I2C PROTOCAL:

I2C stands for Inter-Integrated circuit. It is a bus interface connection protocal incorporated into device for serial communication It was originally designed by phillips semiconductor in 1982. Recently, It is a widely used protocal for short distance communication. It is also known as two wired interface (TWI)

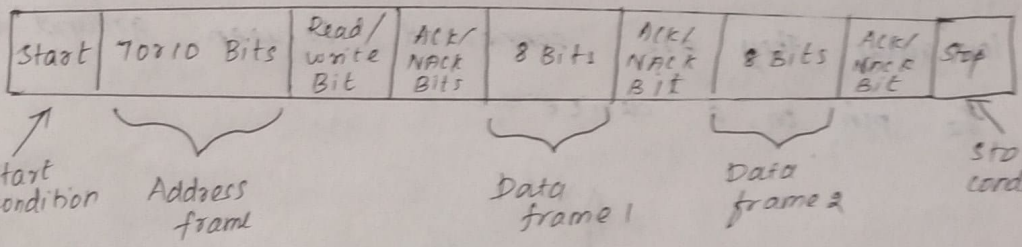
Working:

It uses only 2 bidirectional open drain lines for data communication called SDA and SCL. Both these lines are pulled high.

Serial Data (SDA) - Transfer of data takes place through this pin.

Serial clock (SCL) - It carries the clock signal

I2C operates in two modes - Master mode and slave mode



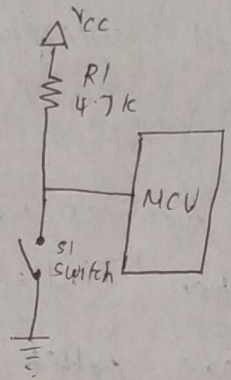
It operates a single master single slave
single master multi slave
Multi master multi slave

It is a synchronous, half duplex and serial communication protocal.

If it has 7 bit address the 2^7 - 127 slaves are connected and for 10 bit address 2^{10} - 1024 slaves are connected.

Pull up resistor

A pull up resistor is a resistor used to ensure a known state for a signal. The microcontroller might unpredictably interpret the input value as either a logical high or logical low. pull up resistors are used to solve the dilema of the microcontroller by pulling the value to logical high state.

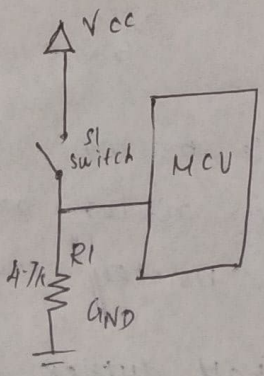


Pull up resistor circuit

Pull up resistors worked to used to ensure that a wire is pulled to high logical level in the absence of a input signal.

pull down resistor:

pull down resistor pull the pin to a logical low value. They are connected between ground and the pin on a device.



pull down resistor

open drain:

An open drain or open collector output pin is driven by single transistor, which pulls the pin to only one voltage (generally to ground). when the output device is off, the pin is left floating (open)

I2C is multimaster and multislave protocal. If the lines are not open drain there may be chance that one slave pulling the device low and other slave is trying to pull the same line high. This situation is called bus contention

active low:

Active low means the function get done when input is in low state. A signal is active low means that signal will be performing its function when its logical level 0. Active low always help to eliminate indeterminant state due to improper supply voltage.

active high:

Active high means function get done when the input is high state. A signal is active high means that signal will be peforming its function when its state is logical level 1.

Linux Booting process:

A Linux boot process is the initialisation of the linux open source operating system on a computer. Also known as the linux startup process, a Linux boot process cover a number of steps from initial bootstrap to launch of the initial user-space application.

The following are the 6 high level stages of typical linux boot process

BIOS - Basic input/output system executes MBR

MBR - Master Boot Record executed GRUB

GRUB - Grand Unified Bootloader executes kernal

kernel - kernal executes /sbin/init

Init - Init executes runlevel programs

Runlevel - Runlevel programs are executed from /etc/rc.d/rc*.d/

BIOS:

It performs some system integrity checks. Searches, loads and executes the boot loader program. It looks for boot loader in floppy, cd-rom or hard drive. you can press a key (typically F12 or F2, but it depends on your system) during the BIOS startup do exchange the boot sequence. Once the boot loader program is detected and loaded into the memory. BIOS gives the control on it. So in simple terms BIOS loads and executes the MBR boot loader.

MBR:

MBR stands for Master boot record. It is located in the 1st sector of the bootable disk. Typically /dev/hda, or /dev/sda. MBR is less than 512 bytes in size. This has three components.

1) primary boot loader info in 1st 446 bytes
2) partition table info in next 64 bytes
3) Mbr validation check last 2 bytes.

It contains information about GRUB (or LILO in old system) so it in simple terms MBR loads and executes the GRUB boot loader.

GRUB:

If you have multiple kernal images installed on your system. you can choose which one to be executed. GRUB displays a splash screen, waits for few seconds. If you don't enter anything. It loads the default kernal image as specified in the grub configuration file. In simple words or term GRUB just loads and executes Kernel initrd images.

Kernal:

Mounts the root file system as specified in the grub. Kernal executes the /sbin/init program. since init was the 1st program to be executed by linux kernal, it has the process id (PID) of 1. Do a ps -ef /grep init and check the pid. Initrd stands for initial RAM Disk. Initrd is used by kernal as temperory root file system until the kernal is booted and the real file system is mounted. It also contains necessary drives compiled inside, which

which help it to access the hard drive partition and other hardware

Init:

looks at the /etc/inittab file to decide the linux run level. The following are the available run levels

0- halt
1- single user mode
2- Multiuser, without NFS
3- Full multiuser mode
4- Unused
5- X11
6- reboot

Init identifies the default init levels from /etc/inittab and uses that to load all appropriate program. If you want to get into trouble you can set the default run level 0 or 6. Since you know what 0 and 6 means, probably you might not do that. Typically you should set the default run level either 3 or 5.

Runlevel programs:

When the linux system is booting up, you might see various service getting started. Depending on your default init level setting, the system will execute the program from one of the following directories.

* Run level 0 - /etc/rc.d/rc0.d/
* Run level 1 - /etc/rc.d/rc1.d/
* Run level 2 - /etc/rc.d/rc2.d/
* Run level 3 - /etc/rc.d/rc3.d/
* Run level 4 - /etc/rc.d/rc4.d/
* Run level 5 - /etc/rc.d/rc5.d/
* Run level 6 - /etc/rc.d/rc6.d/

Under the /etc/rc.d/rc*.d/ directories you would see programs that start with s and k. programs starts with S are used during startup S for Startup programs starts with K are used to during shutdown K for kill. There are numbers right next to s and k in the program names. There are the sequence number in which the programs should be started or killed.

Role of kernal:

The linux kernal is the main component of a linux operating system (OS) and is the core interface between a computer's hardware and its process. The kernal is so named because like a seed inside a hard shell - it exist with the OS and controls the major functions of the hardware, whether its a phone, laptop, server or any kind of computer.

Kernal has four main jobs they are

Memory management - keep track of how much memory is used to store what and where

process management - Determine which processes can be use the central processing unit (CPU) when and for how long

Device drivers: Act as mediator / interpreter between hardware and processes

System calls and security - Receive request for service from the processes.

Zephyr RTOS:

Zephyr is a small real time operating system (RTOS) for connected, resource constrained and embedded device (with an emphasis on microcontroller) supporting multiple architectures and relesed under the Apache License 2.0. Zephyr includes a kernel and all components and libraries, device drivers, protocal stack, file system and firmware updates.

It is originated from virtuoso RTOS for digital signal processors. It has small kernel. A flexible configuration and build system for compile-time definition of required resources and modules. A virtual file system interface with several flash file system for non volatile storage (FATFS, LittleFS, NVS). Management and device firmware update mechanism.

The kernel has several feautes. It has

- single address space
- Multiple scheduling algorithms
- Highly configurable and modular for flexibility with resources defined at compile time.
- Memory protection unit (MPU) based protection.
- Asymmetric multiprocessing (AMP, based on openAMP) and symmetric multiprocessing (SMP) support.