Microprocessor
or
Microcontroller?

A little History

- *What is a computer?*
  - **[**Merriam-Webster Dictionary] one that computes; *specifically :* programmable electronic device that can store, retrieve, and process data.
  - [Wikipedia] A computer is a machine that manipulates data according to a list of instructions.
- Classification of Computers (power and price)
  - Personal computers
  - Mainframes
  - Supercomputers
  - Dedicated controllers – Embedded controllers

Mainframes

IBM 9000

- Massive amounts of memory
- Use large data words…64 bits or greater
- Mostly used for military defense and large business data processing
- Examples: IBM 4381, Honeywell DPS8

Personal Computers

- Any general-purpose computer
  - intended to be operated
  - directly by an end user
- Range from small microcomputers that work with 4-bit words to PCs working with 32-bit words or more
- They contain a **Processor** - called different names
  - Microprocessor – built using Very-Large-Scale Integration technology; the entire circuit is on a single chip
  - Central Processing Unit (CPU)
  - Microprocessor Unit (MPU) – similar to CPU

http://en.wikipedia.org/wiki/Personal_computer

Supercomputers

- Fastest and most powerful mainframes
  - Contain multiple central processors (CPU)
  - Used for scientific applications, and number crunching
  - Now have petaflops performance
    - *FLoating Point Operations Per Second* (FLOPS)
    - Used to measure the speed f the computer
- Examples of special-purpose supercomputers:
  - Belle, Deep Blue, and Hydra, for playing chess
  - Reconfigurable computing machines or parts of machines
  - GRAPE, for astrophysics and molecular dynamics
  - Deep Crack, for breaking the DES cipher
  - MDGRAPE-3, for protein structure computation

http://en.wikipedia.org/wiki/Supercomputer

Microcontrollers – Embedded Systems
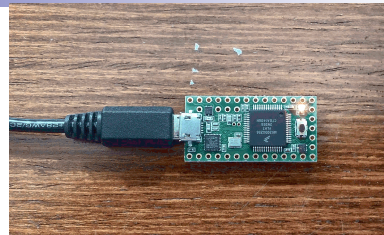
- An embedded system is a special-purpose computer system designed to perform one or a few dedicated functions often with real-time
- An integrated device which consists of multiple devices
  - Microprocessor (MPU)
  - Memory
  - I/O (Input/Output) ports
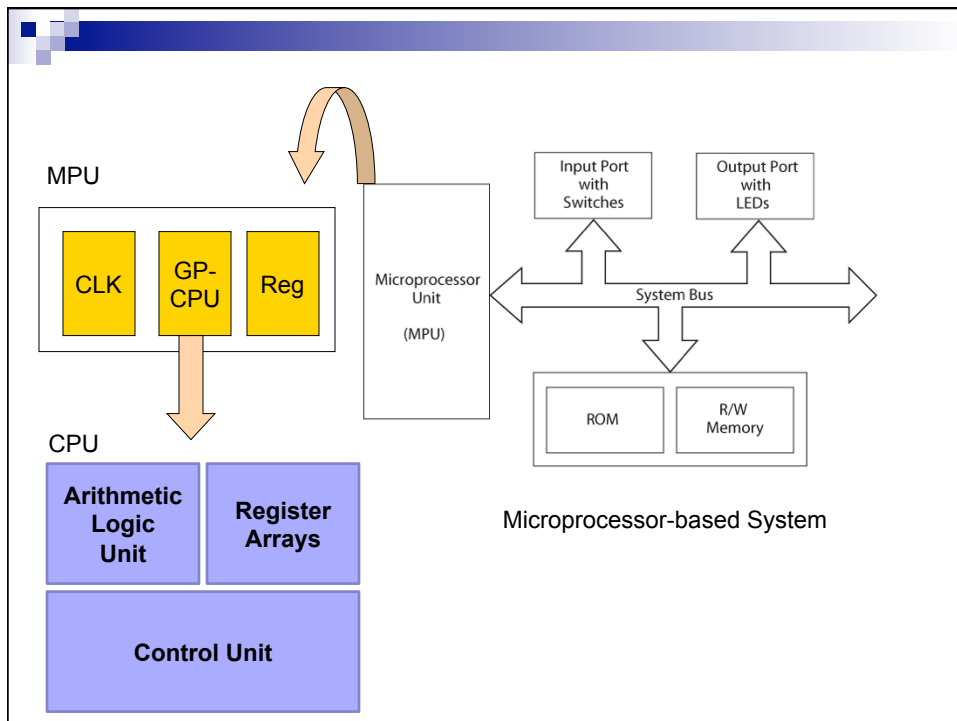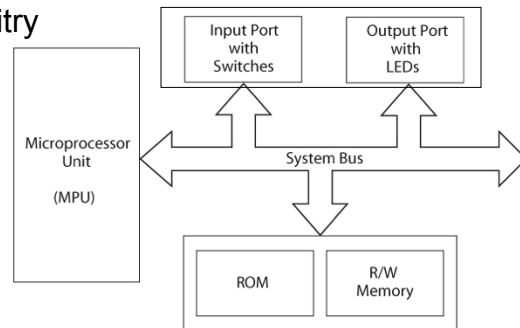- Often has its own dedicated software

A little about Microprocessor-based Systems ……

Evolution

- First came transistors
- Integrated circuits
  - SSI (Small-Scale Integration) to ULSI
  - Very Large Scale Integration circuits (VLSI)
- 1- Microprocessors (MPU)
  - Microcomputers (with CPU being a microprocessor)
  - Components: Memory, CPU, Peripherals (I/O)
  - Example: Personal computers
- 2- Microcontroller (MCU)
  - Microcomputers (with CPU being a microprocessor)
  - Many special function peripheral are integrated on a single circuit
  - Types: General Purpose or Embedded System (with special functionalities)

Microprocessor-Based Systems

- Central Processing Unit (CPU)
- Memory
- Input/Output (I/O) circuitry
- Buses
  - Address bus
  - Data bus
  - Control bus

Microprocessor-based System

Microprocessor-Based System with Buses: Address, Data, and Control

Address Lines — Address Bus

Microprocessor Unit (MPU)

ROM | R/W Memory | Input Port | Switches | Output Port | LEDs

Data Lines — Data Bus

RD → READ
WR → WRITE
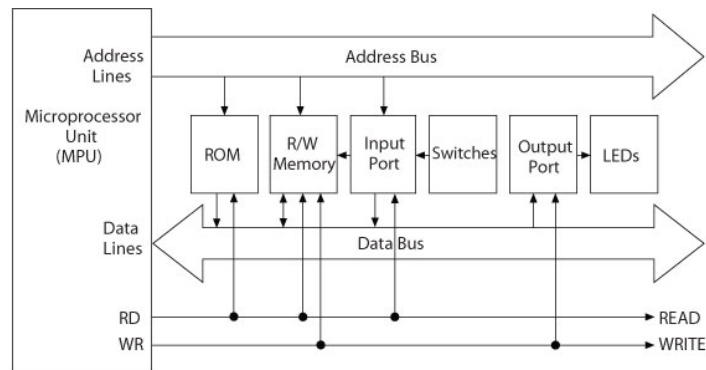
Microprocessor-based Systems
Microprocessor

■ The microprocessor (MPU) is a computing and logic device that executes binary instructions in a sequence stored in memory.

■ Characteristics:
  □ General purpose central processor unit (CPU)
  □ Binary
  □ Register-based
  □ Clock-driven
  □ Programmable

Evolution of CPUs

Transistors

- Vacuum Tubes: A devise to control modify, and amplify electric signals
- Then can transistors
  - Designed by John Bardeen, Will Shockley, and Walter Brattain, scientists at the Bell Telephone Laboratories in Murray Hill, New Jersey - 1947
- In 1960 Jack Kilby and Robert Noyce designed the first integrated circuit (IC)
- Fairchild company manufactured logic gates
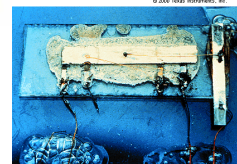
From Computer Desktop Encyclopedia
Reproduced with permission.
© 2000 Texas Instruments, Inc.

Integrated Circuits

- Advances in manufacturing allowed packing more transistors on a single chip
- Transistors and Integrated Circuits from SSI (Small-Scale Integration) to ULSI
- Birth of a microprocessor and its revolutionary impact
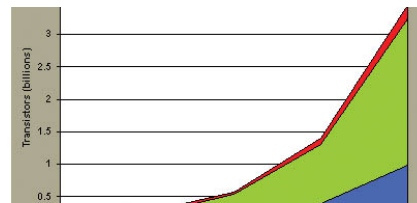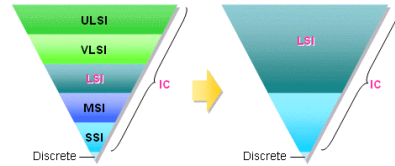
Microprocessors

- Noyce and Gordon Moore started Intel
- Intel designed he first calculator
- Intel designed the first programmable calculator
- Intel designed the first microprocessor in 1971
  - Model 4004
  - 4-bit; 2300 transistors, 640 bytes of memory, 108 KHz clock speed
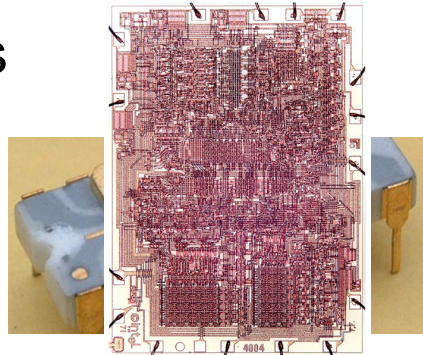
First Processors

- Intel released the 8086, a 16-bit microprocessor, in 1978
- Motorola followed with the MC68000 as their 16-bit processor
  - The 16-bit processor works with 16 bit words, rather than 8 bit words
  - Instructions are executed faster
  - Provide single instructions for more complex instructions such as multiply and divide
- 16 bit processors evolved into 32 bit processors
- Intel released the 80386
- Motorola released the MC68020

Evolution of CPUs

Transistor capacity doubles every 18 months   transistors

MOORE'S LAW

Pentium® 4 Processor — 100,000,000
Pentium® III Processor
Pentium® II Processor — 10,000,000
Pentium® Processor
486™ DX Processor — 1,000,000
386™ Processor
286 — 100,000
8086
8080 — 10,000
8008
4004 — 1000

1970 1975 1980 1985 1990 1995 2000   Source: Intel
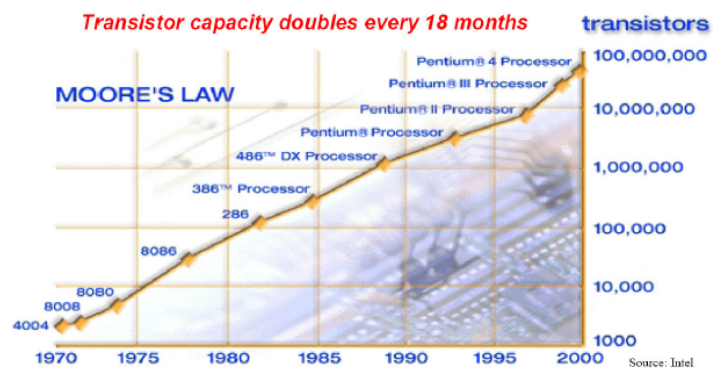
In 1965, Gordon Moore, co-founder of Intel, indicated that the number of transistors per square inch on integrated circuits had doubled every year since the integrated circuit was invented. Moore predicted that this trend would continue for the foreseeable future.

Evolution of CPUs

- Intel® Core™ i7
  - ☐ Intel® Core™ i7-5960X Processor Extreme Edition
  - ☐ (20M Cache, up to 3.50 GHz)
  - ☐ 8 Cores, 16 Threads
  - ☐ 64 bit Instruction Set

http://www.intel.com/technology/architecture-silicon/2billion.htm

Microprocessor-based Systems
Memory Types

☐ **R/W: Read/Write Memory; also called RAM**
  - It is volatile (losses information as power is removed)
  - Write means the processor can store information
  - Read means the processor can receive information from the memory
  - Acts like a Blackboard!

☐ **ROM: Read-Only memory;**
  - It is typically non-volatile (permanent) – can be erasable
  - It is similar to a Page from your textbook

Microprocessor-based Systems
Memory Classification

Basic Technologies:
Semiconductor
Magnetic
Optical
(or combination)

System Memory

Read/Write Memory R/WM

Read-Only Memory ROM

Erasable Memory

Permanent Memory

• Static R/WM

• Dynamic R/WM

• EPROM
• EEPROM
• Flash Memory

• Masked ROM
• PROM

Expensive
Fast/

Cheap
Slow

Onetime programmable

Electronically Erasable PROM

Microprocessor-based Systems
Memory Cla...

–one transistor and one capacitor to store a bit
–Leakage problem, thus requires refreshing
–Used for dynamic data/ program storage
–Cheap and slow!

–4/6 transistor to save a single bit
- Volatile
- Fast but expensive

Read-Only Memory ROM

Erasable Memory

Permanent Memory

• Static R/WM

• Dynamic R/WM

• EPROM
• EEPROM
• Flash Memory

• Masked ROM
• PROM

Expensive
Fast/

Cheap
Slow

Onetime programmable

Electronically Erasable PROM

Erasable ROMs

- Masked Programmed ROM
  - □ Programmed by the manufacturer
- Programmable ROM (PROM)
  - □ Can be programmed in the field via the programmer
- Erasable Programmable ROM (EPROM)
  - □ Uses ultraviolet light to erase (through a quartz window)
  - □ OTP refers to one-time programmable
- Electrically Erasable Programmable ROM (EEPROM)
  - □ Each program location can be individually erased
  - □ Expensive
  - □ Requires programmer
- FLASH
  - □ Can be programmed in-circuit (in-system)
  - □ Easy to erase (no programmer)
  - □ Only one section can be erased/written at a time (typically 64 bytes at a time)
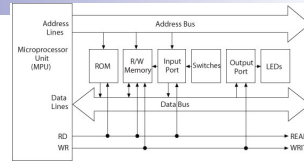
Microprocessor-based Systems
I/O Ports

- **The way the computer communicates with the outside world devices**
- **I/O ports are connected to Peripherals**
  - □ Peripherals are I/O devices
    - **Input devices**
    - **Output devices**
  - □ Examples
    - **Printers and modems,**
    - **keyboard and mouse**
    - **scanner**
   - **Universal Serial Bus (USB)**
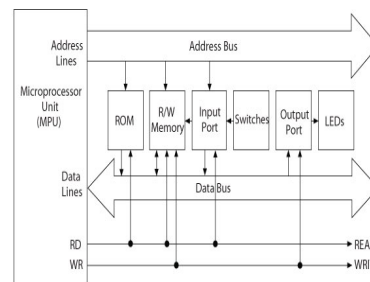
Microprocessor-based Systems - BUS

- **The three components – MPU, memory, and I/O – are connected by a group of wires called the BUS**

- **Address bus**
  - **consists of 16, 20, 24, or 32 parallel signal lines (wires) - <u>unidirectional</u>**
  - **these lines contain the address of the memory location to read or written**
- **Control bus**
  - ☐ **consists of 4 to 10 (or more) parallel <u>signal lines</u>**
  - ☐ **CPU sends signals along these lines to memory and to I/O ports**
    - **examples: Memory Read, Memory Write, I/O Read, I/O Write**
- **Data bus**
  - ☐ **consists of 8,16, or 32 parallel signal lines**
  - ☐ **<u>bi-directional</u>**
  - ☐ **only one device at a time can have its outputs enabled,**
  - ☐ **this requires the devices to have three-state output**

Expanded Microprocessor-Based System

1. **Note the directions of busses**

2. **What is the width of the address bus?**

3. **What is the value of the Address but to access the first register of the R/WM?**
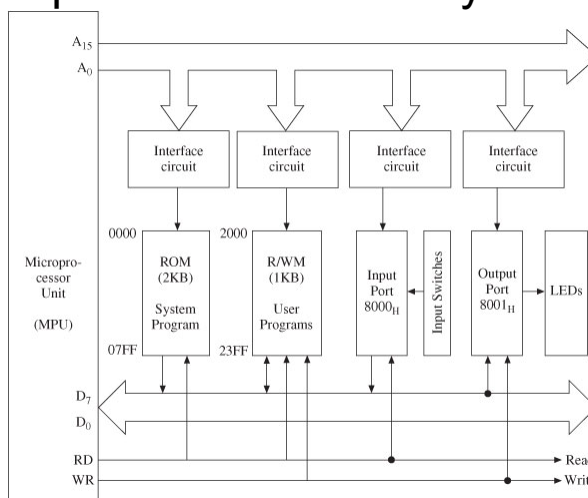
Remember: 111 1111 1111 = 2^11=2K

So what are microcontrollers?

First Microcontrollers

- IBM started using Intel processors in its PC
  - □ Intel started its 8042 and 8048 (8-bit microcontroller) – using in printers
- Apple Macintosh used Motorola 68000
- 1980 Intel abandoned microcontroller business
- By 1989 Microchip was a major player in designing microcontrollers
  - □ PIC: Peripheral Interface Controller

Embedded controllers
Software Characteristics

- No operating systems
- Execute a single program, tailored exactly to the controller hardware
- Assembly language (vs. High-level language)
  - Not transportable, machine specific
  - Programmer need to know CPU architecture
  - Speed
  - Program size
  - Uniqueness

Microcontroller Unit (MCU)
Block Diagram

- An integrated electronic computing and logic device that includes <u>three</u> major components on a <u>single chip</u>
  - Microprocessor
  - Memory
  - I/O ports
- Includes support devices
  - Timers
  - A/D converter
  - Serial I/O
  - Parallel Slave Port
- All components connected by common communication lines called the system bus.
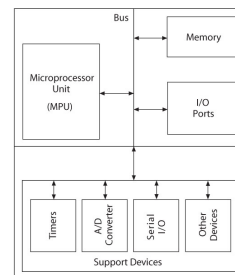
MCU Architecture

- RISC (Harvard)
  - Reduced instruction set computer
  - Simple operations
  - Simple addressing modes
  - Longer compiled program bust faster to execute
  - Uses pipelining
- CISC (Von Neuman)
  - Complex instruction set computer
  - More complex instructions (closer to high-level language support)

Bench marks: How to compare MCUs together
MIPS: Million Instructions / second (Useful when the compilers are the same)

Main 8-bit Controllers

- Microchip-- PIC® Microcontrollers
  - RISC architecture (reduced instruction set computer)
  - Has sold over 2 billion as of 2002
  - Cost effective and rich in peripherals
- Motorola– now Freescale
  - CISC architecture
  - Has hundreds of instructions
  - Examples: 68HC05, 68HC08, 68HC11
- Intel– now Marvell
  - CISC architecture
  - Has hundreds of instructions
  - Examples: 8051, 8052
  - Many difference manufacturers: Philips, Dallas/MAXIM Semiconductor, etc.
- Atmel
  - RISC architecture (reduced instruction set computer) –
  - Cost effective and rich in peripherals
  - AVR

1/2/17

Software: From Machine to High-Level Languages (1 of 3)

| High-level Language |
| Assembly Language |
| Machine Language |

- **Machine Language: binary instructions**
  - □ All programs are converted into the machine language of a processor for execution

  - □ Difficult to decipher and write
  - □ Prone to cause many errors in writing

Software: From Machine to High-Level Languages (2 of 3)

| High-level Language |
| Assembly Language |
| Machine Language |

- **Assembly Language: machine instructions represented in mnemonics**
  - □ Has one-to-one correspondence with machine instructions
  - □ Efficient in execution and use of memory; machine-specific and not easy to troubleshoot

| High-level Language |
| Assembly Language |
| Machine Language |

Software: From Machine to High-Level Languages (3 of 3)

- High-Level Languages (such as BASIC, C, and C++)
  - Written in statements of spoken languages (such as English)
    - machine independent
    - easy to write and troubleshoot
    - requires large memory and less efficient in execution

| Unsigned |
| Signed |

Data Format (8-bit) (1 of 4)

- Unsigned Integers: All eight bits (Bit0 to Bit7) represent the magnitude of a number
  - Range 0 to FF in Hex and 0 to 255 in decimal

Data Format (8-bit) (2 of 4)

Unsigned
Signed

- **Signed Integers:** Seven bits (Bit0 to Bit6) represent the magnitude of a number.
  - ☐ The eighth bit (Bit7) represents the sign of a number. The number is positive when Bit7 is zero and negative when Bit7 is one.
  - ☐ Positive numbers: 0 to 7F (0 to 127)
  - ☐ Negative numbers: 80 to FF (-1 to -128)
  - ☐ All negative numbers are represented in 2′s complement

Data Format (8-bit) (3 of 4)

- **Binary Coded Decimal Numbers (BCD)**
  - ☐ 8 bits of a number divided into groups of four, and each group represents a decimal digit from 0 to 9
  - ☐ Four-bit combinations from A through F in Hex are invalid in BCD numbers
    - Example: 0010 0101 represents the binary coding of the decimal number 25d which is different in value from 25H.

Data Format (8-bit) (4 of 4)

- American Standard Code for Information Interchange (ASCII)
  - Seven-bit alphanumeric code with 128 combinations (00 to 7F)
  - Represents English alphabet, decimal digits from 0 to 9, symbols, and commands

Storing Bits in Memory

- We can store in different memory types
  - EEPROM, FLASH, RAM, etc.
- In an 8-bit RAM
  - Each byte is stored in a single memory register
  - Each word is stored in two memory locations (registers)
  - DATA 0x1234
    - 0x12→REG11 (High-order byte)
      - 0001 0010
    - 0x34→REG10 (Low-order byte)
      - 0011 0100

Binary Addresses

Address		Register
0000		REG0
0001		REG1
0010		REG
0011		REG3
0100	A_3	REG4
0101		REG5
0110	A_2	REG6
0111		REG7
1000	A_1	REG8
1001		REG9
1010	A_0	REG10
1011		REG11
1100		REG12
1101		REG13
1110		REG14
1111		REG15

D_0 ← Data Lines → D_7
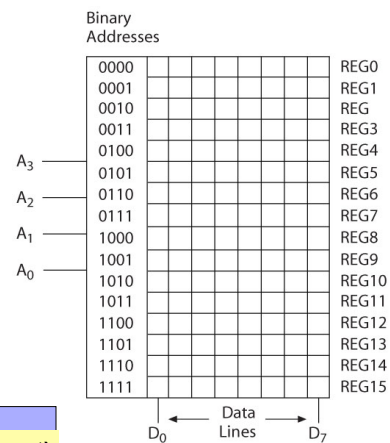
Remember -8→111 1000 (in two's complement)

Design Examples …..

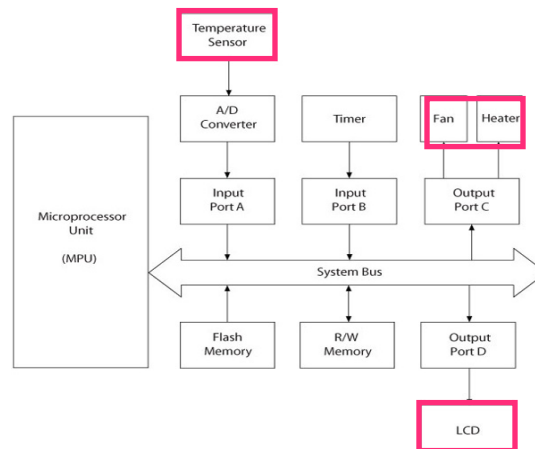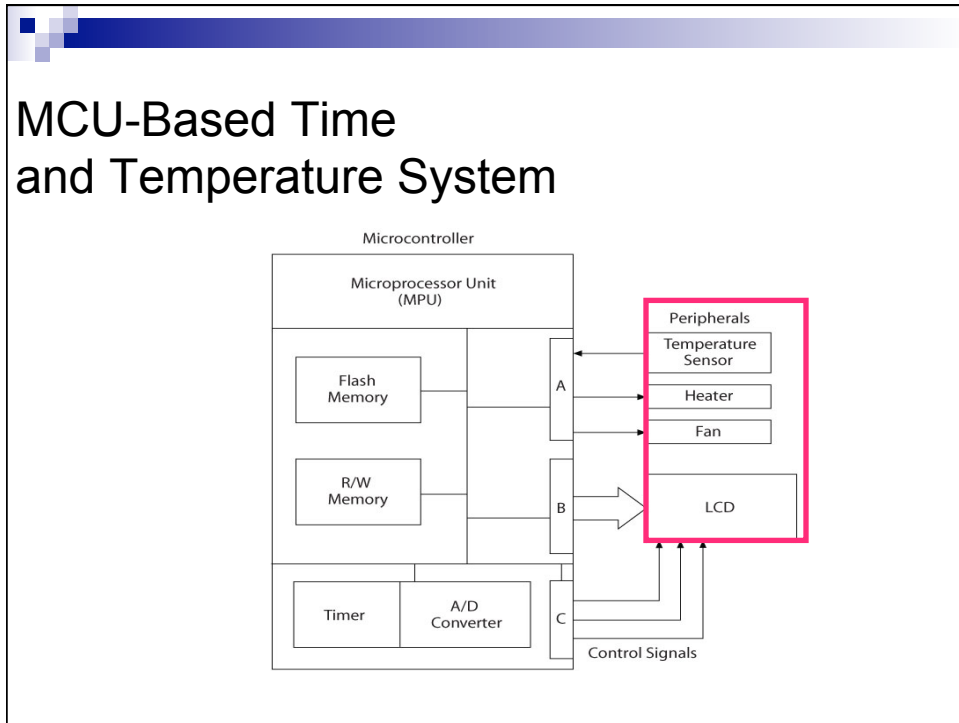Microcontrollers vs. Microprocessors

MPU-Based Time
and Temperature System

MCU-Based Time and Temperature System

Microcontroller
- Microprocessor Unit (MPU)
- Flash Memory
- R/W Memory
- Timer / A/D Converter
- A, B, C

Peripherals
- Temperature Sensor
- Heater
- Fan
- LCD

Control Signals

Kinetis K2x MCU Family Block Diagram

Core	System	Memories		Clocks
ARM® Cortex®-M4 50/72/100/120 MHz	Internal and External Watchdogs	Program Flash (32 KB to 1 MB)	SRAM 8 to 256 KB	Phase-Locked Loop
Debug Interfaces / DSP	16-Ch. DMA	Serial Programming Interface (EZ Port)	External Bus Interface (FlexBus)	Frequency-Locked Loop
Interrupt Controller / Floating-Point Unit (FPU)	Low-Leakage Wake-Up Unit	NAND Flash Controller	Cache	Low/High-Frequency Oscillators
	Memory Protection Unit (MPU)	FlexMemory (32 to 512 KB) (2 to 16 KB EE)		Internal Reference Clocks

Security and Integrity	Analog	Timers	Communication Interfaces		HMI
Cyclic Redundancy Check (CRC)	16-bit ADC	FlexTimer	I²C	I²S	GPIO
Cryptographic Acceleration Unit (CAU)	Analog Comparator with 6-bit DAC	Programmable Delay Block	UART (optional ISO 7816)	USB Device Charger Detect (DCD)	Xtrinsic Low-Power Touch-Sensing Interface
HW Tamper Detection Unit	PGA	Periodic Interrupt Timers	SPI	USB On-the-Go (LS/FS)	
Random Number Generator	12-bit DAC	Low-Power Timer	CAN	USB Voltage Regulator	
	Voltage Reference	Independent Real-Time Clock (RTC)	Secure Digital Host Controller (SDHC)	USB On-the-Go (HS)	
		Carrier Modulator Transmitter			

☐ Standard Feature   ⌐ ¬ Optional Feature