Gnu Tools

make command

```
make
```

- Output

```
murthu@desd:~/desd/embedded linux/Assignments/Day-2$ make
gcc -ggdb -c file.c
gcc -ggdb -c main.c
gcc -ggdb main.o file.o -o main
```

- using arm-linux-gnueabi-gcc compiler

```
murthu@desd:~/desd/embedded linux/Assignments/Day-2$ make CC=arm-linux-gnueabi-gcc
arm-linux-gnueabi-gcc -ggdb -c file.c
arm-linux-gnueabi-gcc -ggdb -c main.c
arm-linux-gnueabi-gcc -ggdb main.o file.o -o main
```

- using cortex-A53 cpu (raspberrypi 3b+)

```
murthu@desd:~/desd/embedded linux/Assignments/Day-2$ make CC=arm-linux-gnueabi-gcc
arm-linux-gnueabi-gcc -mtune="cortex-a53" -ggdb -c file.c
arm-linux-gnueabi-gcc -mtune="cortex-a53" -ggdb -c main.c
arm-linux-gnueabi-gcc -mtune="cortex-a53" -ggdb main.o file.o -o main
```

objdump command

```
arm-linux-gnueabi-objdump --architecture=cortex-a53 -S file.o
```

- Description
  - -S --> Display source code intermixed with disassembly, if possible
- Sample output for cortex a-53 processor

```
file.o:     file format elf32-littlearm


Disassembly of section .text:

00000000 <read_file>:
#include "file.h"

#define SIZE 1024

char* read_file(FILE* file_path)
{
   0:   e92d4800        push    {fp, lr}
   4:   e28db004        add     fp, sp, #4
   8:   e24dd018        sub     sp, sp, #24
   c:   e50b0018        str     r0, [fp, #-24]  ; 0xffffffe8
  10:   e59f3078        ldr     r3, [pc, #120]  ; 90 <read_file+0x90>
  14:   e5933000        ldr     r3, [r3]
  18:   e50b3008        str     r3, [fp, #-8]
  1c:   e3a03000        mov     r3, #0
    size_t count=0;
  20:   e3a03000        mov     r3, #0
  24:   e50b3014        str     r3, [fp, #-20]  ; 0xffffffec
    char* buffer=NULL;
  28:   e3a03000        mov     r3, #0
  2c:   e50b3010        str     r3, [fp, #-16]
    ssize_t nread=0;
  30:   e3a03000        mov     r3, #0
  34:   e50b300c        str     r3, [fp, #-12]

    //allocates size dynamically if buffer is null and count is 0
    nread=getline(&buffer,&count,file_path);
  38:   e24b1014        sub     r1, fp, #20
  3c:   e24b3010        sub     r3, fp, #16
```

ar command

```
ar -rc <library_name>.a .ofiles
ar -rc libfileio.a file.o
```

r-->inserts the file with replacement

c-->if archive file is not there it creates a new one

```
gcc -ggdb main.c -lfileio -L. -o main
```

- using libfileio.a to generate binary.

```
gcc -fPIC -shared libfile.so file.o
```

- To create binary from shared object file use below command

```
gcc main.c -lfileio -L"/home/murthu/desd/embedded linux/Assignments/files" -o
main
```