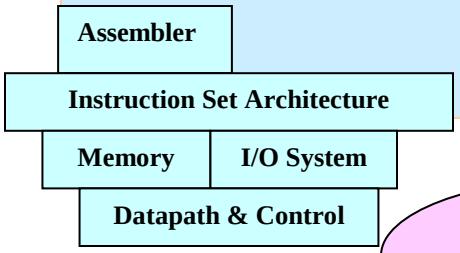
ELEC 3300
Introduction to Embedded Systems

Topic 2

More about Embedded Systems

Prof. Tim Woo

Course Overview



Introduction to Embedded Systems

More about Embedded Systems

Basic Computer Structure

MCU Main Board

Digital and Analog Interfacing

USART, IEEE1394, USB, I2C and SPI

Microcontroller Structure

A/D Port

Serial Port

CPU

External Memory Port

External Interrupt Port

Interrupt Organization

Timer and Counter

External Timer Port

Simple I/O Port

Buffering and Direct Memory Access (DMA)

Memory, Interfacing to Memory, Memory Timing and applications

Interfacing LCD

Motor Interfacing

In this course, STM32 is used as a driving vehicle for delivering the concepts.

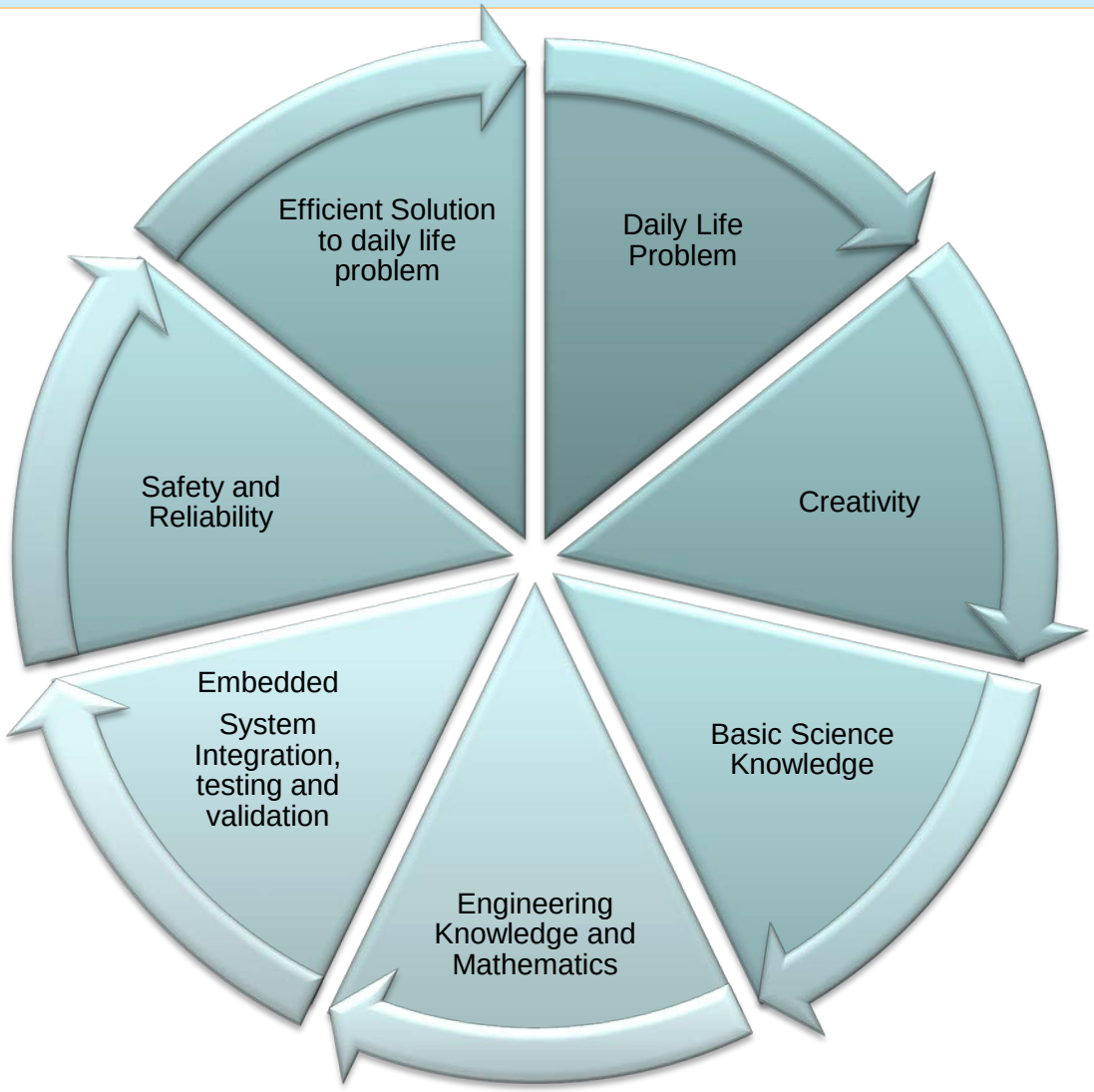
To be covered	In progress	Done

Expected Outcomes

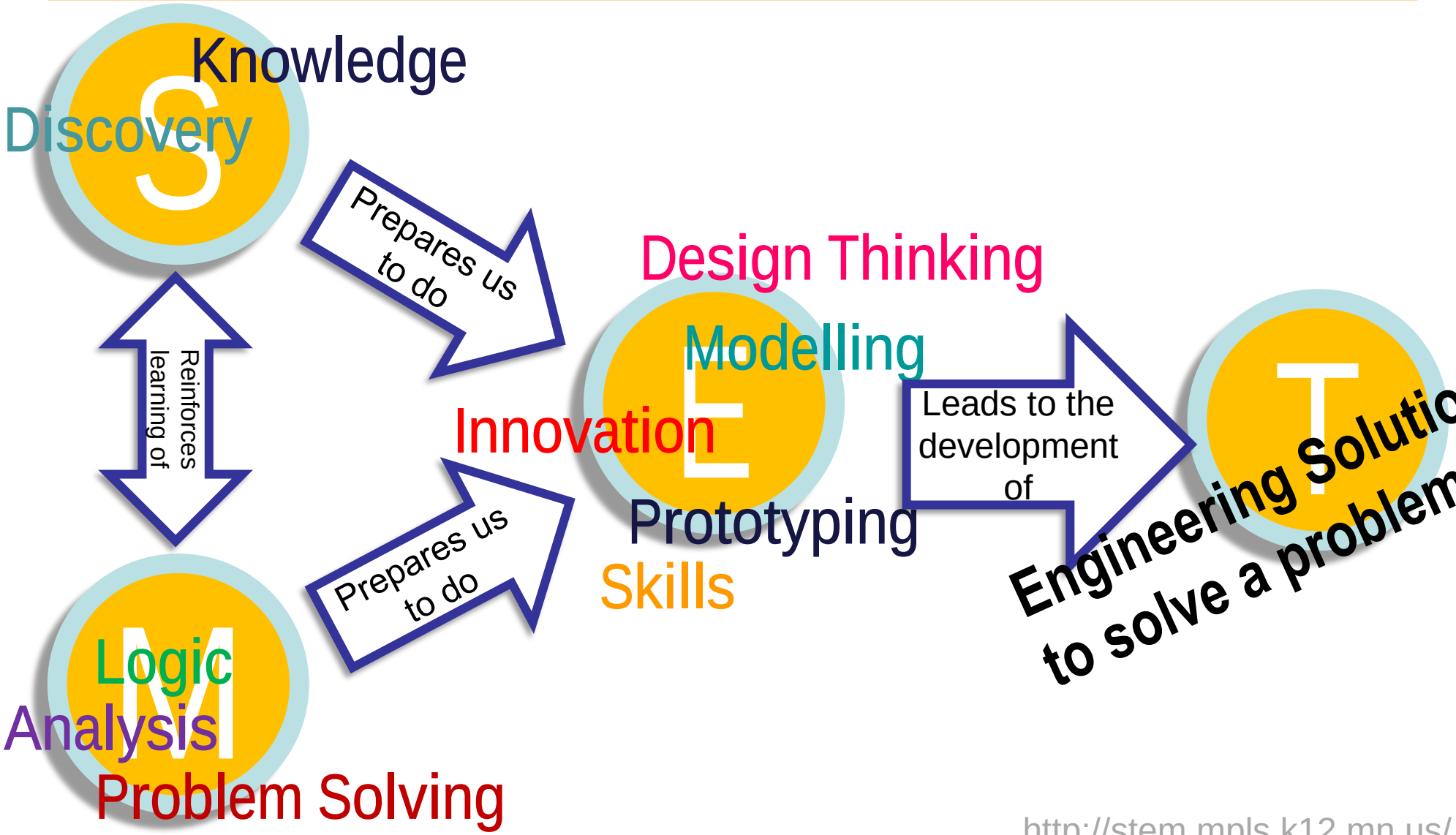
- On successful completion of this topic, you will be able to
  - Draft the project plan from your abstract idea
  - Investigate your design concept of an embedded systems

Project Design from abstract idea to implementation

How do engineers solve daily life problems?



How do engineers solve daily life problems?



http://stem.mpls.k12.mn.us/

Design architecture of an embedded system

- We have to consider 6 components but not limited to these

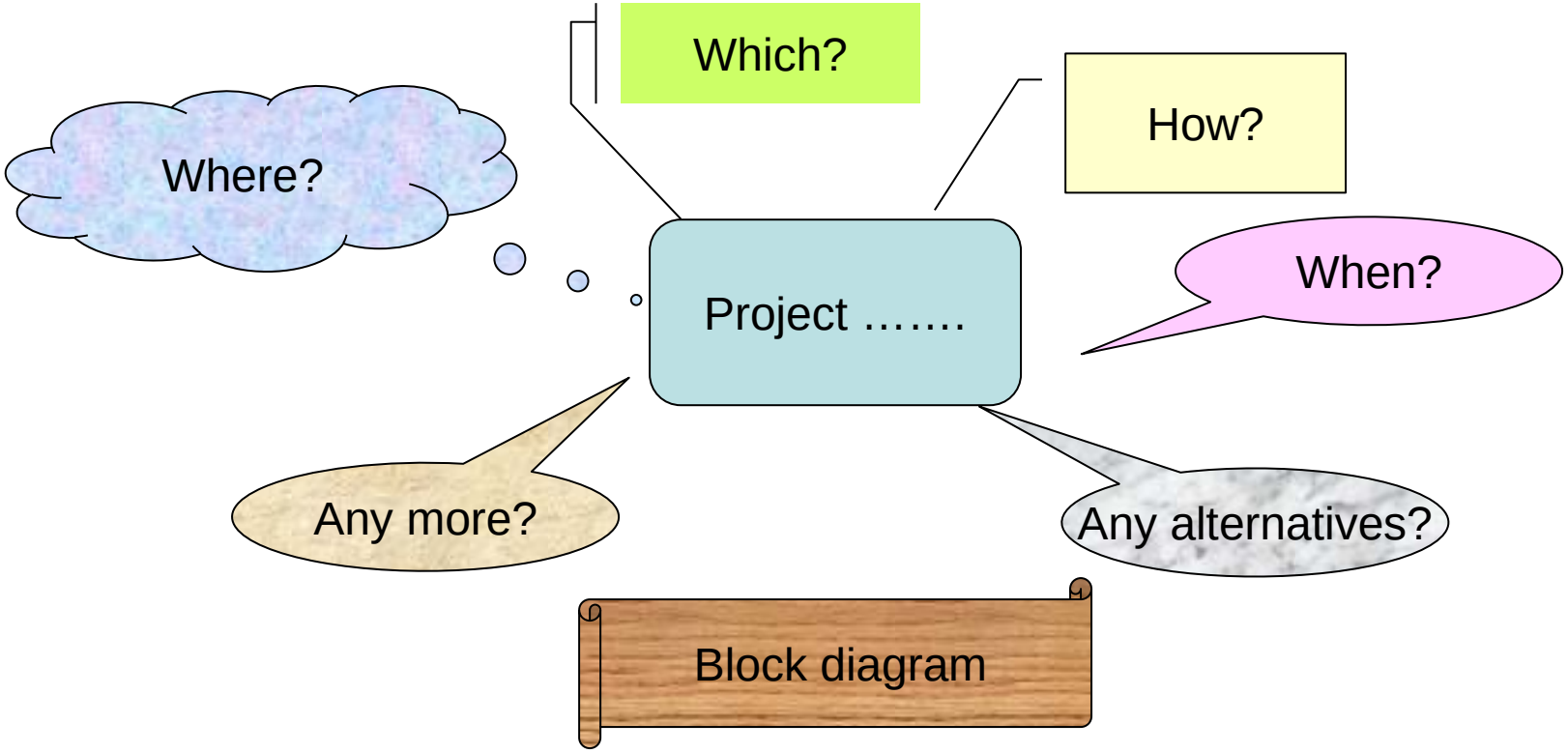
Products

Components

Description	Choices in this course
Abstract idea of project (Define the functionality of the system)	Many
Data format / representation	Many
Programming Language	C-language
Communication Protocol	Many
Physical connection (Pins assignment)	Many
Hardware devices (Microcontroller, Peripherals)	Microcontroller: STM32 ARM Platform Peripherals: Many

Design architecture of an embedded system

- Some inquiries, but not limited to these

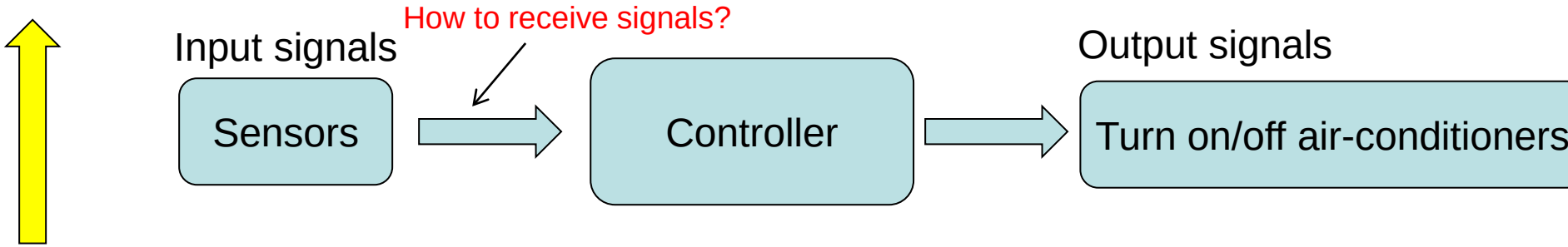


Design architecture of an embedded system

- Let's start from a simple project idea:

Product

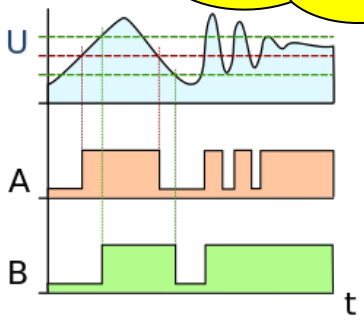
A device is used to monitor and control the temperature of this lecture room.



Components

Temperature sensor(s)

Design architecture of an embedded system



Schmitt trigger
(a dual threshold)

Monitor the temperature
of this lecture room

Measure the temperature?

Does the actuator always
trigger a change (on/off) ?

Temperature sensor(s)

Embedded system:
STM32 ARM Platform

Mapping between
temperature (Celsius)
and
electrical signal (Voltage)

Efficient Solution to daily life problem

Daily Life Problem

Creativity

Basic Science Knowledge

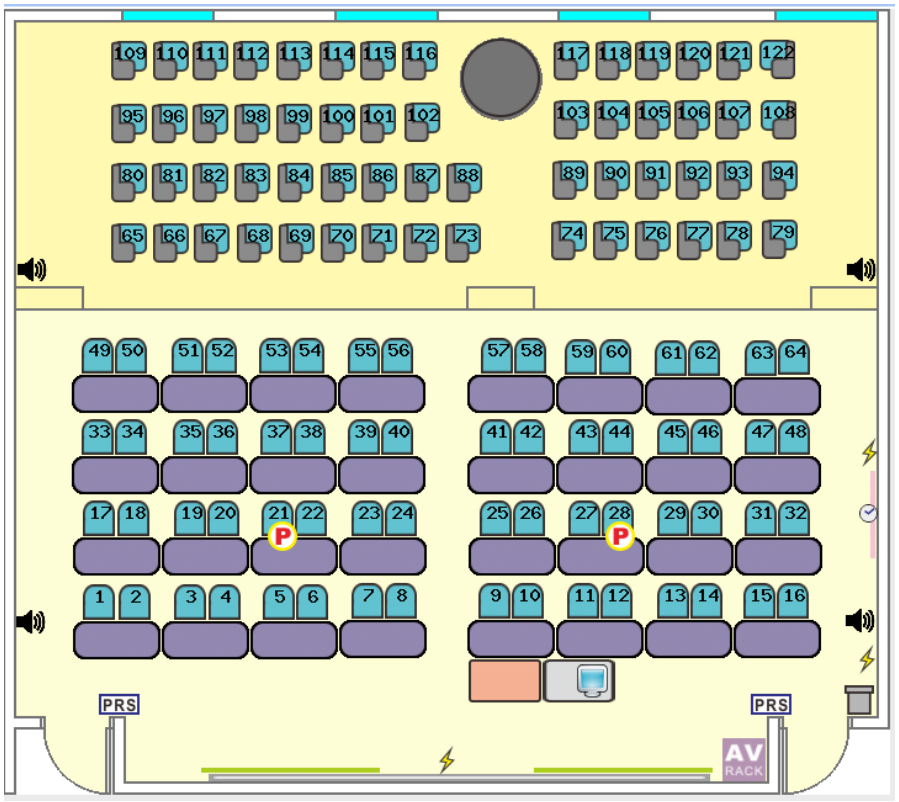
Engineering Knowledge and Mathematics

Embedded System Integration

Safety and Reliability

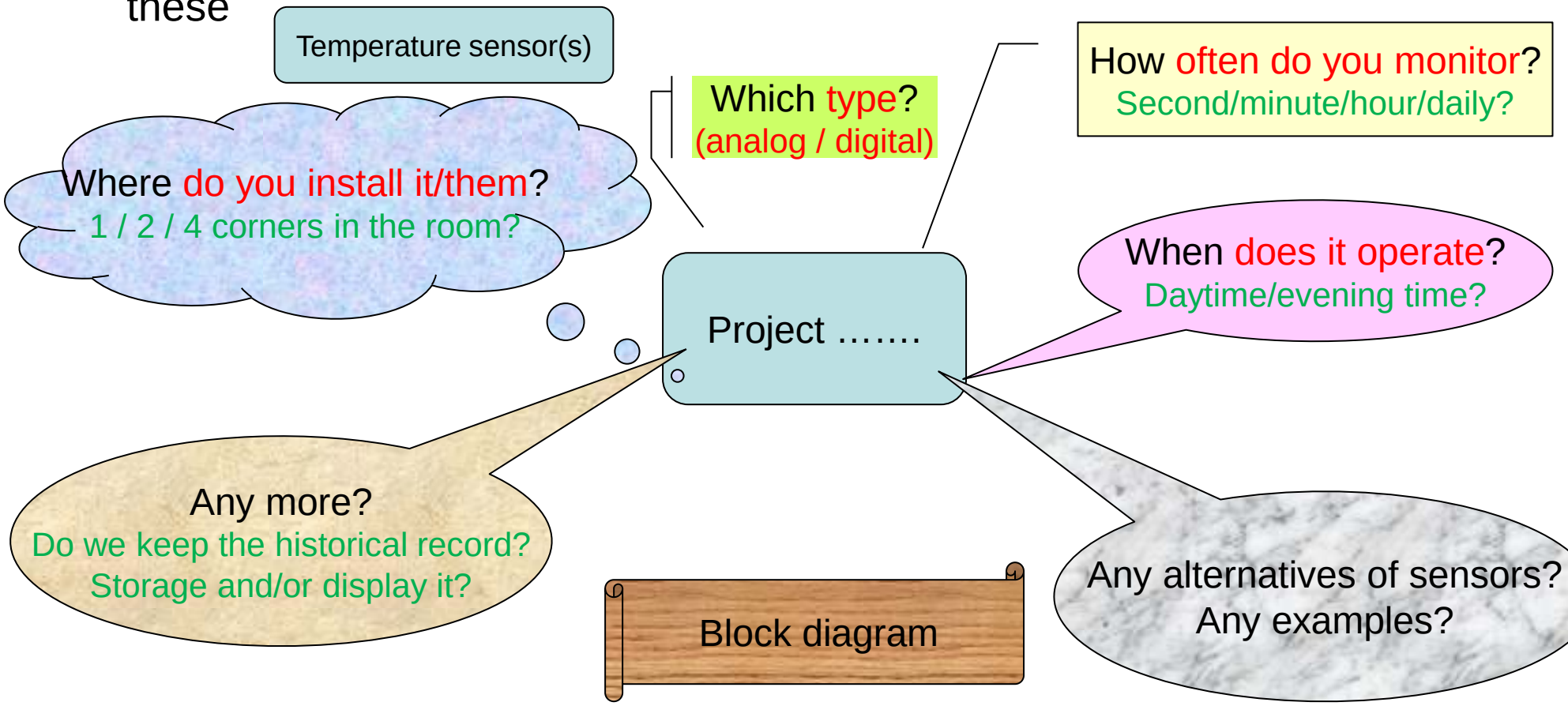
Design architecture of an embedded system

- Do you have any additional features in the design?



Design architecture of an embedded system

- Some inquiries may help you to explore the features, but not limited to these

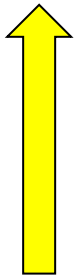


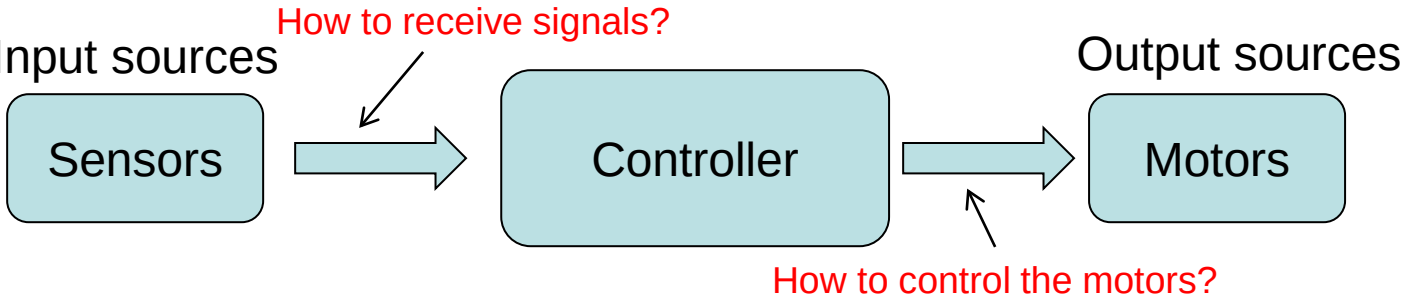
Design architecture of an embedded system

- Take another example: Fan control

Products

Now, you are asked to enhance the design of Fan control system.
Which features do you have in the design?





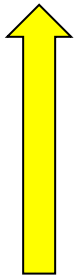
Components

Design architecture of an embedded system

- Another example:

Which features of monitor do you ask for?

Products



How to decode the image?

Input sources

Image

Controller

Output sources

Monitor

How to display onto monitor?

Components

Design architecture of an embedded system

- We have to consider 6 components but not limited to these

Description	Choices in this course
Abstract idea of project (Define the functionality of the system)	Many
Data format / representation	Many
Programming Language	C-language
Communication Protocol	Many
Physical connection (Pins assignment)	Many
Hardware devices (Microcontroller, Peripherals)	Microcontroller: STM32 ARM Platform Peripherals: Many

Products

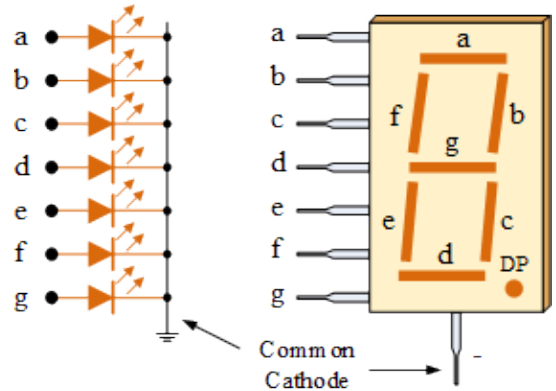
Components

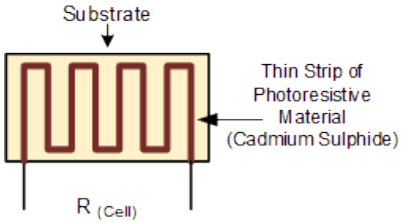
This is the part we have to fill up.

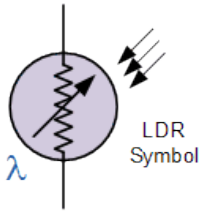
Design architecture of an embedded system

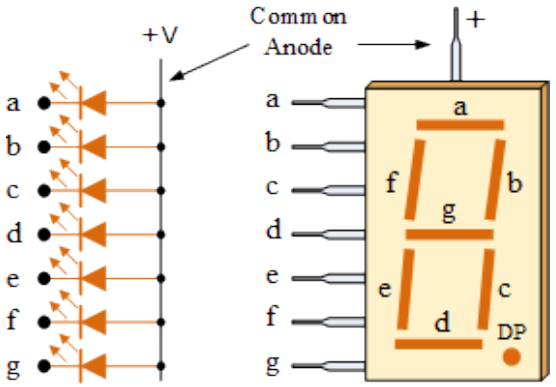
Get to know the hardware devices and their characteristics

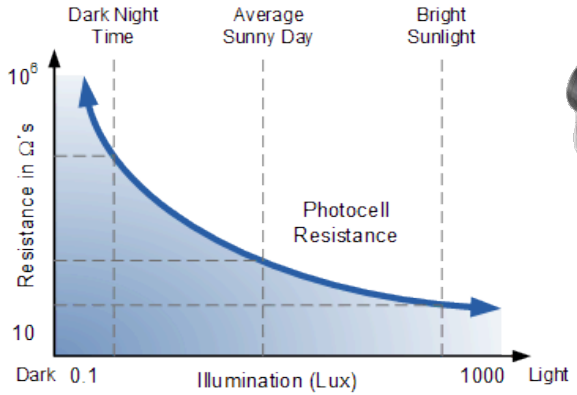
Description
Abstract idea of project (Define the functionality of the system)
Data format / representation
Programming Language
Communication Protocol
Physical connection (Pins assignment)
Hardware devices (Microcontroller, Peripherals)





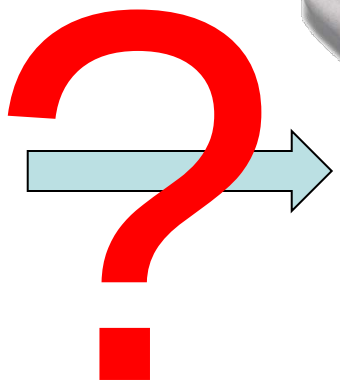






Design architecture of an embedded system

How to connect the embedded system board to other device(s)?



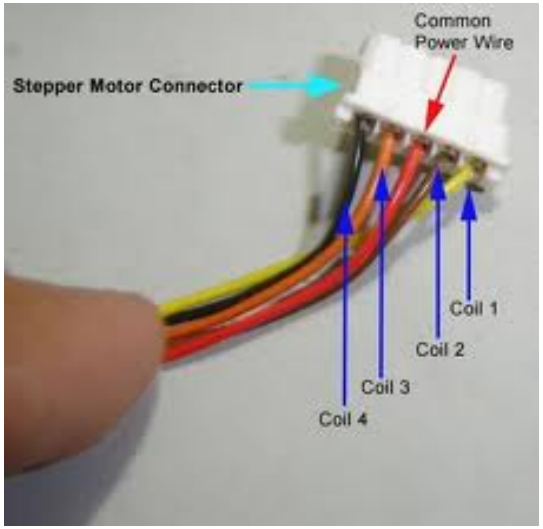
Pins assignment

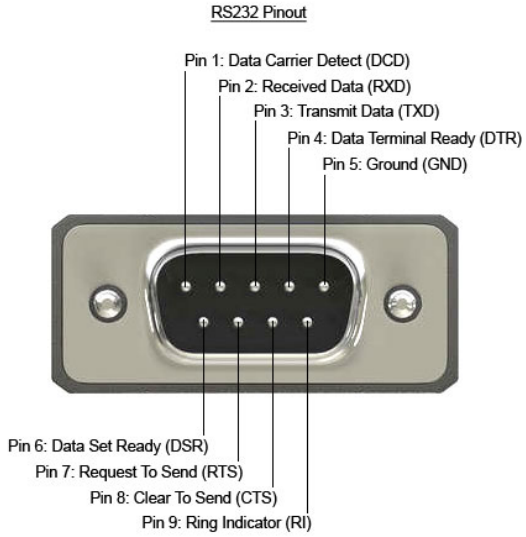
Description
Abstract idea of project (Define the functionality of the system)
Data format / representation
Programming Language
Communication Protocol
Physical connection (Pins assignment)
Hardware devices (Microcontroller, Peripherals)

Connect other devices

Typical concerns when we connect other devices to the embedded system board:

- Physical Connections
  - GPIO
  - Serial port / USB port





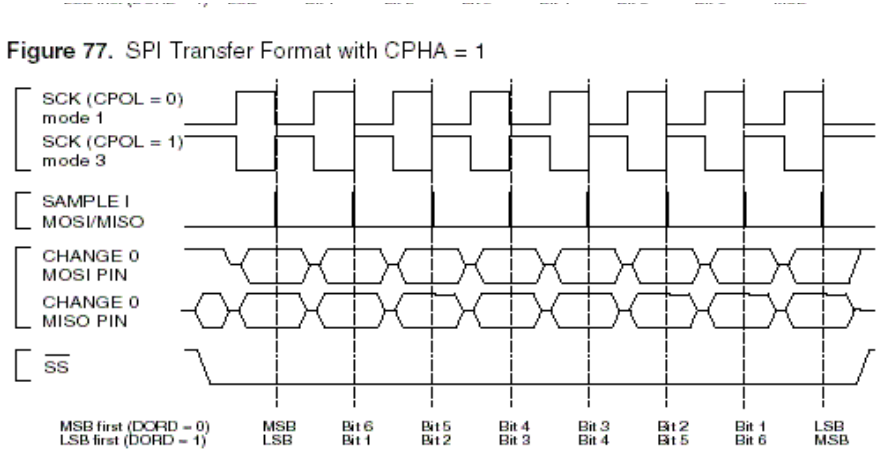
Description
Abstract idea of project (Define the functionality of the system)
Data format / representation
Programming Language
Communication Protocol
Physical connection (Pins assignment)
Hardware devices (Microcontroller, Peripherals)

Connect other devices

Typical concerns when we connect other devices to the embedded system board:

- Communication Protocol
  - Example: TCP/IP, USB, I^2C
- This includes
  - Signal Types
  - Signal Direction flow

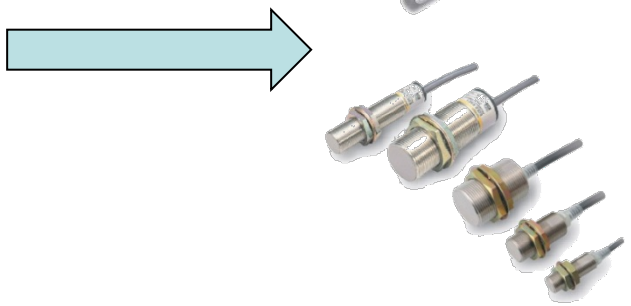
Description
Abstract idea of project (Define the functionality of the system)
Data format / representation
Programming Language
Communication Protocol
Physical connection (Pins assignment)
Hardware devices (Microcontroller, Peripherals)



Design architecture of an embedded system

- Programming Language
- Control other devices

Control



Description
Abstract idea of project (Define the functionality of the system)
Data format / representation
Programming Language
Communication Protocol
Physical connection (Pins assignment)
Hardware devices (Microcontroller, Peripherals)

How to program the MCU board to control other device(s)?

High-level and low-level programming languages

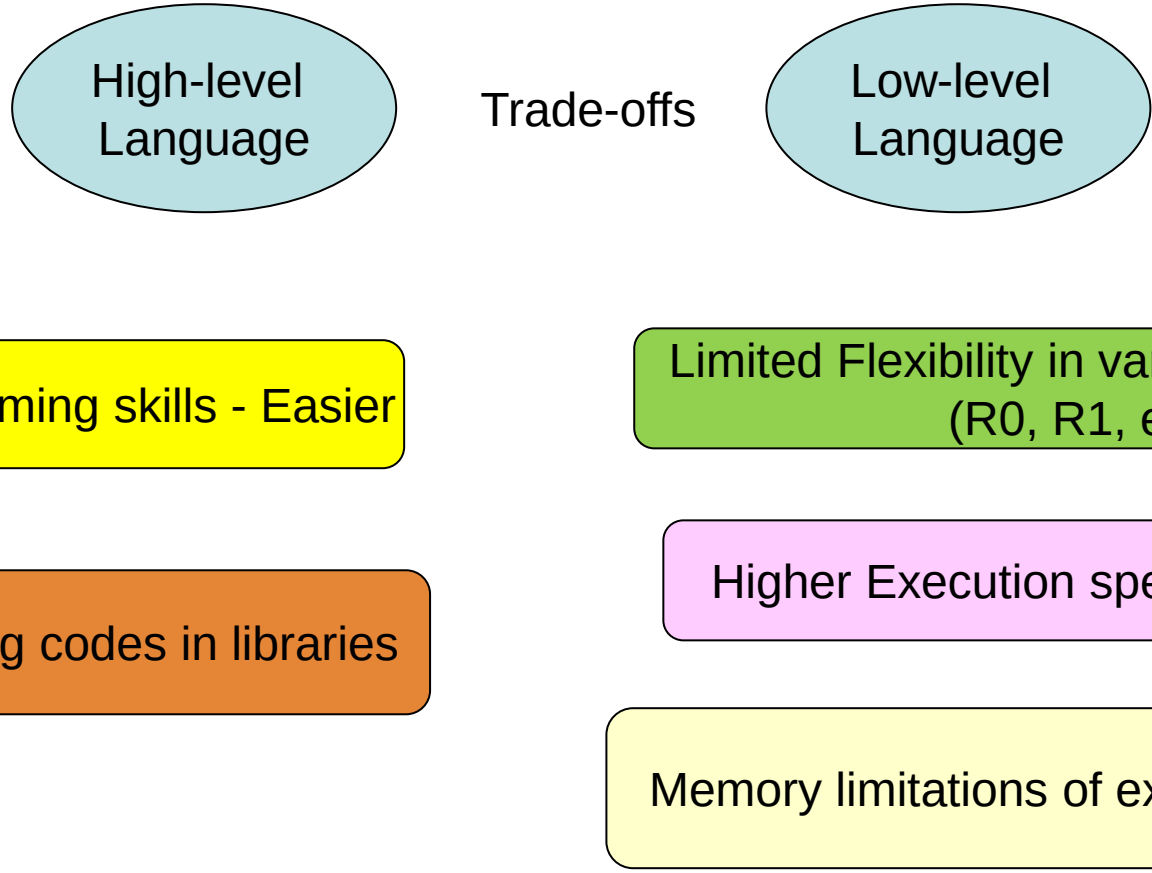
High-level Language

```
void main()
{
      int i, z = 0;




      for (i = 10, i>0,i--)
      {
          z+=i;
      }

}
```

Low-level Language

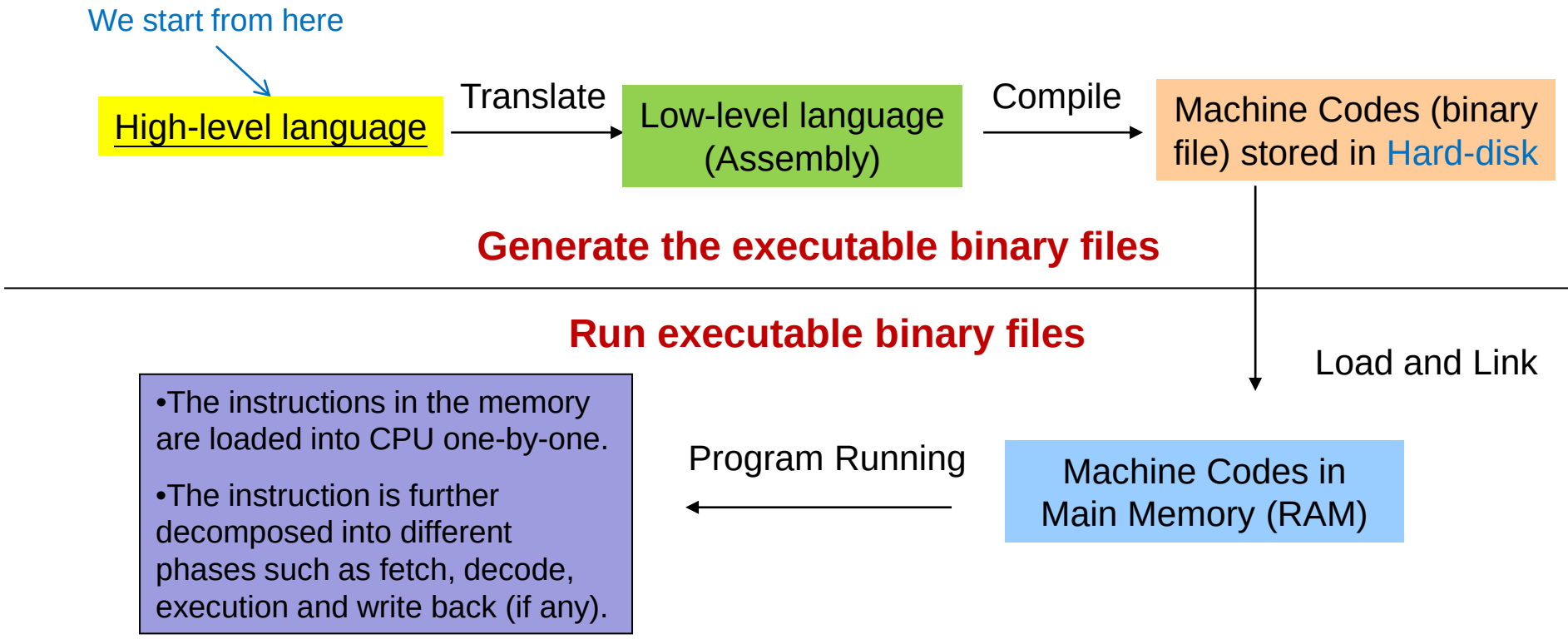
```
ORG   00H

      MOV  A, #0     ; A is accumulator
                     ; for addition operation
      MOV  R0, #0    ; Assign: R0 = z
      MOV  R1, #10 ; Assign: R1 = i

LOOP:
      ADD   A, R1
      DJNZ R1, LOOP
      MOV  R0, A


END
```

High-level and low-level programming languages



From high-level programming languages to program running

- Typical method on bringing high-level programming language to execution



From high-level programming languages to program running

- Example on translation

High-level language
C-language

Low-level language
8051 Assembly Language

Initialization

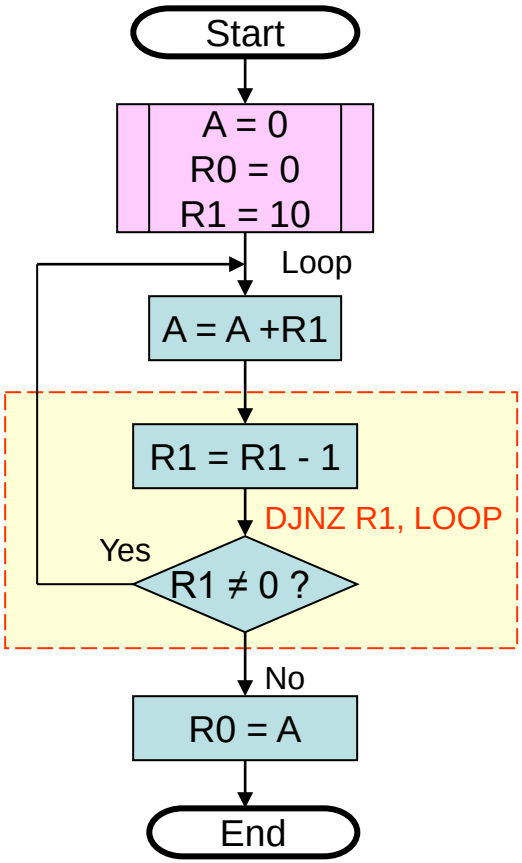
```
void main()
{
    int i, z = 0;

    for (i = 10, i>0,i--)
    {
        z+=i;
    }

}
```

```
ORG   00H

      MOV  A, #0     ; A is accumulator
                     ; for addition operation
      MOV  R0, #0    ; Assign: R0 = z
      MOV  R1, #10 ; Assign: R1 = i

LOOP:
      ADD   A, R1
      DJNZ R1, LOOP
      MOV  R0, A

END
```

implementation

Opcode | Operand(s)

Flow chart



From high-level programming languages to program running

- Example (Cont'd)

Initialization

Low-level language
8051 Assembly Language

```
ORG   00H
      MOV A, #0      ; A is accumulator
                     ; for addition operation
      MOV R0, #0     ; Assign: R0 = z
      MOV R1, #10 ; Assign: R1 = i
LOOP:
      ADD A, R1
      DJNZ R1, LOOP
      MOV R0, A
END
```

implementation

Need to know where the data / instruction is stored in the hardware system such as ROM / RAM

Address of R0
Address of R1
Address of A

Software

8051 Machine Codes in Main Memory

Code Segment (Internal Code Memory) (ROM)

Address (in Hex)	Content (in Hex)
0000	74
0001	00 ; 7400 (2 bytes) represents MOV A, #0
0002	78
0003	00 ; 7800 (2 bytes) represents MOV R0, #0
0004	79
0005	0A ; 790A (2 bytes) represents MOV R1, #10
0006	29 ; 29 (1 byte) represents ADD A, R1
0007	D9 ;
0008	FD ; D9FD (2 bytes) represents DJNZ R1, LOOP (actually FD means jump 3 bytes backward, 0006-0009 = -3)
0009	F8 ; F8 (1 bytes) represents MOV R0, A

Data Segment (Internal Data Memory) (RAM)

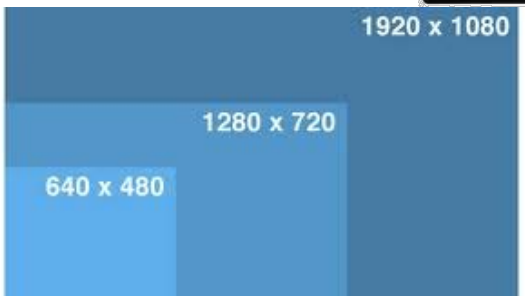
Address	Content (in Hex)
00	00 ; Value of R0, assume data bank 0 is selected
01	10 ; Value of R1, assume data bank 0 is selected
E0	00 ; Value of Accumulator

Hardware

Design architecture of an embedded system

Description
Abstract idea of project (Define the functionality of the system)
Data format / representation
Programming Language
Communication Protocol
Physical connection (Pins assignment)
Hardware devices (Microcontroller, Peripherals)

- Data format / representation
- Bitmap format:
  - Image resolutions



  - Color format

black and white

256-level grey scale

24-bit true color

32-bit true color

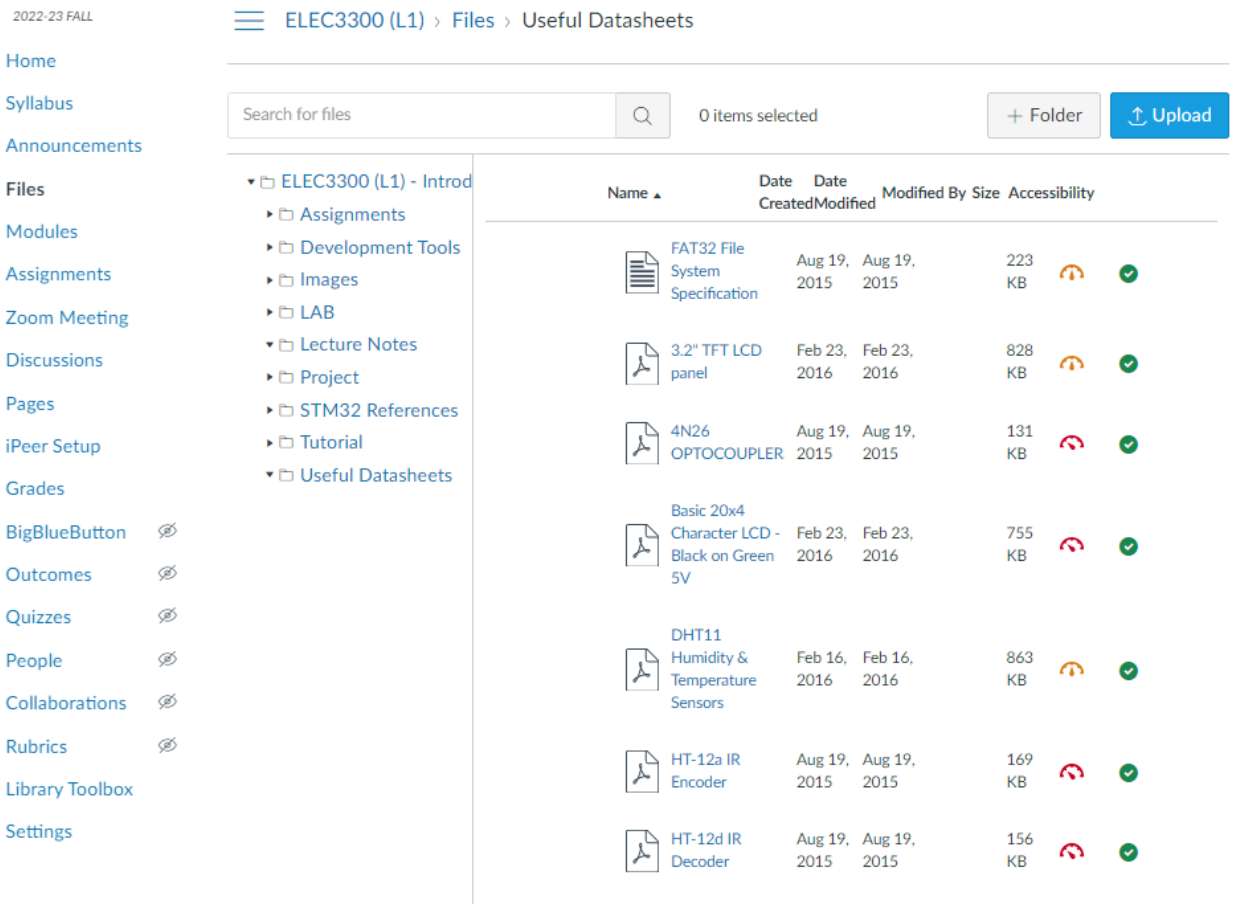
Design architecture of an embedded system

- Class exercise: Let's complete the following information of DHT11

Description	Choices in this course
Abstract idea of project (Define the functionality of the system)	What's are measured data of this sensor? Do you have any other features in the project?
Data format / representation	What is the data format? How to present RH and T?
Programming Language	Which Programming language will you use?
Communication Protocol	How to communicate between MCU and DHT11 sensor?
Physical connection (Pins assignment)	How many pins are there?
Hardware devices (Microcontroller, Peripherals)	Which devices / components do you need?

Design architecture of an embedded system

- Class exercise: Let's complete these information of DHT11
  - Tips: Search the Datasheet of DHT 11 via internet / CANVAS site



Design architecture of an embedded system

- We have to consider 6 components but not limited to these

Products

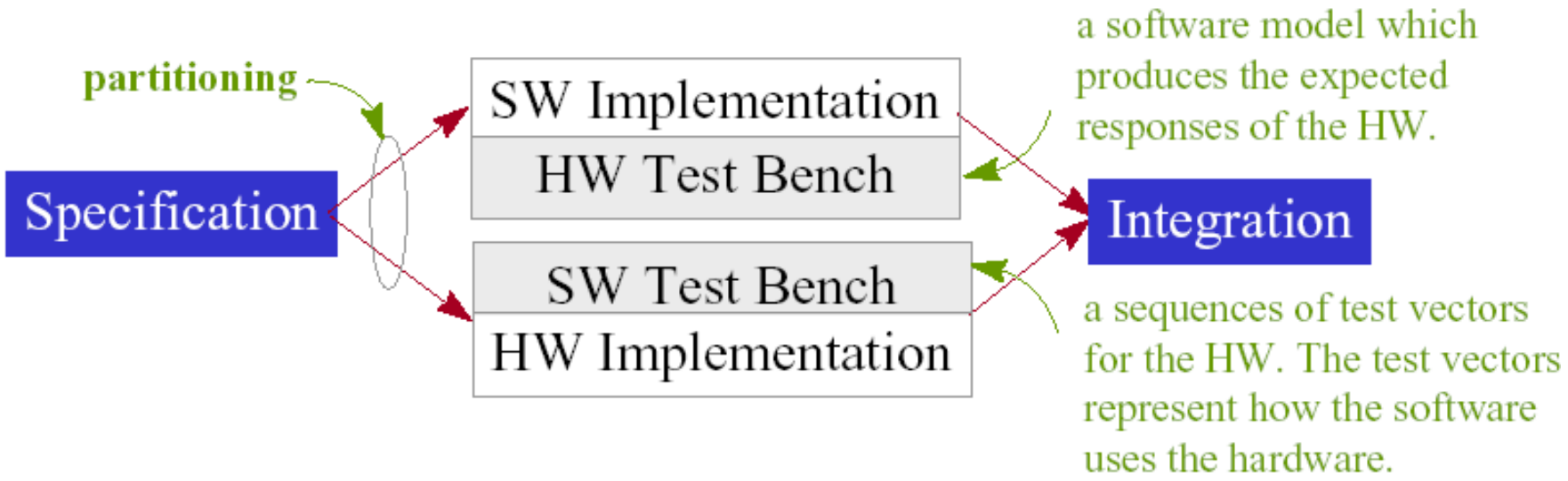
Description	Choices in this course
Abstract idea of project (Define the functionality of the system)	Many
Data format / representation	Many
Programming Language	C-language
Communication Protocol	Many
Physical connection (Pins assignment)	Many
Hardware devices (Microcontroller, Peripherals)	Microcontroller: STM32 ARM Platform Peripherals: Many

Components

Please apply this concepts in your laboratory experiments and project design.

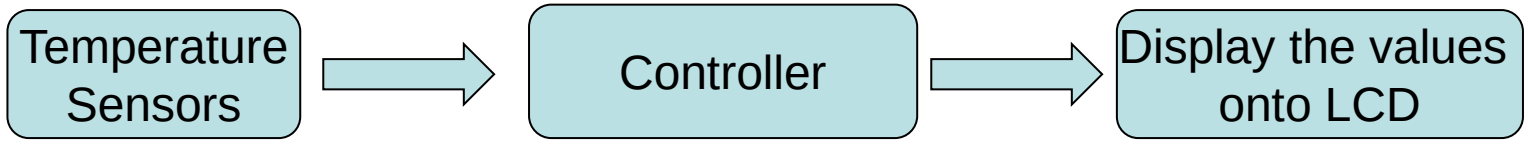
Mixed Hardware/Software Co-Development

- Traditionally, in a mixed hardware/software system, hardware and software are seen as independent
- First partition into H/W and S/W, and then development independently
- In general, changes in hardware imply changes in software and vice versa
- The overall verification is not done until the integration phase, which means that the cost of detecting hardware/software errors is very high

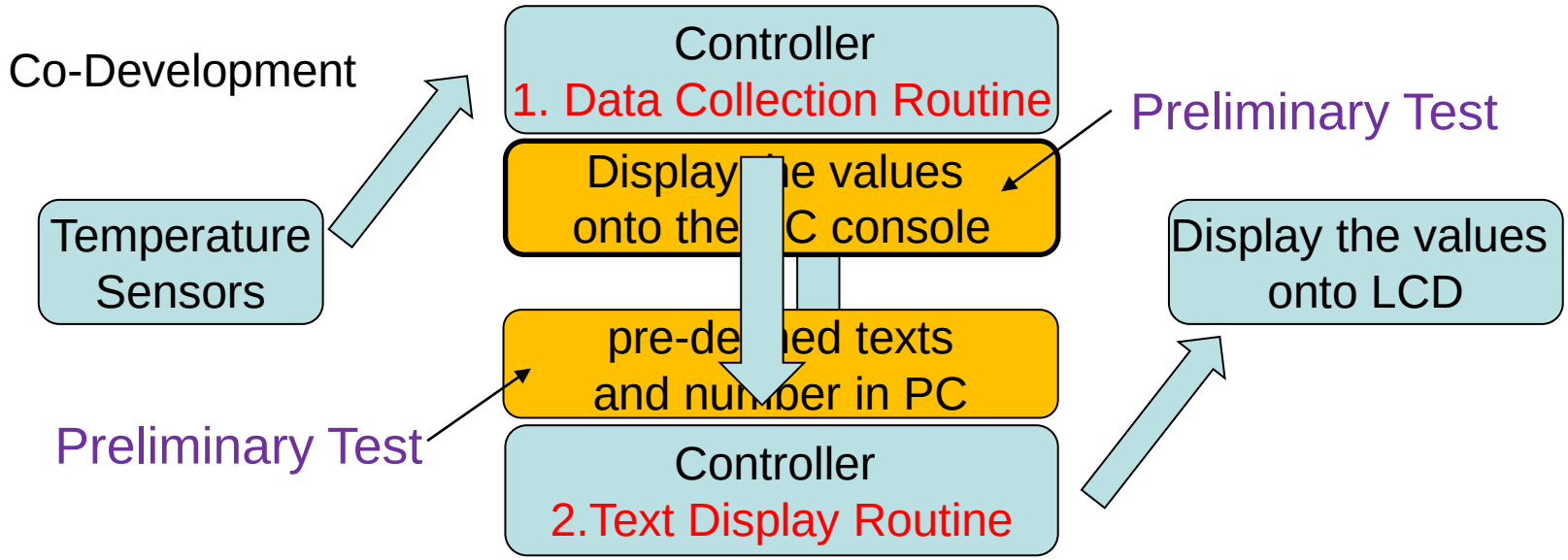


Mixed Hardware/Software Co-Development

- Take an example:

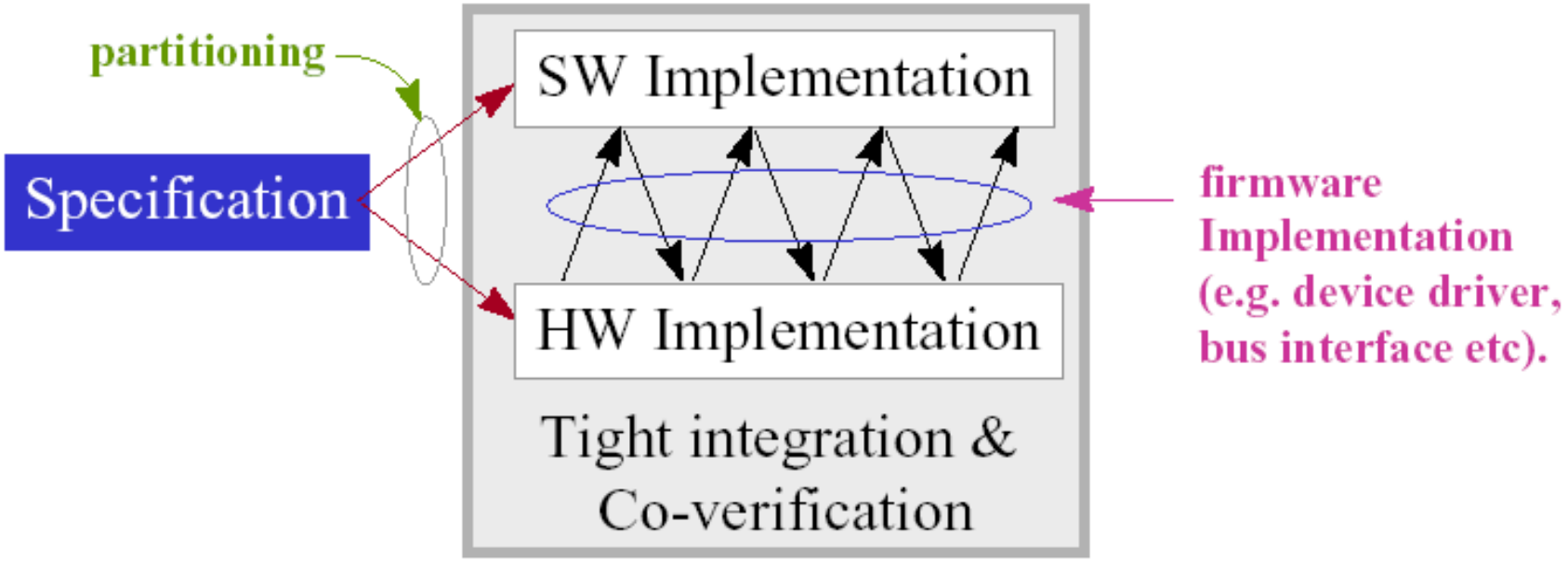


Data Collection Routine and Text Display Routine



Mixed Hardware/Software Co-Development

- It is obviously better if the hardware and software engineers can work together through the design and integration so that debugging can be done during the development rather than during the post integration process



Example

- Mixed Hardware/Software Co-Development
  - Japan Robot Trash Can

Trash can level: JAPAN

Mechanical Design

Electronics Circuit Design

Joystick controller

Wireless communication

Computer

Kinect

Object detection algorithm

https://www.youtube.com/watch?v=ZNWd4FFYDv0

What is the impact of embedded system?

Challenge

- A doctor is doing a surgery for a patient. Meanwhile, s/he would like to locate the cancer cells. How could you advise her / him?
  - Review the corresponding X-ray film (which was took before)
  - By experience
  - Consults with his / her colleagues
  - Use a device to “see” the cancer cells in the real-time

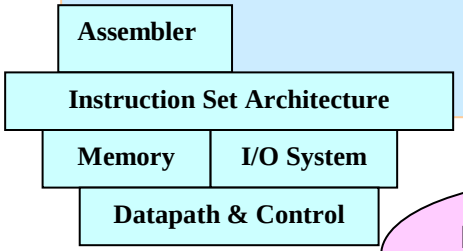
Hi-tech goggles detects cancer cells

https://www.insidescience.org/video/cancer-glasses-detect-tiny-tumors

Reflection (Self-evaluation)

- Do you ……
  - Draft a roadmap / design plan of your design project?
  - Understand the six design layers of your project?
  - Describe the Mixed Hardware/Software Co-Development?
  - Design embedded systems in tackling any challenges in COVID-19 pandemic?

Course Overview

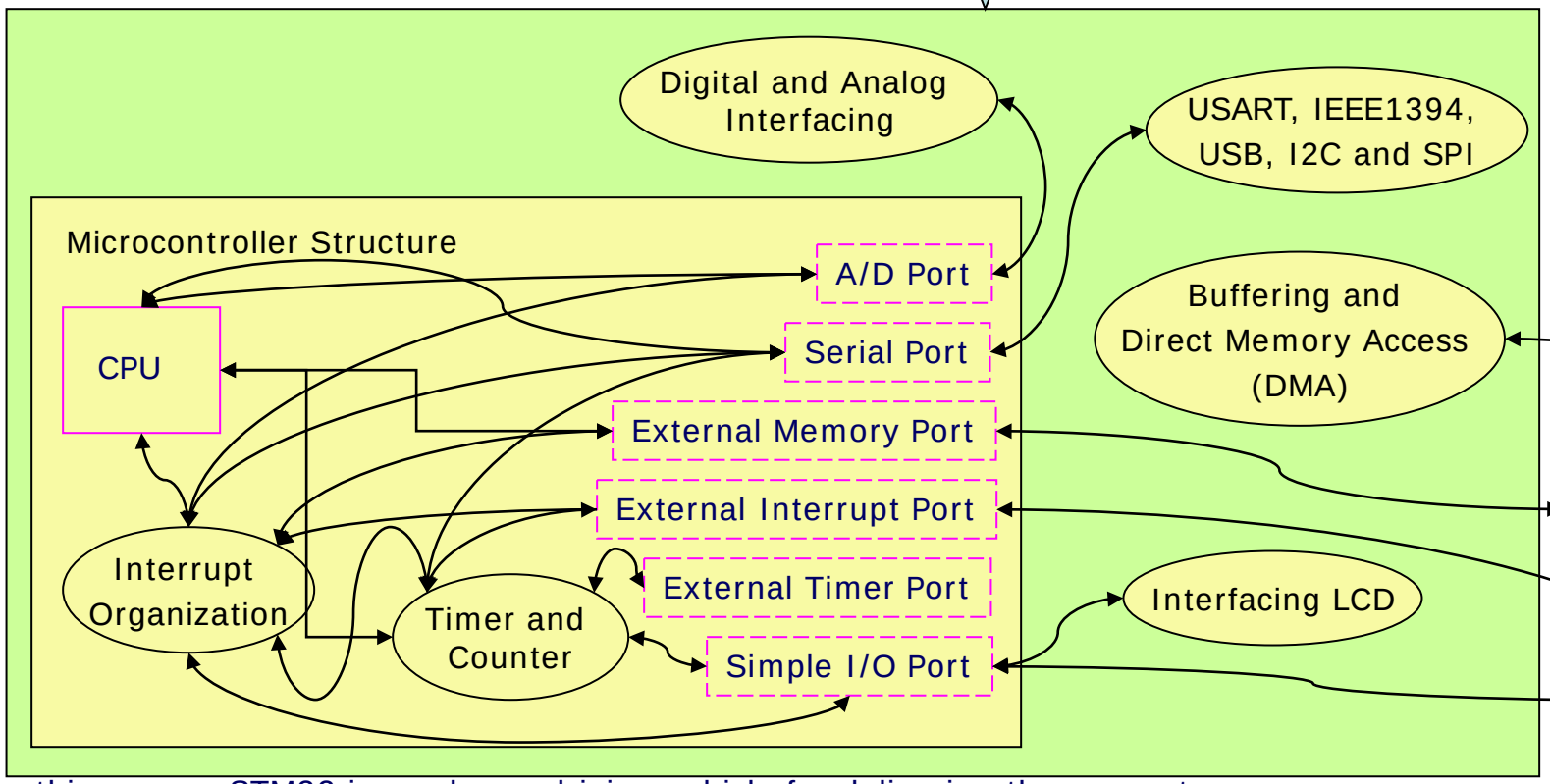


Introduction to Embedded Systems

More about Embedded Systems

Basic Computer Structure

MCU Main Board

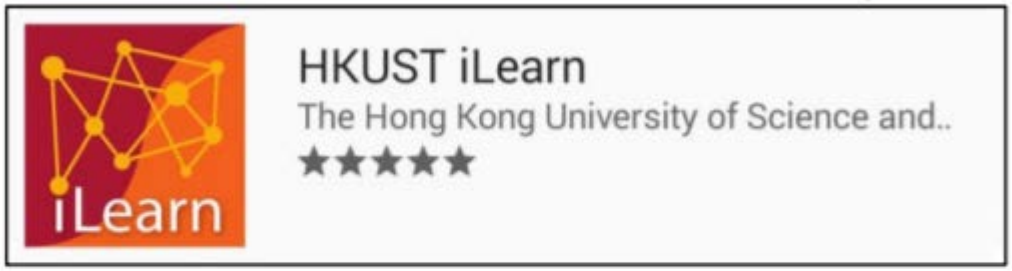


In this course, STM32 is used as a driving vehicle for delivering the concepts.

To be covered	In progress	Done

In-class activities

For Android devices, search **HKUST iLearn** at Play Store.



For iOS devices, search **HKUST iLearn** at App Store.

Topic 2 More about Embedded Systems