4 UNIT OF EMBEDDED SYSTEM

EMBEDDED HARDWARE

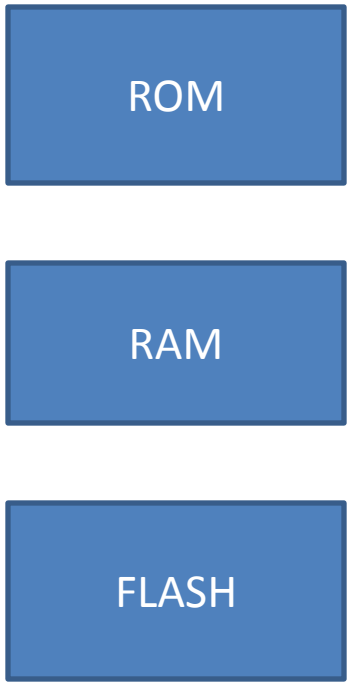
Embedded hardware

1) Memory map, i/o map,
2) interrupt map
3) processor family
4) external peripherals
5) memory – RAM , ROM, types of RAM and ROM
6)memory testing
7)CRC ,Flash memory

Memory map

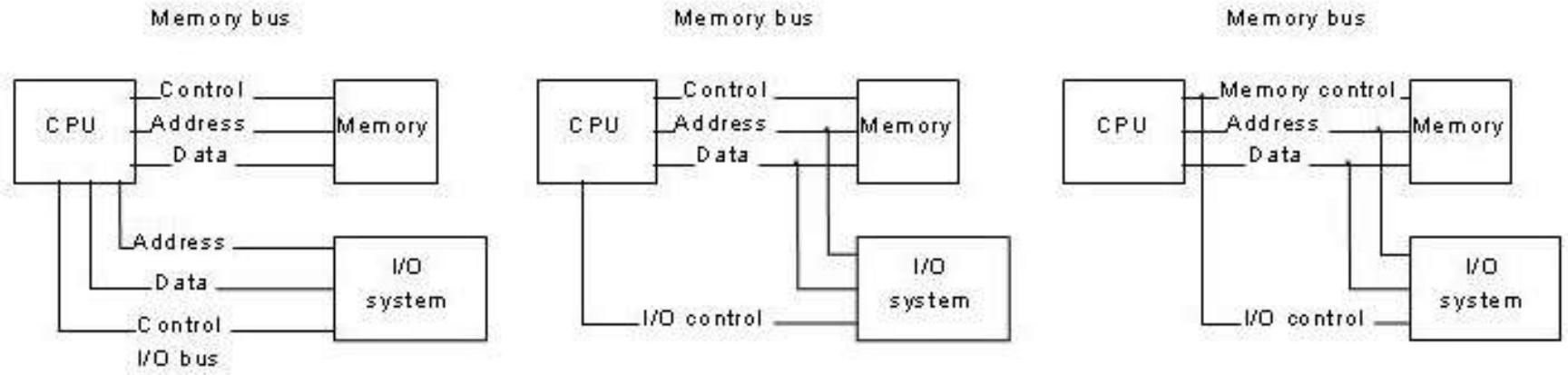
- Map to show the program and data
- allocation of the addresses to
- ROM, RAM, EEPROM or Flash in
- the system

- ROM : For program and input data
- RAM: For program stack
- Flash: For Current balance and other output data



Memory-Mapped I/O

- Memory-Mapped I/O is a mechanism by which the processor performs I/O access by using memory access techniques(An **access method** is a function of a mainframe operating system that enables access to data on disk, tape or other external devices). This is often put into effect because the memory bus is frequently much faster than the I/O bus. Another reason that memory mapped I/O might be used is that the architecture in use does not have a separate I/O bus.
- In memory mapped IO, certain range of CPU's address space is kept aside for the external peripherals. These locations can be accessed using the same instructions as used for other memory accesses. But instead, the read/writes to these addresses are interpreted as access to device rather than a location on the main memory.
- A CPU may expect a particular device at a fixed location or can dynamically assign a space for it.
- The way this works is that memory interfaces are often designed as a bus (a shared communications resource), where many devices are attached. These devices are usually arranged as master and slave devices, where a master device can send and receive data from any of the slave devices. A typical system would have:
- A CPU as the master
- One or more RAM and/or ROM devices for program code and data storage
- Peripheral devices for interfacing with the outside world. Examples of these might be a UART (serial communications), Display device or Input device



(a) Separate memory and I/O buses (isolated I/O)

(b) Shared address and data lines

(c) Shared address, data, and control lines (memory-mapped I/O)

Source: Heuring – Jordan: Computer Systems Architecture and Design

- **Memory-mapped I/O** (**MMIO**) and **port-mapped I/O** (**PMIO**) (which is also called *isolated I/O*) are two complementary methods of performing input/output between the CPU and peripheral devices in a computer. An alternative approach is using dedicated I/O processors—commonly known as channels on mainframe computers—that execute their own instructions.
- Memory-mapped I/O (not to be confused with memory-mapped file I/O) uses the same address bus to address both memory and I/O devices – the memory and registers of the I/O devices are mapped to (associated with) address values. So when an address is accessed by the CPU, it may refer to a portion of physical RAM, but it can also refer to memory of the I/O device. Thus, the CPU instructions used to access the memory can also be used for accessing devices.

Interrupt map in embedded hardware

- Interrupts
- Sometimes things will happen in a system when the processor is simply not ready. In fact, sometimes things change that require immediate attention. Can you imagine, sitting at your PC, that you were to hit buttons on the keyboard, and nothing happens on your computer? Maybe the processor was busy, and it just didnt check to see if you were hitting any buttons at that time. The solution to this problem is something called an "Interrupt." Interrupts are events that cause the microprocessor to stop what it is doing, and handle a high-priority task first. After the interrupt is handled, the microprocessor goes back to whatever it was doing before. In this way, we can be assured that high-priority inputs are never ignored.

- Hardware and Software
- There are two types of interrupts: Hardware and Software. Software interrupts are called from software, using a specified command. Hardware interrupts are triggered by peripheral devices outside the microcontroller. For instance, your embedded system may contain a timer that sends a pulse to the controller every second. Your microcontroller would wait until this pulse is received, and when the pulse comes, an interrupt would be triggered that would handle the signal.

- Interrupt Service Routines
- Interrupt Service Routines (ISR) are the portions of the program code that handle the interrupt requests. When an Interrupt is triggered (either a hardware or software interrupt), the processor breaks away from the current task, moves the instruction pointer to the ISR, and then continues operation. When the ISR has completed, the processor returns execution to the previous location.

- Interrupt Vector Table
- The "Interrupt Vector Table" is a list of every interrupt service routine. It is located at a fixed location in program memory.
  (Some processors expect the interrupt vector table to be a series of "call" instructions, each one followed by the address of the ISR. Other processors expect the interrupt vector table to hold just the ISR addresses alone.)
- You must make sure that every entry in the interrupt vector table is filled with the address of some actual ISR, even if it means making most of them point to the "do nothing and return from interrupt" ISR.

Processor family

- A microcontroller is a small computer on a single integrated circuit containing a processor, core, memory, and programmable input/output peripherals.
- Program memory in the form of NOR flash or OTP ROM is also often included on chip as well as a typically small amount of RAM.
- microcontroller are designed for embedded application in contrast to the microprocessor used in personal computer or general purpose application.
- Some microcontroller may use 4bit words and operate at clock rate frequencies as low as 4kHs , for low power consumption
- They will generally have the ability to retain functionality while waiting for an event such as a button press or other interrupt; power consumption while sleeping may be just nanowatts making many of them well suited for long lasting battery application

- Microcontroller usually contain from several to dozens of general purpose input/output pins.
- GPIO pins are software configurable to either an input or an output state,
- When GPIO pins are configured to an input state , they are often used to read sensors or external signals.
- Configured to the output state , GPIO pins can drive external devices such as LEDs or motors.

- Many embedded system need to read sensors that produce analog signal.
- The analog to digital converter is used to convert the incoming data into a form that the processor can recognize.
- In addition to the converters , many embedded microprocessor include a variety of timers as well.

- One of the most types of timers is the programmable Interval Timer (PIT).
- Once it reaches zero, it sends an interrupt to the processor indicating that it has finished counting.
- A PIT may either count down from some value to zero or up to capacity of the count register, overflowing to zero.

- Time processing unit is a sophisticated timer. In addition to counting down, the TPU can detect input events , generate output events and perform other useful operation.
- A dedicated pulse width modulation blocks matter it possible for the cpu to control power consumption , resistive loads and motors etc. without using lots of CPU RESOURCES
- Universal Asynchronous Receiver/Transmitter blocks makes it possible to receive and transmit data over a serial line with very little load on the CPU
- Dedicated on chip hardware is often help to communicate with other devices in digital format.

cots

- Real-time embedded systems are increasingly being built
- using commercial-off-the-shelf (COTS) components such as
- mass-produced peripherals and buses to reduce costs, timeto-
- market, and increase performance

- Integrating high-speed commercial-off-the-shelf (COTS)
- peripherals within a real-time system offers substantial benefits
- in terms of cost reduction, time-to-market, and overall
- performance.

- Due to mass production, COTS components
- are significantly cheaper to produce than their application
- Since COTS components are already designed,
- a system’s time-to-market can be reduced by reusing
- existing components instead of creating new ones. Additionally,
- overall performance of mass produced components
- is often significantly higher than custom made systems

- example, a PCI Express bus [12] can transfer data three orders
- of magnitude faster than the real-time SAFEbus [7].
- However, the main challenge when integrating COTS peripherals
- within a real-time system is the unpredictable timing
- of the I/O subsystem since COTS components are typically
- designed paying little or no attention to worst-case
- timing behaviors. In particular, we are concerned with I/O
- subsystems with high bandwidth requirements; a modern
- real-time system such as a search and rescue helicopter [16]
- may include several high-bandwidth components

- COTS-based components, and achieves high realtime
- bus utilization without degrading peripherals' throughput.
- The proposed framework acts like a "transparent layer"
- that does not add any additional burden at the operating system
- or user level, except for assigning a certain priority to each real-time I/O flow

Embedded firmware

- In electronicsystems and computing, **firmware** is the combination of persistent memory and program code and data stored in it.
- Typical examples of devices containing firmware areembedded systems (such as traffic lights, consumer appliances, and digital watches), computers, computer peripherals, mobile phones, and digital cameras.

- The firmware contained in these devices provides the control program for the device.
- Firmware is held in non-volatilememory devices such as ROM Changing the firmware of a device may rarely or never be done during its economic lifetime; some firmware memory devices are permanently installed and cannot be changed after manufacture.

- Common reasons for updating firmware include fixing bugs or adding features to the device. This may require physically changing ROM integrated circuits, or reprogramming flash memory with a special procedure.
- Firmware such as the program of an embedded system may be the only program that will run on the system and provide all of its functions.

Flash memory

- Flash memory is a nonvolatile memory that can be erased and reprogrammed in unit of memory called blocks.
- It is a variation of eeprom which is erased and rewritten at the byte level.
- Flash memory is often used to hold the control code such as the basic input/output system in computer
- When need to be changed the flash memory can be rewritten and make it easy to update but not useful as RAM .

- Flash memory gets its name because the microchip is organized so that the section of memory cells are erased in a single section .
- This memory is used in digital cellular phones , digital cameras, LAN switches, digital set-up boxes.
- Note (A chip that holds programs and data either temporarily or permanently. RAM chips are the computer's temporary workspace, while flash memory chips are used like disk drives (permanent until erased). ROM and PROM chips can never be changed, while EPROMs and EEPROMs can be modified.)

External peripherals

- Serial communication interface: A **serial communications interface** (**SCI**) is a device that enables the serial (one bit at a time) exchange of data between a microprocessor and peripherals such **..RS-232, RS-422,RS-485.**
- Synchronous serial communication interface:**Synchronous Serial Interface (SSI)** is a widely used serial interface standard for industrial applications between a master (e.g. controller) and a slave (e.g. sensor). Such as **I2C,SPI,SSC AND ESSI.**

- Universal serial bus(USB):USB was designed to standardize the connection of computer peripherals, such as keyboards, pointing devices, digital cameras, printers, portable media players, disk drivesand network adapters to personal computers, both to communicate and to supply electric power.

- MULTI MEDIA CARD
  :The **MultiMediaCard** (**MMC**) is a flash memory memory card standard. an MMC is used as a storage medium for a portable device, in a form that can easily be removed for access by a PC.

- TIMERS :Programmable interval timer, time processing unit
- ANALOG TO DIGITAL CONVECTOR
- LCD OR KEYBOARD
- APPLICATION SPECIFIC ICS
- INTERUPPT CONTROLLER

Random-access memory (RAM)

- **Random-access memory** (**RAM**) is a form of computer data storage.
- Random access memory comes in two varieties
  1) static
  2)dynamic
- random-access memory takes the form of integrated circuits.

TYPES OF RAM

- Static ram
- In SRAM, a bit of data is stored
- **Static random-access memory** (**SRAM**) is a type of semiconductor memory that usesbistable latching circuitry to store each bit
- The term *static* differentiates it from *dynamic*RAM (DRAM) which must be periodically refreshed. (data is eventually lost when the memory is not powered.
- This form of RAM is more expensive to produce, but is generally faster and requires less power than DRAM and, in modern computers, is often used as cache memory for the CPU.
- Even when the disturbance , such as electrical noise , perturbs the voltages, the circuit will return to the stable value when the disturbance is removed

- Dynamic ram:
- DRAM stores a bit of data using a transistor and capacitor pair, which together comprise a memory cell.
- The capacitor holds a high or low charge (1 or 0, respectively), and the transistor acts as a switch that lets the control circuitry on the chip read the capacitor's state of charge or change it
- As this form of memory is less expensive to produce than static RAM
- Both static and dynamic RAM are considered *volatile*, as their state is lost or reset when power is removed from the system

rom

- **Read-only memory** (**ROM**) is a class of storage medium used in computersand other electronic devices
- Data stored in ROM cannot be modified, or can be modified only slowly or with difficulty, so it is mainly used to distributefirmware (software that is very closely tied to specific hardware, and unlikely to need frequent updates).
- ***ROM*** refers which is fabricated with the desired data permanently stored in it, and thus can never be modified. Despite the simplicity, speed ,field-programmability often make reprogrammable memories more flexible and inexpensive

Types of rom

- Programmable rom:
- **Programmable read-only memory** (PROM), or **one-time programmable ROM** (OTP), can be written to or programmed via a special device called a **PROM programmer**
- a PROM can only be programmed once.

Erasable programmed ROM(EPROM)

- An EPROM is a ROM that can be erased and reprogrammed.
- IT needs higher than usual voltage applied
- A littlie glass window is installed in the top of the ROM PACAKAGE ,through which you can actually see the chip that hold the memory.
- Ultraviolet light of a specific frequency can be shined through this window for a specified period of time, which will erase the EPROM and allow it to be reprogrammed again.

Electrically Erasable programmed ROM(EEPROM)

- (EEPROM or Flash ROM) are sometimes referred to, in an abbreviated way, as "read-only memory" (ROM); although these types of memory can be erased and re-programmed multiple times, writing to this memory takes longer and may require different procedures than reading the memory

I/O MAPPED I/O

- The main advantage of using I/O mapped I/O is on CPU with a limited addressing capability
- Because I/O Mapped I/O separates I/O access for memory access
- I/O operation can slow the memory access if the address and data bus buses are shared this is because the peripheral device is usually much slower than main memory. To overcome the extra complexity of I/O can bring by reduced instruction set computing which makes CPU acquire cheaper , faster , easier to build , consumes less power etc.

- **Memory-mapped I/O** (**MMIO**) and **port-mapped I/O** (**PMIO**) (which is also called *isolated I/O*) are two complementary methods of performing input/output between the CPU and peripheral devices in a computer. An alternative approach is using dedicated I/O processors—commonly known as channels on mainframe computers—that execute their own instructions.