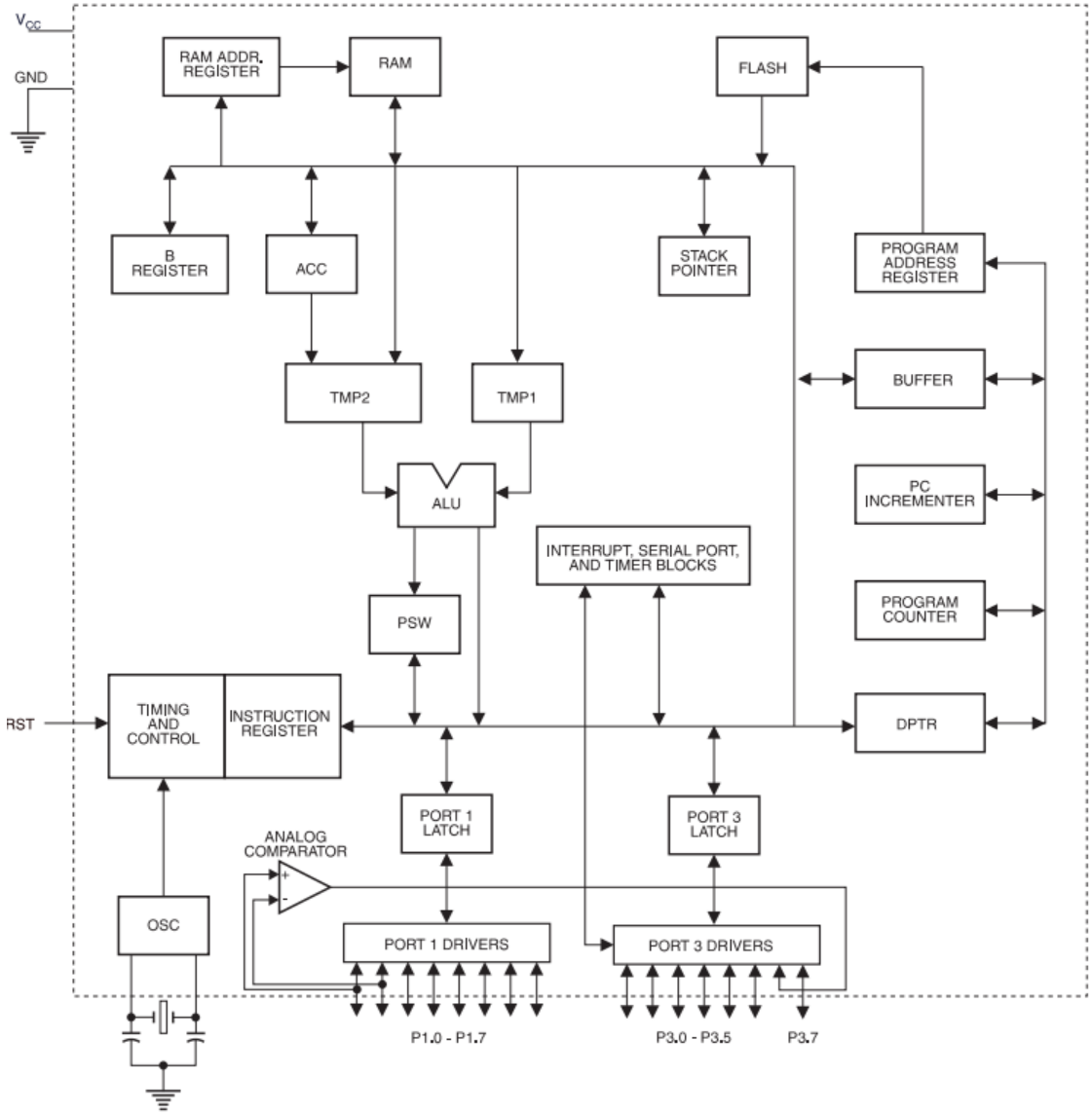


Image source: https://www.electronics-tutorials.ws/logic/logic_9.html

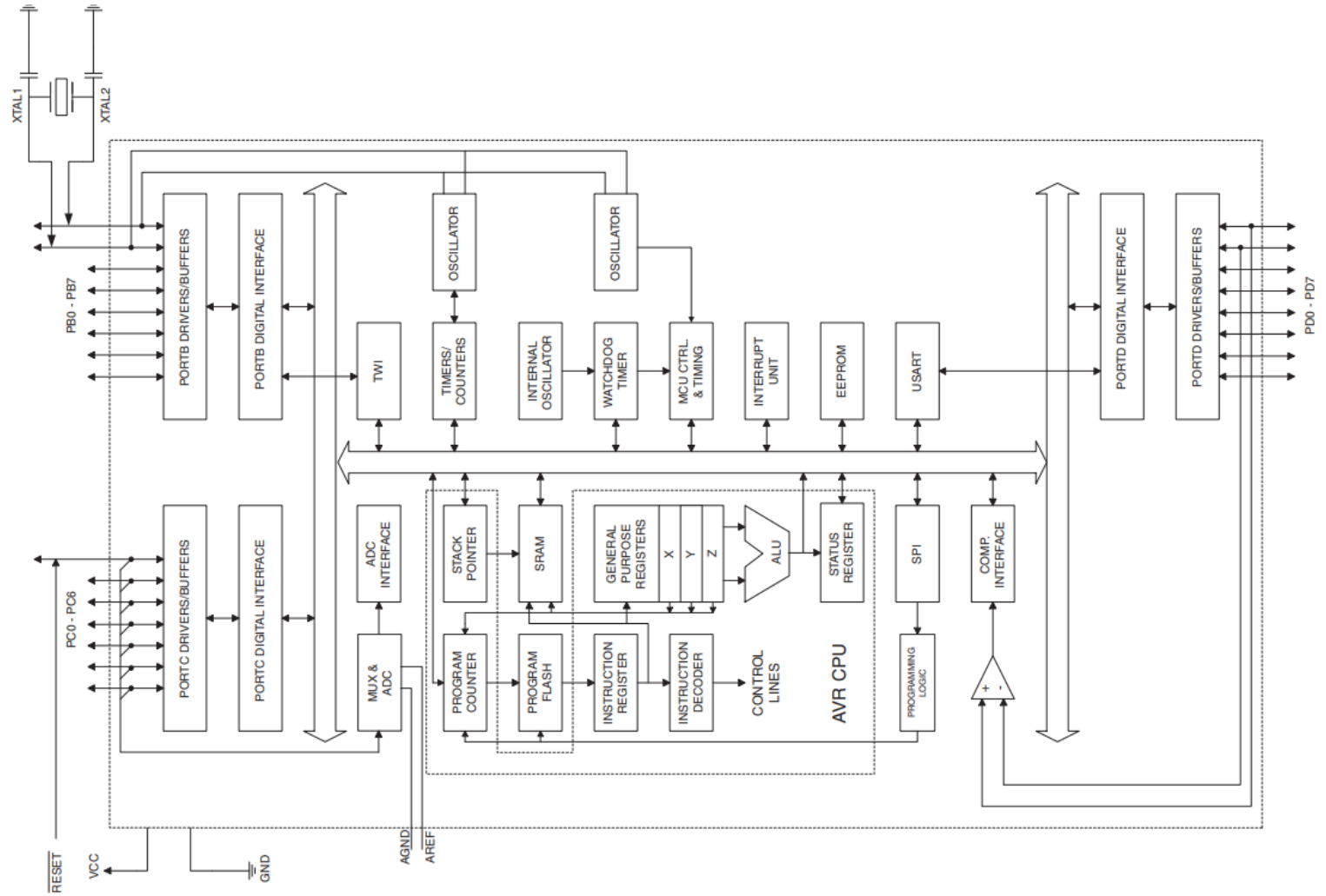
8051 architecture



8051 architecture

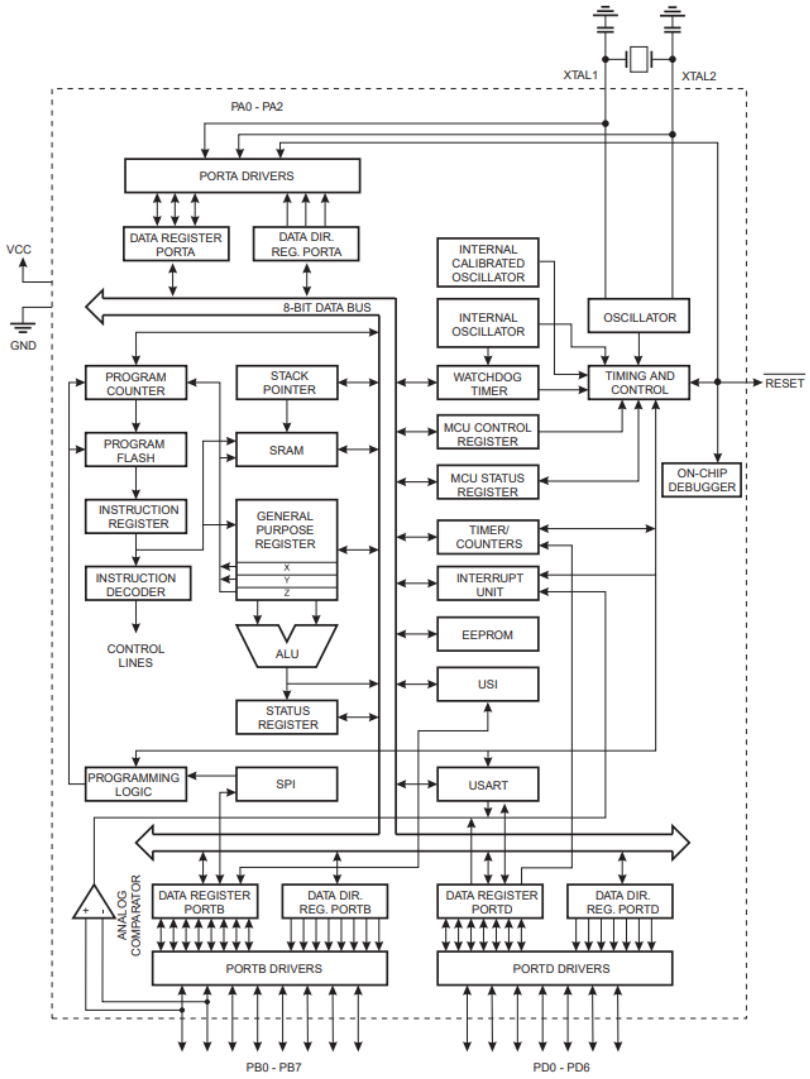


AVR architecture – ATMega 8



AVR architecture – ATTiny 2313

Figure 2. Block Diagram



RISC and CISC architectures

CISC – Complex Instruction Set Computer

- Large instruction list – up to 300 different instructions possible
- Instructions may realize many operations – memory access, data processing
- Many different addressing modes available – will be discussed next week...
- Small set of operational registers
- Instructions may differ in size, argument number, etc.
- Instructions require different number of clock cycles
- Examples of microcontroller : 8051, Motorola 68000, AMD and Intel CPUs
- CISC is hardware friendly – optimized and focused on hardware

RISC – Reduced Instruction Set Computer

- In 1980' it turned out that statistically more than 70% of CISC instructions are not used by the designers
- The RISC architecture was developed
- Reduced set of instructions(instructions) – up to 128
- Unification of instructions – size
- Reduced numer of addressing modes
- Instructions do not realize many operations – separated instructions for memory access and data processing
- Unification of instructions – same execution time for most of the instructions
- Example of microcontroller : AVR, ARM, PIC, SPARC,
- RISC is software friendly – it is more focused on software

STM32 ARM architecture

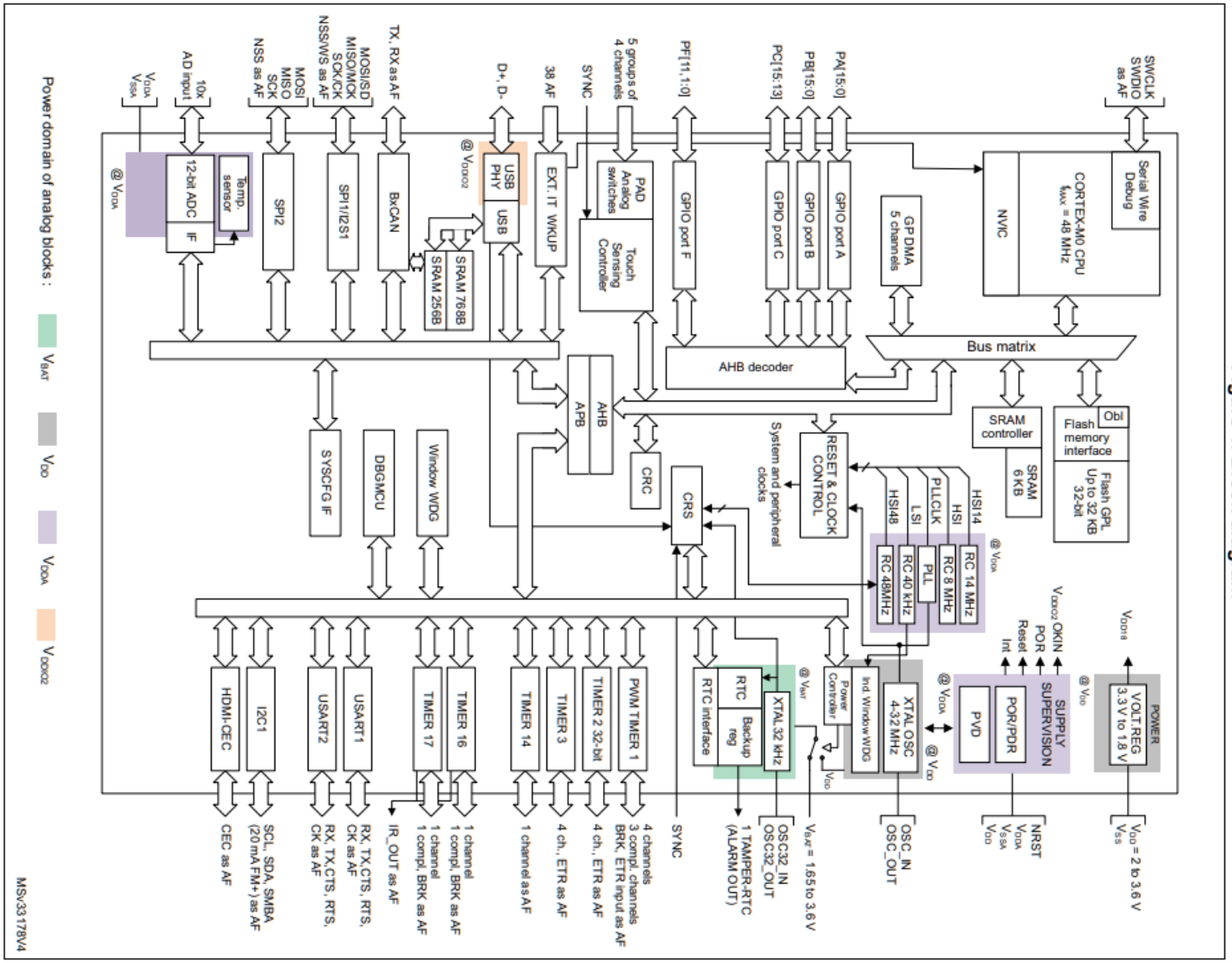


Figure 1. Block diagram

STM32 ARM architecture

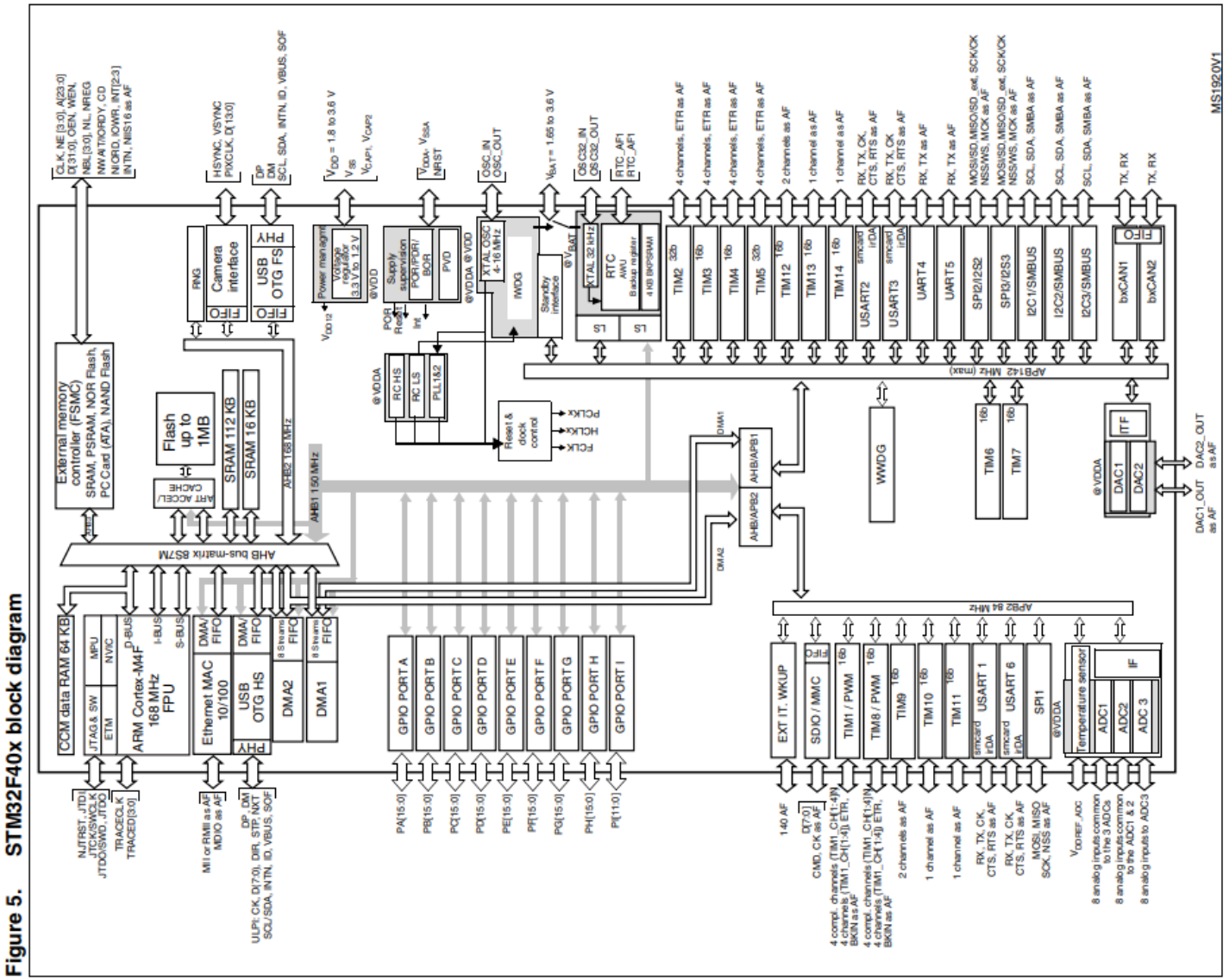


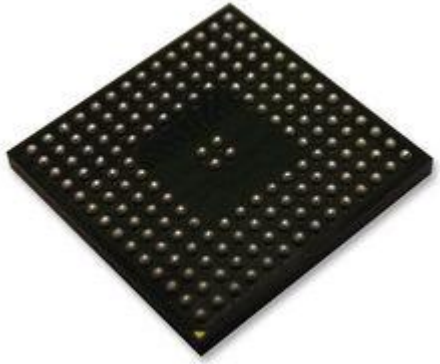
Figure 5. STM32F40x block diagram

1. The timers connected to APB2 are clocked from TIMxCLK up to 168 MHz, while the timers connected to APB1 are clocked from TIMxCLK up to 84 MHz.

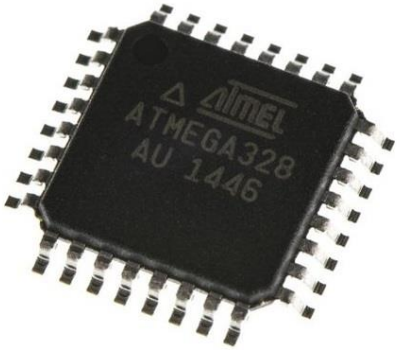
Chip packages, sizes, pinouts

Microcontrollers come in various packages and sizes

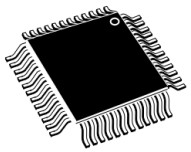
DIP-20 THT package
(ATTiny2313, AT89C2051)



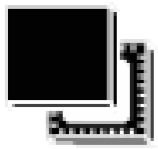
BGA package



LQFP32 package
(ATMega8, STM32)

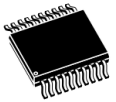


LQFP48 7x7 mm
LQFP32 7x7 mm



UFQFPN48 7x7 mm
UFQFPN32 5x5 mm
UFQFPN28 4x4 mm

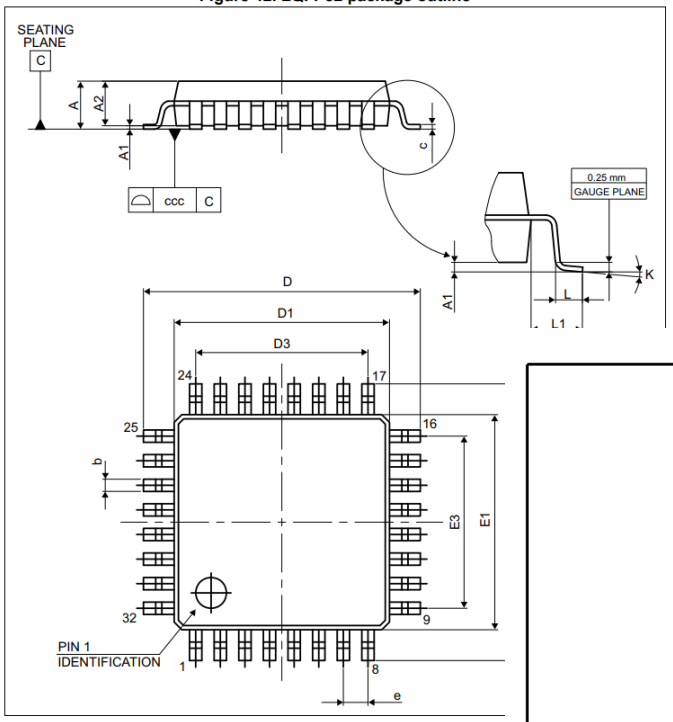
WLCSP36
2.6x2.7 mm



TSSOP20
6.5x4.4 mm

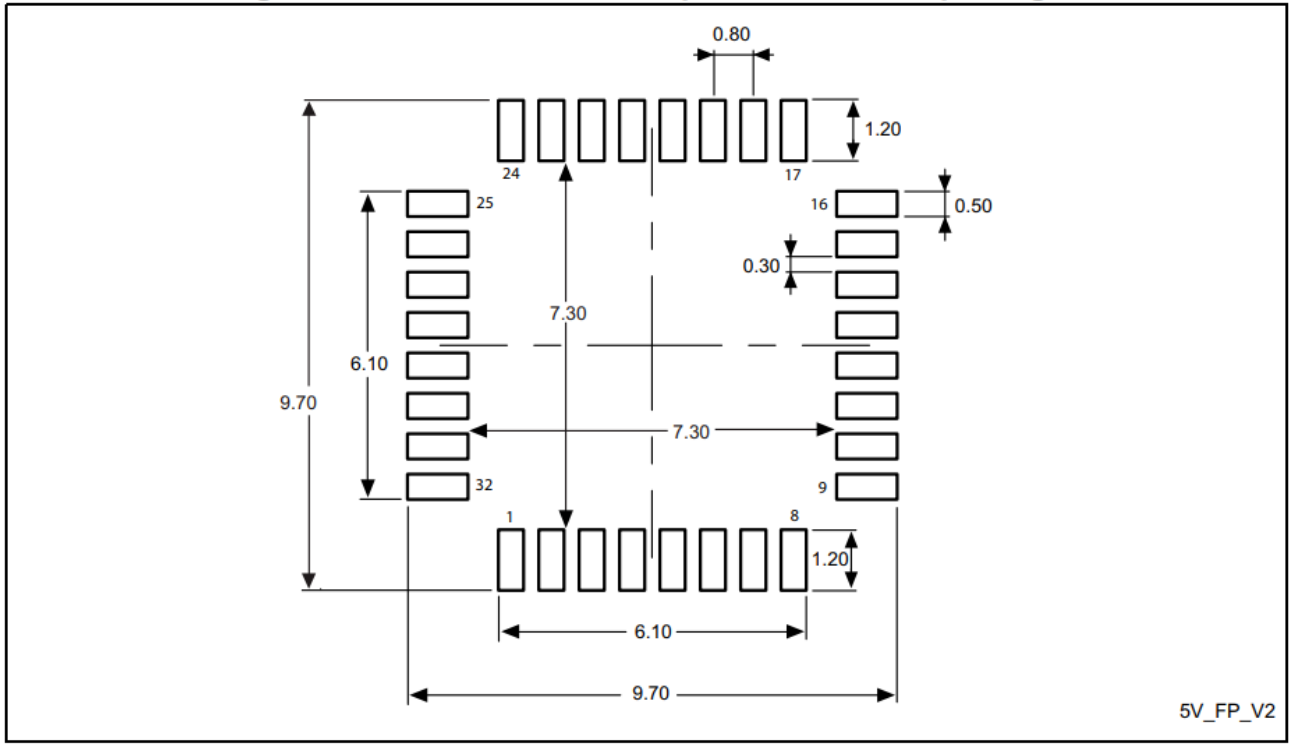
Chip packages, sizes, pinouts

Figure 42. LQFP32 package outline



1. Drawing is not to scale.

Figure 43. Recommended footprint for LQFP32 package



1. Dimensions are expressed in millimeters.

How to select a microcontroller for the application

Guideline based on: https://community.arm.com

1. Make a list of required hardware interfaces
2. Examine the software architecture
3. Select the architecture
4. Identify the memory needs and requirements
5. Look on the market
6. Examine costs and power constraints
7. Check part availability
8. Check the part development kits availability
9. Check the part support, both from manufacturer and the designers society
10. Investigate compilers and tools
11. Start experimenting and prepare prototype
12. If all went well, design the final hardware
13. Go for a ~~beer~~ milkshake ☺

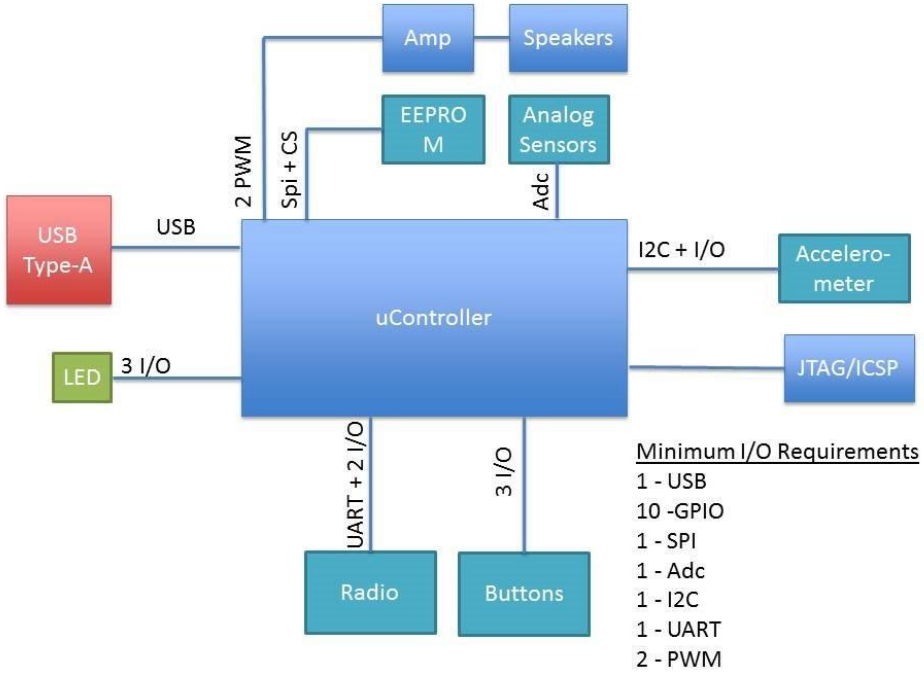


Image source: https://community.arm.com/iot/embedded/b/embedded-blog/posts/10-steps-to-selecting-a-microcontroller

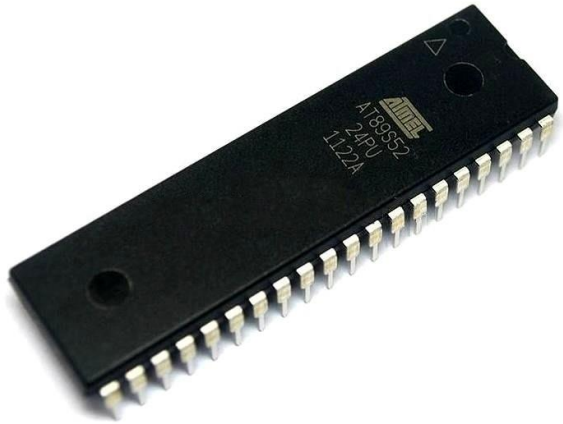
How to select a microcontroller for the ~~application~~ project

Be aware that there are many 8051 microcontroller manufacturers, not only Atmel/Microchip

https://www.silabs.com/mcu/8-bit

DS80C323-MCD+ Datasheet

Digi-Key Part Number	DS80C323-MCD+-ND
Manufacturer	Maxim Integrated
Manufacturer Part Number	DS80C323-MCD+
Description	IC MCU 8BIT ROMLESS 40DIP
Manufacturer Standard Lead Time	11 Weeks
Detailed Description	8051 80C Microcontroller IC 8-Bit 18MHz ROMless 40-PDIP



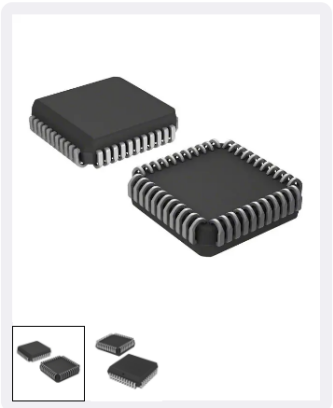
SILICON LABS

About ▾ Products ▾ Solutions ▾ Community & Support ▾ Search silabs.com

Silicon Labs » MCU » 8-bit » EFM8 Sleepy Bee

EFM8™ Sleepy Bee 8-bit Microcontrollers (MCUs)

Optimal for low-power and battery-powered systems, Sleepy Bee EFM8™ microcontrollers (MCUs) offer the lowest power consumption available by combining innovative low-energy techniques and short wake-up times. The Sleepy Bee family of MCUs operate at up to 25 MHz and include up to 14 robust, high-sensitivity capacitive sense channels.



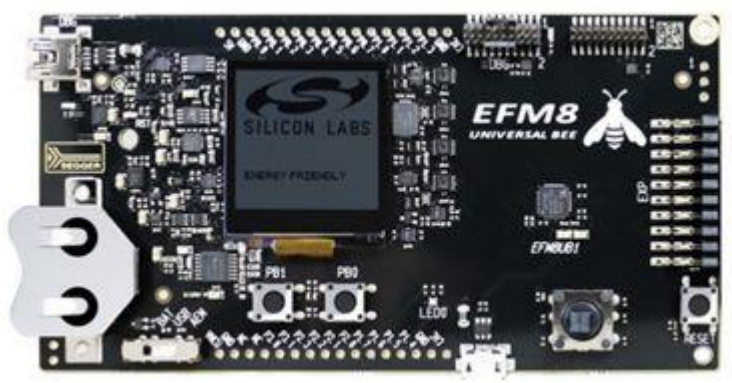
P87C51MC2BA/02,529 Datasheet

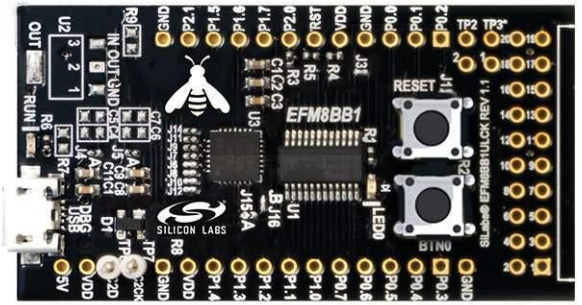
Digi-Key Part Number	568-1244-5-ND
Manufacturer	NXP USA Inc.
Manufacturer Part Number	P87C51MC2BA/02,529
Description	IC MCU 8BIT 96KB OTP 44PLCC
Manufacturer Standard Lead Time	12 Weeks
Detailed Description	8051 87C Microcontroller IC 8-Bit 24MHz 96KB (96K x 8) OTP 44-PLCC (16.59x16.59)

Image source: https://community.arm.com/iot/embedded/b/embedded-blog/posts/10-steps-to-selecting-a-microcontroller

How to select a microcontroller for the ~~application~~ project

Be aware that there are many 8051 microcontroller manufacturers, not only Atmel/Microchip





RENESAS

NORDIC SEMICONDUCTOR

Image source: https://community.arm.com/iot/embedded/b/embedded-blog/posts/10-steps-to-selecting-a-microcontroller

Where to look for?

Octopart

MOUSER ELECTRONICS

Digi-Key ELECTRONICS

Farnell AN AVNET COMPANY

botland

KAMAMI

TME Electronic Components

Image source: https://community.arm.com/iot/embedded/b/embedded-blog/posts/10-steps-to-selecting-a-microcontroller