Embedded System

Dr. Munesh Singh

Indian Institute of Information Technology Design and Manufacturing, Kancheepuram, Chennai, Tamil-Nadu 600127

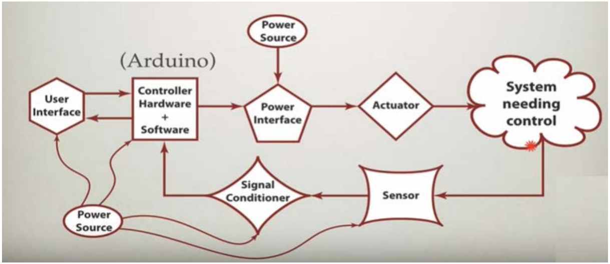
January 5, 2020

Table of Contents

Embedded System

What is Embedded System ?

- An embedded system is a special computer system designed to perform one or a few dedicated functions.
- Embedded system is a system where a microcontroller based or microprocessor based programmable system is embedded in a larger system.



History of Embedded System

- One of the first recognizable modern embedded system appear in 1961 developed by Charles Stark Draper at MIT instrumentation laboratory.
- It was designed for Appollo Guidance
- The first consumer-oriented microprocessor was introduced by Intel 4004 (for calculator and small system)
- In 1971-1972 8 bit microprocessor was born, but in general still external memory chip.
- In mid 1980's higher level of integration introduce a complete microprocessors.

Characteristics of Embedded System

Special Purpose

- Typically is designed to execute a single program, repeatedly
- It used to be single purpose
- Now, multi-functional, but single-purpose

Tightly Constrained

- Low cost, simple system, fewer component, perform function fast enough, minimum power

Characteristics of Embedded System

Reacive and Realtime

- Reactive: Continually react to external events
- Realtime: Must compute result in realtime

Hardware & Software

- The software written for embedded system often called firmware
- Is stored in read-only memory or flash memory chip rather than a disk drive

Classification of Embedded System

- Based on functionality and performance requirements, embedded systems are classified as:
  1. Stand alone embedded system
  2. Real-time embedded system
     - Hard RTS (Stringent time constraint)
     - Soft RTS (Stringent time constraint is not required)
  3. Network informational applications
     - Internet of Things
     - Cyber Physical System
  4. Mobile Device

Embedded System vs General Purpose System

Embedded System

1. Single purpose or single application
2. Tightly Constrained
   - Low cost
   - Low power
   - Portable
   - Sometime real time

General Purpose System

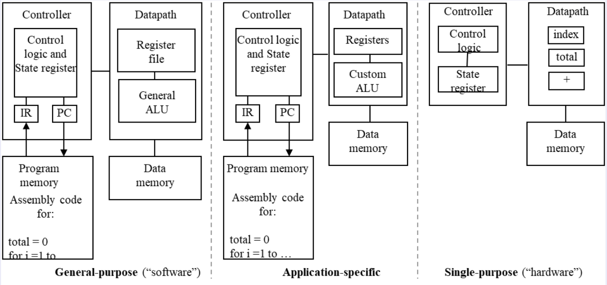
1. Multipurpose or Multitasking
2. Designer can decide
   - No of IO port
   - Memory
   - Costlier
   - Heavy
   - Power inefficient

Types of Embedded Processor

Processors

1. General Purpose Processor (GPP)
2. Application Specific Instruction Processor (ASIP)
3. Application Specific Integrated Circuit (ASIC)

Architecture of Processor



Architecture of Processor

Von-Neumann Architecture

- Single memory to be shared by both code and data
- Processor needs to fetch code in a separate clock cycle and data in another clock cycle.
- Higher speed, thus less time consuming
- Simple in design

Harvard Architecture

- Separate memories for code and data.
- Single clock cycle is sufficient, as separate buses are used to access code and data
- Slower in speed, thus more time-consuming
- Complex in design

How Processor Differ from each other

- An instruction set, or instruction set architecture (ISA), is the part of the computer architecture related to programming
  - It includes the native data types
  - instructions
  - registers
  - addressing modes
  - memory architecture
  - interrupt and exception handling
  - external I/O
- ISA define the complexity of the processor
- ISA differ in every processor
- ISA classified into two categorizes:
  - Reduce instruction set Computer (RISC)
  - Complex instruction set Computer (CISC)

RISC vs CISC

CISC

- Larger set of information,
- Easy to program
- Simpler design of compiler, considering larger set of instructions.
- Many addressing mode causing complex instruction formats
- Instruction length is variable.
- Higher clock cycle per second
- Pipelining is not possible

RISC

- Smaller set of instructions.
- Difficult to program
- Complex design of compiler
- Few addressing mode, fixed instruction formate
- Instruction length varies
- Low clock cycle per second
- Faster execution, as each instruction to be executed by hardware
- Pipelining is possible

Building an Embedded System

- We embed 3 basic kinds of computing engines into our systems:
  - microprocessor
  - microcomputer
  - microcontrollers.
- These computing engines are connected to other hardware via a system bus
- The system bus is further classified into address ,data and control bus
  - data bus to carry information
  - address bus to determine where it should be sent
  - control bus to determine its operation
- The microprocessor controls the whole system by executing a set of instructions call firmware that is stored in ROM

Building Blocks of Embedded System

Element of Embedded System

Connectivity

- USB
- RJ45/Ethernet

Computation

- CPU
- Timer
- Interrupt Controller
- Oscillator circuit

Audio/Video Interfaces

- HDMI
- 3.5 mm Audio
- RCA Audio

I/O Interfaces for Sensors,Acuator,etc

- UART
- SPI
- I2C
- CAN

Memory Interfaces

- NAND/NOR
- DDR1/DDR2/DDR3

Graphics

- GPU

Storage Interfaces

- MMC
- SD Card
- SDIO

Embedded Communication Interfaces

General Purpose Input/Output Port (GPIO)

1. Digital Ports & Analog Ports
2. Serial Peripheral Interfacing Ports (SPI), ESSI,JTAG
3. Inter-Integrated Communication (I2C)
4. Pulse Width Modulation (PWM)
5. Universal Asynchronous Receiver/Transmitter
6. Interrupt Port

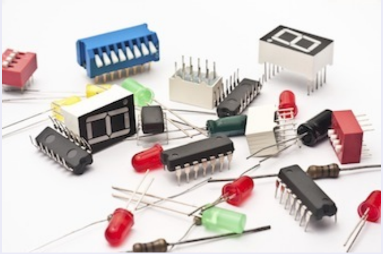
Pheripheral Interfacing Ports

1. Universal Serial Bus (USB), RS232
2. High Definition Multimedia Interface (HDMI), Audio, Camera
3. SD card Interface
4. Ethernet Port (RJ45)

GPIO Digital Ports

Digital I/O ports use

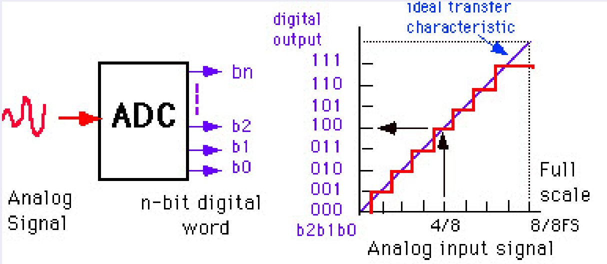
- All GPIO pins have true Read-Modify-Write ability
- The direction of one port pin can be changed independently of the other pins
- Can be used as Digital input or output



GPIO Analog Ports

Analog I/O ports use

- Analog I/O Ports are interfaced with A/D and D/A converter
- Analog I/O used for interfacing sensors with analog I/O.

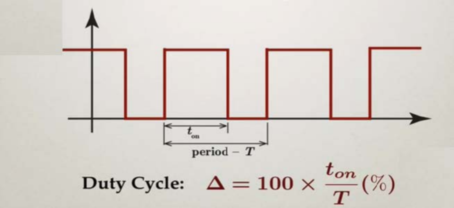


GPIO PWM Ports

PWM I/O ports use

- PWM is defined in terms of its period and its duty cycle
- PWM is used to simulate the analog output.
- Equation of PWM:

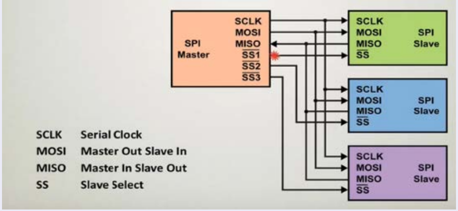
$$Output\ voltage = \frac{T_{on}}{T_{on+off}} * 5v \tag{1}$$



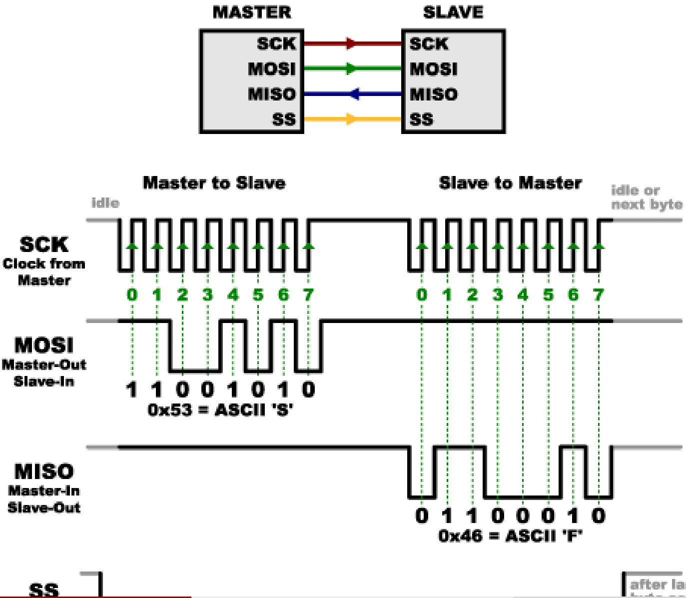
GPIO SPI Ports

SPI I/O ports use

- Industry standard serial protocol for communication between local devices
- It is a synchronous Master/Slave communication protocol
- 4 wire interface
- SS line is used to selected specific slave device



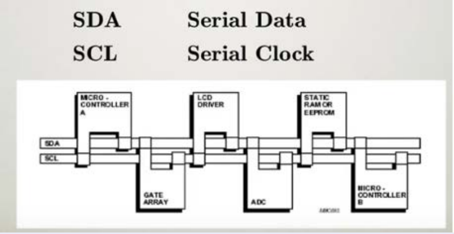
Working of SPI Communication



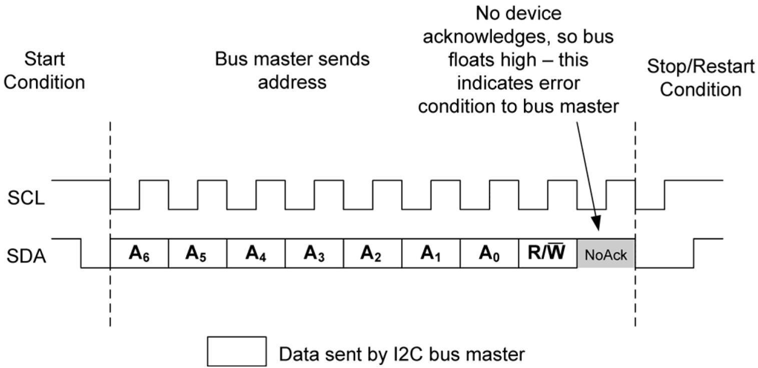
GPIO I2C Ports

I2C I/O ports use

- It is a synchronous Master/Slave communication protocol
- 2 wire interface
- Slaves addressed embedded in command
- Byte oriented message



Working of I2C Communication



GPIO Interrupt Ports

Interrupt I/O ports use

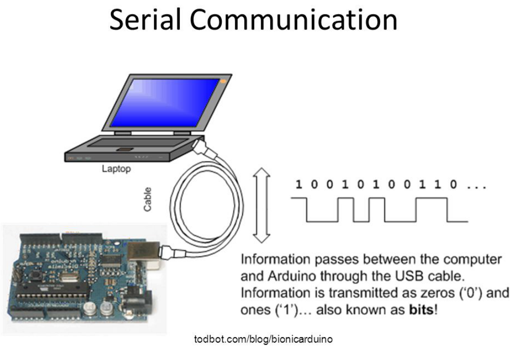
- Allow program to respond to events when they occur
- Allow program to ignore event until they occur
- External events
  - UART ready with/for next character
  - Signal change on pin

Use of Interrupt

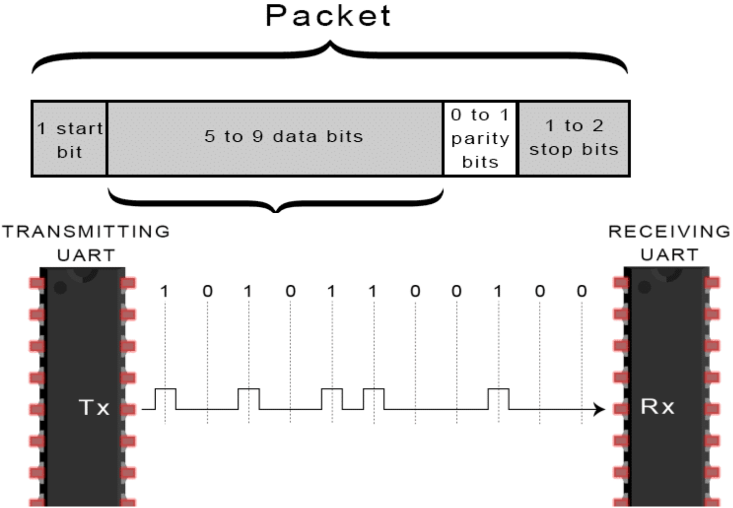
- To detect pin chnage (e.g rotary encoders, button presses)
- Watchdog timer (eg. if nothing happens after 8 seconds,interrupt me)
- Timer Interrupt-Used for comparing/over using timers SPI, I2C, UART data transfer, ADC conversion, EPROM ready signal.

UART GPIO Port

- UARTs transmit data asynchronously
- UART adds start and stop bits to the data packet being transferred
- These bits define the beginning and end of the data packet



Working of UART Communication



Thank You