1. Define cross-toolchain and explore what is inside a cross-compiling toolchain.

Cross-Toolchain Definition:

- A set of tools for cross-compiling software.
- Enables building for a target architecture different from the host.

Components Inside a Cross-Compiling Toolchain:

- **Cross-Compiler**: Generates code for the target architecture.
- **Linker**: Connects and creates the final executable.
- **Assembler**: Translates assembly code to machine code.
- **Standard C Library (libc**): Adapted for the target system.
- **Header Files:** Target-specific declarations and definitions.
- **Runtime Libraries:** Support for executing programs.
- **Loader**: Loads and executes programs on the target.
- **Debugging Tools:** (e.g., GDB) for target-specific debugging.

2. Read the datasheet, user manual/ getting started manual specific to your Embedded Linux-board. Write a 4 page summary based on your understanding about board details such as CPU architecture & type, memory types supported by the board, on-board buses, booting sequence, peripherals supported, various booting options and any other relevant information.

CPU Architecture & Type:

- CPU: Texas Instruments Sitara AM3358/9 ARM Cortex-A8 processor.
- Architecture: ARMv7-A architecture.
- Memory Types Supported:
- RAM: 512MB DDR3.
- Storage: 4GB 8-bit eMMC on-board flash storage.
- MicroSD: External storage option.
- On-board Buses:
- GPIO: General Purpose Input/Output pins.
- I2C, SPI, UART: Various communication interfaces for connecting peripherals.

Booting Sequence:

- The BeagleBone Black can boot from different sources, including the on-board eMMC, microSD card, or USB.

Peripherals Supported:

- Ethernet: 10/100 Ethernet.
- USB: USB 2.0 ports for connecting peripherals.
- HDMI: HDMI interface for video output.
- Audio: Stereo output/input.
- CAN: Controller Area Network interface.
- PWM: Pulse Width Modulation for controlling motors and other devices.

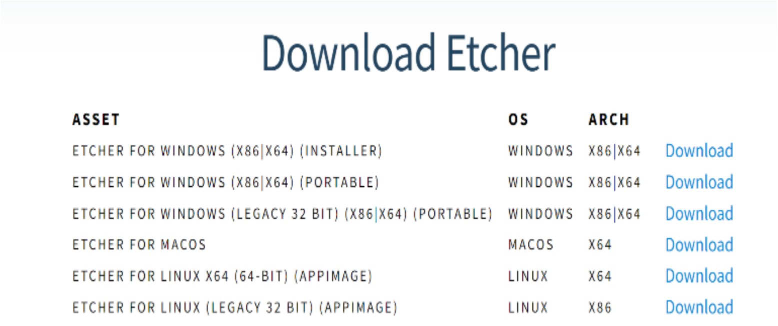
Various Booting Options:

- eMMC: Booting from the on-board eMMC storage.
- microSD: Booting from a microSD card.
- USB: Booting from a USB device.

3. Boot up an Embedded Linux Board, login into the system and grab a shell. Try basic Linux commands on the shell.

   1. Download Debian Beaglebone Black Image File:- https://www.beagleboard.org/distros/am335x-12-2-2023-10-07-4gb-microsd-iot

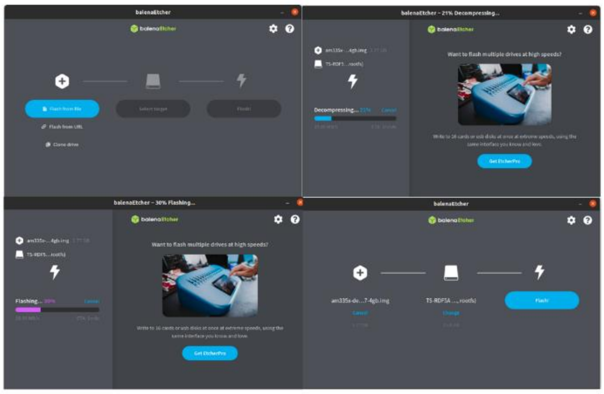
   2. Download Balena etcher for flashing the Image file SD card: https://etcher.balena.io/#download-etcher



Download Etcher

ASSET	OS	ARCH	
ETCHER FOR WINDOWS (X86|X64) (INSTALLER)	WINDOWS	X86|X64	Download
ETCHER FOR WINDOWS (X86|X64) (PORTABLE)	WINDOWS	X86|X64	Download
ETCHER FOR WINDOWS (LEGACY 32 BIT) (X86|X64) (PORTABLE)	WINDOWS	X86|X64	Download
ETCHER FOR MACOS	MACOS	X64	Download
ETCHER FOR LINUX X64 (64-BIT) (APPIMAGE)	LINUX	X64	Download
ETCHER FOR LINUX (LEGACY 32 BIT) (APPIMAGE)	LINUX	X86	Download

   3. Install the balena etcher and transfer the image to SD card



balenaEtcher

balenaEtcher – 21% Decompressing...

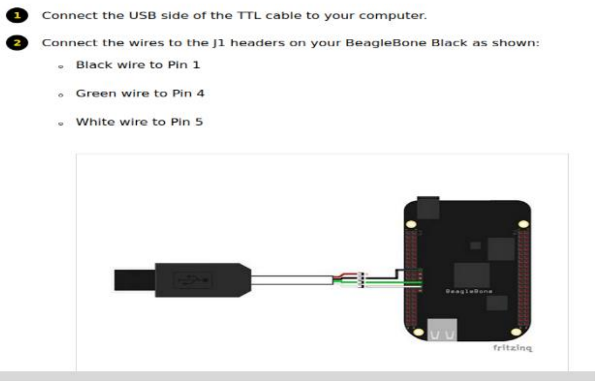
balenaEtcher – 30% Flashing...

balenaEtcher

Want to flash multiple drives at high speeds?

Write to 16 cards or usb disks at once at extreme speeds, using the same interface you know and love.

4. Complete serial connection on the beagleboneboard

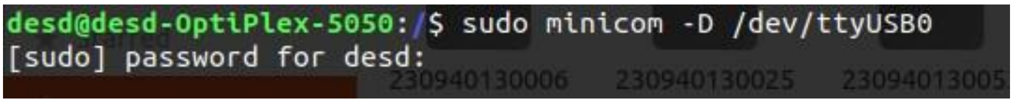


1. Connect the USB side of the TTL cable to your computer.
2. Connect the wires to the J1 headers on your BeagleBone Black as shown:
   - Black wire to Pin 1
   - Green wire to Pin 4
   - White wire to Pin 5

5. Install minicom:

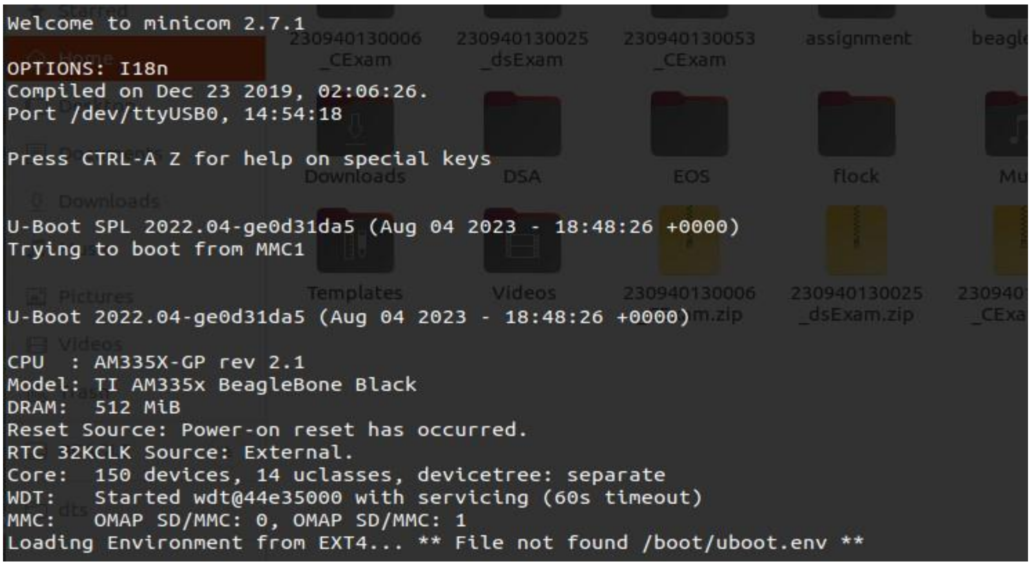
   $ sudo apt-get install minicom

6. **$ sudo minicom -D /dev/ttyUSB0**



```
desd@desd-OptiPlex-5050:/$ sudo minicom -D /dev/ttyUSB0
[sudo] password for desd:
```

Place the card inside the BBB; provide power supply to board



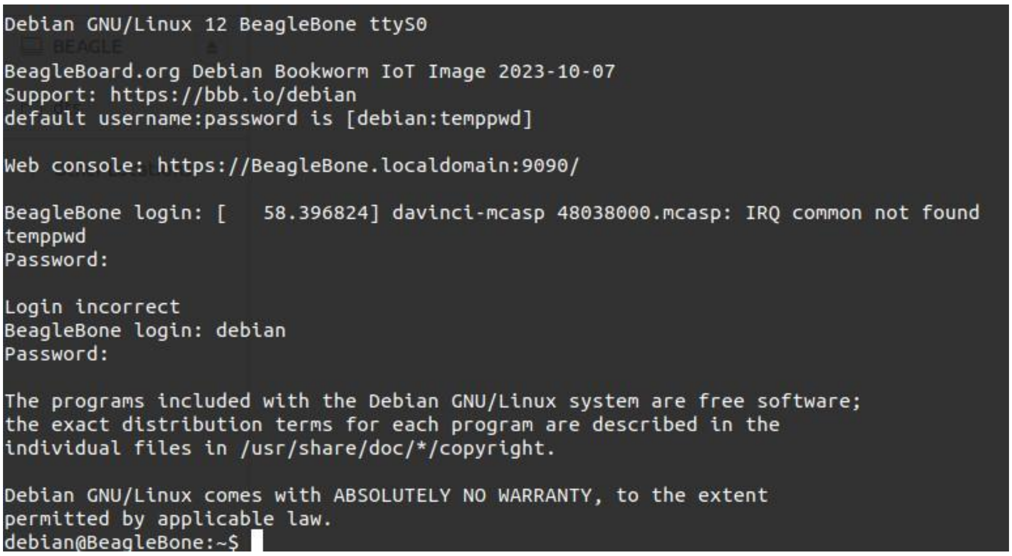
```
Welcome to minicom 2.7.1

OPTIONS: I18n
Compiled on Dec 23 2019, 02:06:26.
Port /dev/ttyUSB0, 14:54:18

Press CTRL-A Z for help on special keys


U-Boot SPL 2022.04-ge0d31da5 (Aug 04 2023 - 18:48:26 +0000)
Trying to boot from MMC1


U-Boot 2022.04-ge0d31da5 (Aug 04 2023 - 18:48:26 +0000)

CPU  : AM335X-GP rev 2.1
Model: TI AM335x BeagleBone Black
DRAM:  512 MiB
Reset Source: Power-on reset has occurred.
RTC 32KCLK Source: External.
Core:  150 devices, 14 uclasses, devicetree: separate
WDT:   Started wdt@44e35000 with servicing (60s timeout)
MMC:   OMAP SD/MMC: 0, OMAP SD/MMC: 1
Loading Environment from EXT4... ** File not found /boot/uboot.env **
```

7. Check device details

   $ ifconfig | more

8. Login to BBB board

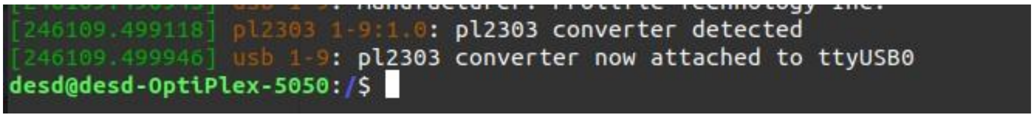
   $ ssh debian@192.168.7.2



```
Debian GNU/Linux 12 BeagleBone ttyS0

BeagleBoard.org Debian Bookworm IoT Image 2023-10-07
Support: https://bbb.io/debian
default username:password is [debian:temppwd]

Web console: https://BeagleBone.localdomain:9090/

BeagleBone login: [   58.396824] davinci-mcasp 48038000.mcasp: IRQ common not found
temppwd
Password:

Login incorrect
BeagleBone login: debian
Password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
debian@BeagleBone:~$
```

9. Checing and confirming device connection setup or not

   $ dmesg | more

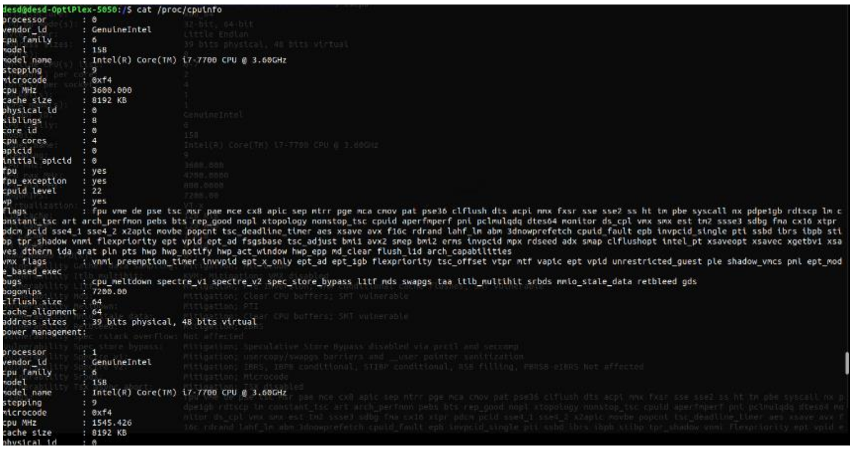


```
[246109.499118] pl2303 1-9:1.0: pl2303 converter detected
[246109.499946] usb 1-9: pl2303 converter now attached to ttyUSB0
desd@desd-OptiPlex-5050:/$
```

4. Use the shell commands to find details about CPU, Memory capacity, Memory and I/O maps, persistent storage details.

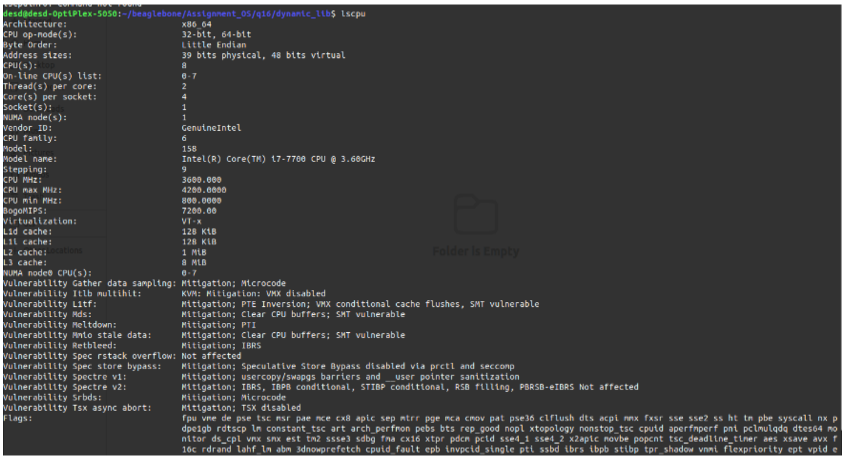
1. CPU Details:

- **$ cat /proc/cpuinfo**
  - ➢ Displays detailed information about the CPU, including model, architecture, and clock speed etc.



```
desd@desd-OptiPlex-5050:/$ cat /proc/cpuinfo
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 158
model name      : Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz
stepping        : 9
microcode       : 0xf4
cpu MHz         : 3600.000
cache size      : 8192 KB
physical id     : 0
siblings        : 8
core id         : 0
cpu cores       : 4
apicid          : 0
initial apicid  : 0
fpu             : yes
fpu_exception   : yes
cpuid level     : 22
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm c
onstant_tsc art arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc cpuid aperfmperf pni pclmulqdq dtes64 monitor ds_cpl vmx smx est tm2 ssse3 sdbg fma cx16 xtpr
pdcm pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer aes xsave avx f16c rdrand lahf_lm abm 3dnowprefetch cpuid_fault epb invpcid_single pti ssbd ibrs ibpb sti
bp tpr_shadow vnmi flexpriority ept vpid ept_ad fsgsbase tsc_adjust bmi1 avx2 smep bmi2 erms invpcid mpx rdseed adx smap clflushopt intel_pt xsaveopt xsavec xgetbv1 xsa
ves dtherm ida arat pln pts hwp hwp_notify hwp_act_window hwp_epp md_clear flush_l1d arch_capabilities
vmx flags       : vnmi preemption_timer invvpid ept_x_only ept_ad ept_1gb flexpriority tsc_offset vtpr mtf vapic ept vpid unrestricted_guest ple shadow_vmcs pml ept_mod
e_based_exec
bugs            : cpu_meltdown spectre_v1 spectre_v2 spec_store_bypass l1tf mds swapgs taa itlb_multihit srbds mmio_stale_data retbleed gds
bogomips        : 7200.00
clflush size    : 64
cache_alignment : 64
address sizes   : 39 bits physical, 48 bits virtual
power management:

processor       : 1
vendor_id       : GenuineIntel
cpu family      : 6
model           : 158
model name      : Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz
stepping        : 9
microcode       : 0xf4
cpu MHz         : 1545.426
cache size      : 8192 KB
physical id     : 0
```

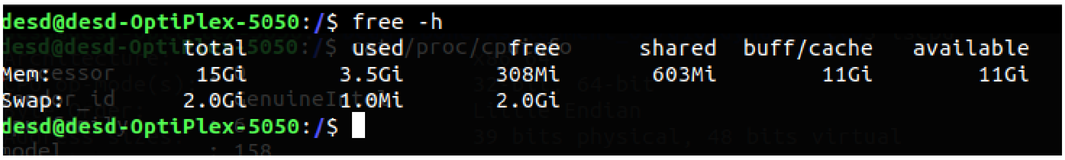
- **$ lscpu**
  - ➢ Provides detailed information about the CPU, including its architecture, number of cores, threads per core, and other relevant details in a concise format.



```
desd@desd-OptiPlex-5050:~/beaglebone/Assignment_OS/q16/dynamic_lib$ lscpu
Architecture:                    x86_64
CPU op-mode(s):                  32-bit, 64-bit
Byte Order:                      Little Endian
Address sizes:                   39 bits physical, 48 bits virtual
CPU(s):                          8
On-line CPU(s) list:             0-7
Thread(s) per core:              2
Core(s) per socket:              4
Socket(s):                       1
NUMA node(s):                    1
Vendor ID:                       GenuineIntel
CPU family:                      6
Model:                           158
Model name:                      Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz
Stepping:                        9
CPU MHz:                         3600.000
CPU max MHz:                     4200.0000
CPU min MHz:                     800.0000
BogoMIPS:                        7200.00
Virtualization:                  VT-x
L1d cache:                       128 KiB
L1i cache:                       128 KiB
L2 cache:                        1 MiB
L3 cache:                        8 MiB
NUMA node0 CPU(s):               0-7
Vulnerability Gather data sampling: Mitigation; Microcode
Vulnerability Itlb multihit:     KVM: Mitigation: VMX disabled
Vulnerability L1tf:              Mitigation; PTE Inversion; VMX conditional cache flushes, SMT vulnerable
Vulnerability Mds:               Mitigation; Clear CPU buffers; SMT vulnerable
Vulnerability Meltdown:          Mitigation; PTI
Vulnerability Mmio stale data:   Mitigation; Clear CPU buffers; SMT vulnerable
Vulnerability Retbleed:          Mitigation; IBRS
Vulnerability Spec rstack overflow: Not affected
Vulnerability Spec store bypass: Mitigation; Speculative Store Bypass disabled via prctl and seccomp
Vulnerability Spectre v1:        Mitigation; usercopy/swapgs barriers and __user pointer sanitization
Vulnerability Spectre v2:        Mitigation; IBRS, IBPB conditional, STIBP conditional, RSB filling, PBRSB-eIBRS Not affected
Vulnerability Srbds:             Mitigation; Microcode
Vulnerability Tsx async abort:   Mitigation; TSX disabled
Flags:                           fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx p
                                 dpe1gb rdtscp lm constant_tsc art arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc cpuid aperfmperf pni pclmulqdq dtes64 mo
                                 nitor ds_cpl vmx smx est tm2 ssse3 sdbg fma cx16 xtpr pdcm pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer aes xsave avx f
                                 16c rdrand lahf_lm abm 3dnowprefetch cpuid_fault epb invpcid_single pti ssbd ibrs ibpb stibp tpr_shadow vnmi flexpriority ept vpid e
```

2. Memory Capacity:

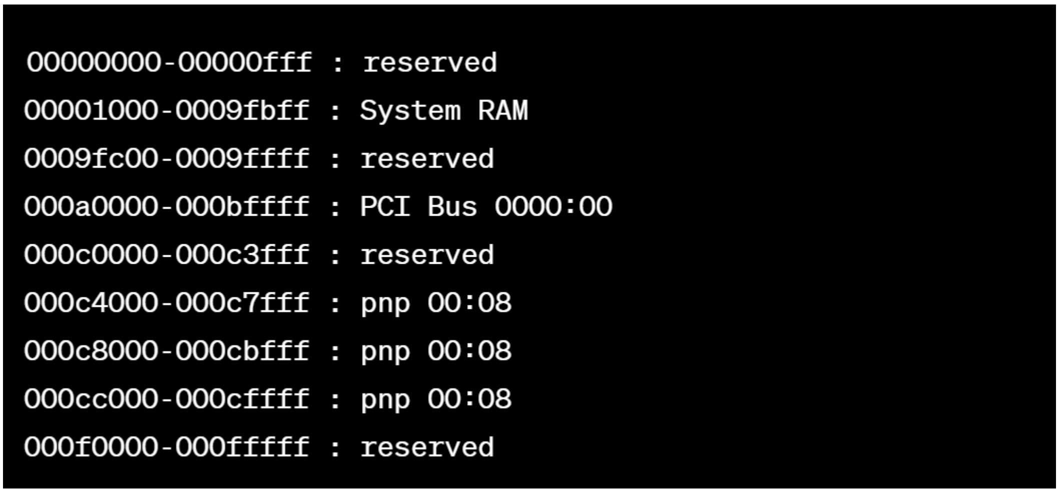
- **$ free -h**
  - ➢ Shows information about system memory usage and capacity in a human-readable format.



```
desd@desd-OptiPlex-5050:/$ free -h
               total        used        free      shared  buff/cache   available
Mem:            15Gi       3.5Gi       308Mi       603Mi        11Gi        11Gi
Swap:          2.0Gi       1.0Mi       2.0Gi
desd@desd-OptiPlex-5050:/$ 
```

3. Memory Map:

- **$ cat /proc/iomem**
  - ➢ Provides a list of current memory ranges used by different devices, showing the memory map of the system.

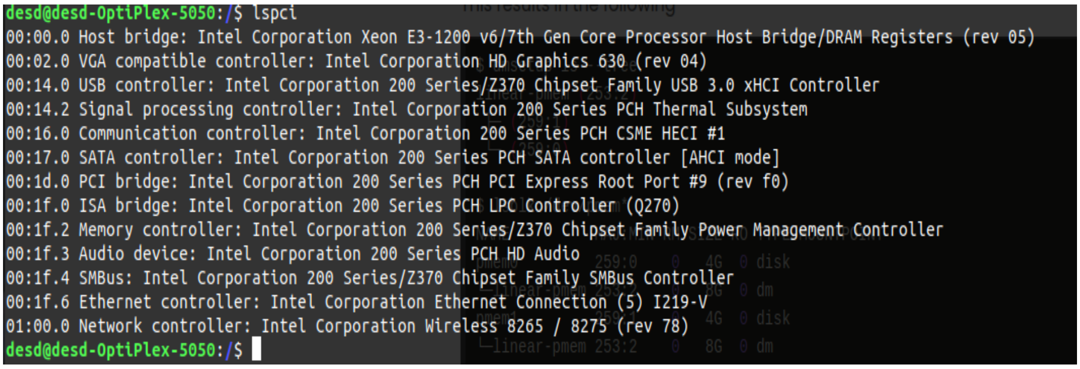


```
00000000-00000fff : reserved
00001000-0009fbff : System RAM
0009fc00-0009ffff : reserved
000a0000-000bffff : PCI Bus 0000:00
000c0000-000c3fff : reserved
000c4000-000c7fff : pnp 00:08
000c8000-000cbfff : pnp 00:08
000cc000-000cffff : pnp 00:08
000f0000-000fffff : reserved
```

4. I/O Map:

- **$ lspci**

  ➢ Lists all PCI buses and devices connected to them, providing information about I/O devices and their addresses.

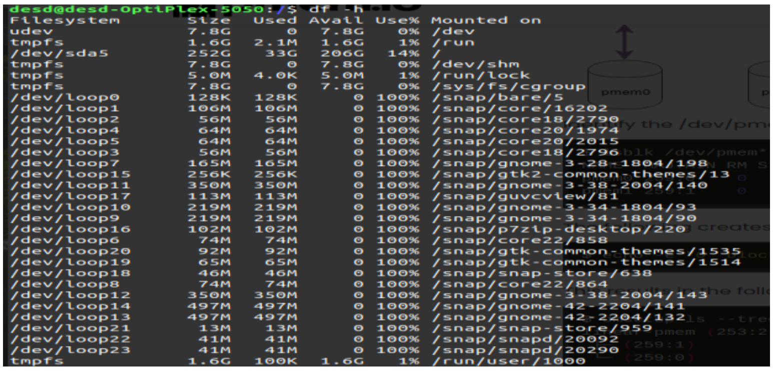


```
desd@desd-OptiPlex-5050:/$ lspci
00:00.0 Host bridge: Intel Corporation Xeon E3-1200 v6/7th Gen Core Processor Host Bridge/DRAM Registers (rev 05)
00:02.0 VGA compatible controller: Intel Corporation HD Graphics 630 (rev 04)
00:14.0 USB controller: Intel Corporation 200 Series/Z370 Chipset Family USB 3.0 xHCI Controller
00:14.2 Signal processing controller: Intel Corporation 200 Series PCH Thermal Subsystem
00:16.0 Communication controller: Intel Corporation 200 Series PCH CSME HECI #1
00:17.0 SATA controller: Intel Corporation 200 Series PCH SATA controller [AHCI mode]
00:1d.0 PCI bridge: Intel Corporation 200 Series PCH PCI Express Root Port #9 (rev f0)
00:1f.0 ISA bridge: Intel Corporation 200 Series PCH LPC Controller (Q270)
00:1f.2 Memory controller: Intel Corporation 200 Series/Z370 Chipset Family Power Management Controller
00:1f.3 Audio device: Intel Corporation 200 Series PCH HD Audio
00:1f.4 SMBus: Intel Corporation 200 Series/Z370 Chipset Family SMBus Controller
00:1f.6 Ethernet controller: Intel Corporation Ethernet Connection (5) I219-V
01:00.0 Network controller: Intel Corporation Wireless 8265 / 8275 (rev 78)
desd@desd-OptiPlex-5050:/$
```

5. Persistent Storage Details:

- **$ df -h**

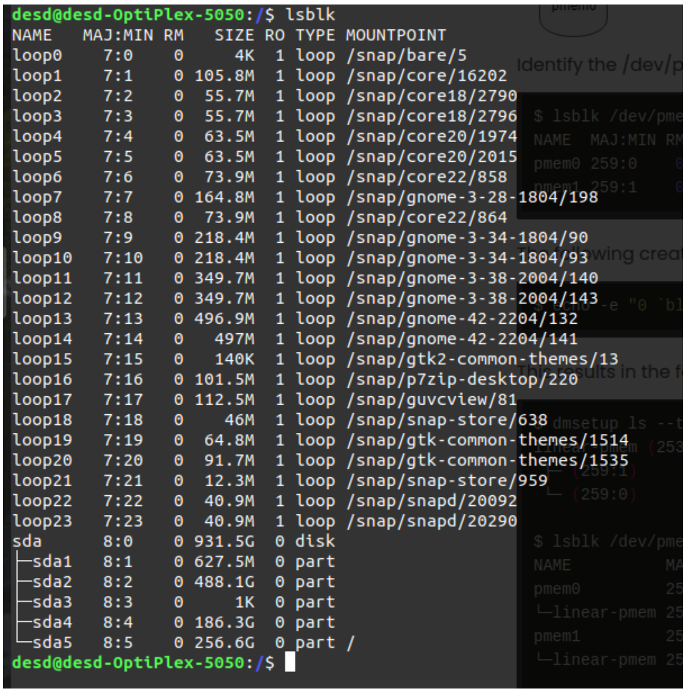
  ➢ Displays information about disk space usage, including details about mounted filesystems and their capacities.



```
desd@desd-OptiPlex-5050:/$ df -h
Filesystem       Size  Used Avail Use% Mounted on
udev             7.8G     0  7.8G   0% /dev
tmpfs            1.6G  2.1M  1.6G   1% /run
/dev/sda5        252G   33G  206G  14% /
tmpfs            7.8G     0  7.8G   0% /dev/shm
tmpfs            5.0M  4.0K  5.0M   1% /run/lock
tmpfs            7.8G     0  7.8G   0% /sys/fs/cgroup
/dev/loop0       128K  128K     0 100% /snap/bare/5
/dev/loop1       106M  106M     0 100% /snap/core/16202
/dev/loop2        56M   56M     0 100% /snap/core18/2790
/dev/loop4        64M   64M     0 100% /snap/core20/1974
/dev/loop5        64M   64M     0 100% /snap/core20/2015
/dev/loop3        56M   56M     0 100% /snap/core18/2796
/dev/loop7       165M  165M     0 100% /snap/gnome-3-28-1804/198
/dev/loop15      256K  256K     0 100% /snap/gtk2-common-themes/13
/dev/loop11      350M  350M     0 100% /snap/gnome-3-38-2004/140
/dev/loop17      113M  113M     0 100% /snap/guvcview/81
/dev/loop10      219M  219M     0 100% /snap/gnome-3-34-1804/93
/dev/loop9       219M  219M     0 100% /snap/gnome-3-34-1804/90
/dev/loop16      102M  102M     0 100% /snap/p7zip-desktop/220
/dev/loop6        74M   74M     0 100% /snap/core22/858
/dev/loop20       92M   92M     0 100% /snap/gtk-common-themes/1535
/dev/loop19       65M   65M     0 100% /snap/gtk-common-themes/1514
/dev/loop18       46M   46M     0 100% /snap/snap-store/638
/dev/loop8        74M   74M     0 100% /snap/core22/864
/dev/loop12      350M  350M     0 100% /snap/gnome-3-38-2004/143
/dev/loop14      497M  497M     0 100% /snap/gnome-42-2204/141
/dev/loop13      497M  497M     0 100% /snap/gnome-42-2204/132
/dev/loop21       13M   13M     0 100% /snap/snap-store/959
/dev/loop22       41M   41M     0 100% /snap/snapd/20092
/dev/loop23       41M   41M     0 100% /snap/snapd/20290
tmpfs            1.6G  100K  1.6G   1% /run/user/1000
```

- **$ lsblk**

  ➢ Lists information about block devices, including disks and partitions, providing details about their sizes and mount points.

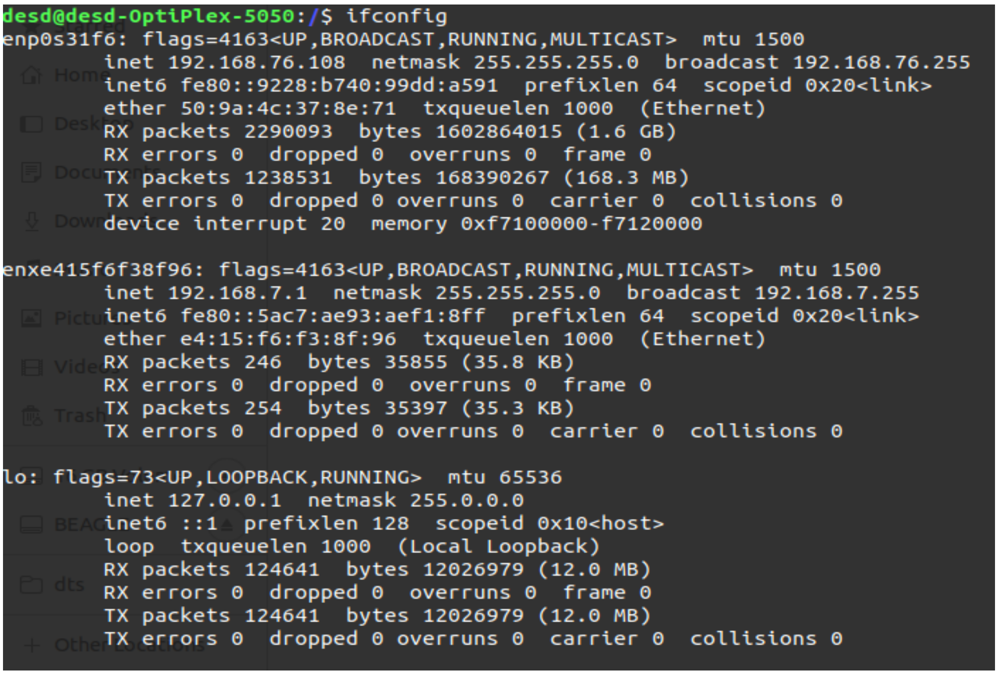


```
desd@desd-OptiPlex-5050:/$ lsblk
NAME    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
loop0     7:0    0     4K  1 loop /snap/bare/5
loop1     7:1    0 105.8M  1 loop /snap/core/16202
loop2     7:2    0  55.7M  1 loop /snap/core18/2790
loop3     7:3    0  55.7M  1 loop /snap/core18/2796
loop4     7:4    0  63.5M  1 loop /snap/core20/1974
loop5     7:5    0  63.5M  1 loop /snap/core20/2015
loop6     7:6    0  73.9M  1 loop /snap/core22/858
loop7     7:7    0 164.8M  1 loop /snap/gnome-3-28-1804/198
loop8     7:8    0  73.9M  1 loop /snap/core22/864
loop9     7:9    0 218.4M  1 loop /snap/gnome-3-34-1804/90
loop10    7:10   0 218.4M  1 loop /snap/gnome-3-34-1804/93
loop11    7:11   0 349.7M  1 loop /snap/gnome-3-38-2004/140
loop12    7:12   0 349.7M  1 loop /snap/gnome-3-38-2004/143
loop13    7:13   0 496.9M  1 loop /snap/gnome-42-2204/132
loop14    7:14   0   497M  1 loop /snap/gnome-42-2204/141
loop15    7:15   0   140K  1 loop /snap/gtk2-common-themes/13
loop16    7:16   0 101.5M  1 loop /snap/p7zip-desktop/220
loop17    7:17   0 112.5M  1 loop /snap/guvcview/81
loop18    7:18   0    46M  1 loop /snap/snap-store/638
loop19    7:19   0  64.8M  1 loop /snap/gtk-common-themes/1514
loop20    7:20   0  91.7M  1 loop /snap/gtk-common-themes/1535
loop21    7:21   0  12.3M  1 loop /snap/snap-store/959
loop22    7:22   0  40.9M  1 loop /snap/snapd/20092
loop23    7:23   0  40.9M  1 loop /snap/snapd/20290
sda       8:0    0 931.5G  0 disk
├─sda1    8:1    0 627.5M  0 part
├─sda2    8:2    0 488.1G  0 part
├─sda3    8:3    0     1K  0 part
├─sda4    8:4    0 186.3G  0 part
└─sda5    8:5    0 256.6G  0 part /
desd@desd-OptiPlex-5050:/$ 
```

5. Find the IP address of your target board. Find a tool / mechanism to a certain if your host and target machines are connected.

- Finding the IP address of the board

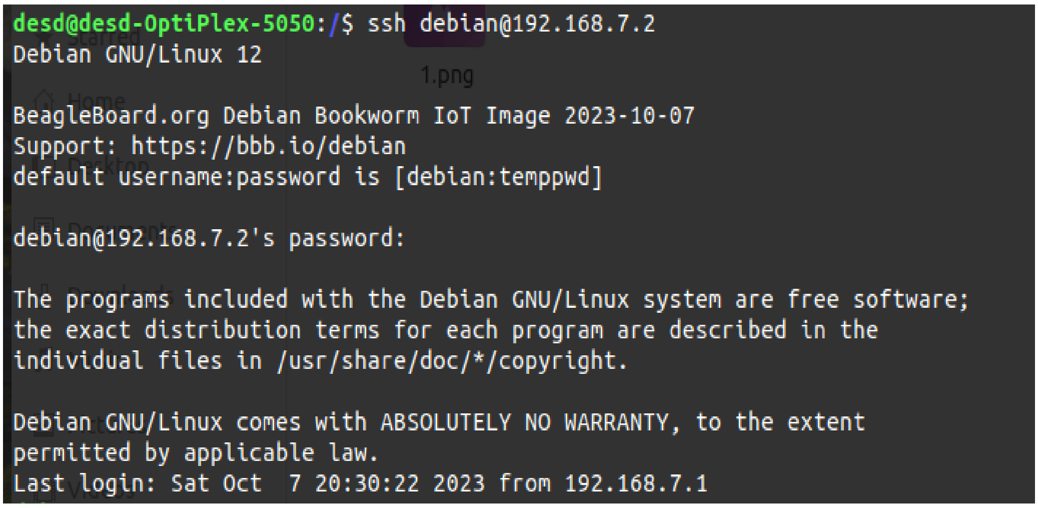
$ if config



```
desd@desd-OptiPlex-5050:/$ ifconfig
enp0s31f6: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.76.108  netmask 255.255.255.0  broadcast 192.168.76.255
        inet6 fe80::9228:b740:99dd:a591  prefixlen 64  scopeid 0x20<link>
        ether 50:9a:4c:37:8e:71  txqueuelen 1000  (Ethernet)
        RX packets 2290093  bytes 1602864015 (1.6 GB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1238531  bytes 168390267 (168.3 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 20  memory 0xf7100000-f7120000

enxe415f6f38f96: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.7.1  netmask 255.255.255.0  broadcast 192.168.7.255
        inet6 fe80::5ac7:ae93:aef1:8ff  prefixlen 64  scopeid 0x20<link>
        ether e4:15:f6:f3:8f:96  txqueuelen 1000  (Ethernet)
        RX packets 246  bytes 35855 (35.8 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 254  bytes 35397 (35.3 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 124641  bytes 12026979 (12.0 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 124641  bytes 12026979 (12.0 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

- Host and target connection check

$ ssh debian@192.168.7.2

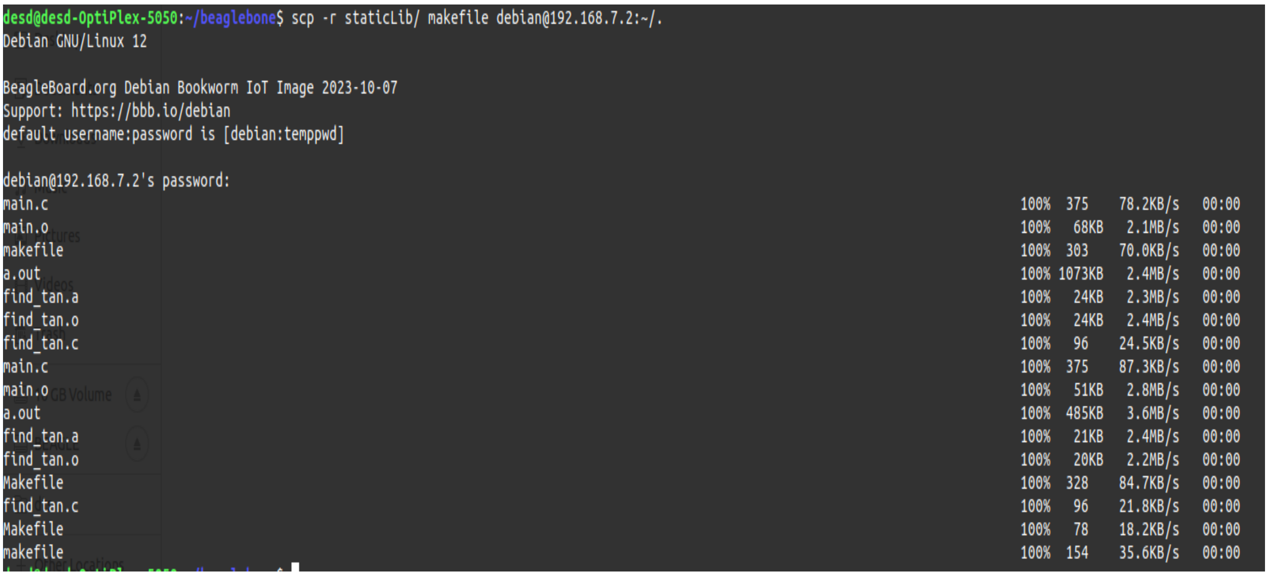


```
desd@desd-OptiPlex-5050:/$ ssh debian@192.168.7.2
Debian GNU/Linux 12

BeagleBoard.org Debian Bookworm IoT Image 2023-10-07
Support: https://bbb.io/debian
default username:password is [debian:temppwd]

debian@192.168.7.2's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Oct  7 20:30:22 2023 from 192.168.7.1
```

6. Use the secure copy (scp) tool to copy files from host machine to target machine and vice versa.

- Copy file from host machine to BBB board

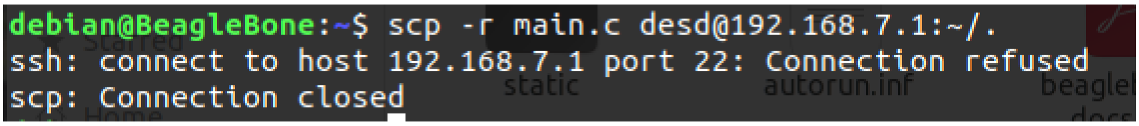
  $ scp -r <file name> deabian@192.168.7.2 :~/.



```
desd@desd-OptiPlex-5050:~/beaglebone$ scp -r staticLib/ makefile debian@192.168.7.2:~/.
Debian GNU/Linux 12

BeagleBoard.org Debian Bookworm IoT Image 2023-10-07
Support: https://bbb.io/debian
default username:password is [debian:temppwd]

debian@192.168.7.2's password:
main.c                                                    100%  375    78.2KB/s   00:00
main.o                                                    100%   68KB   2.1MB/s   00:00
makefile                                                  100%  303    70.0KB/s   00:00
a.out                                                     100% 1073KB   2.4MB/s   00:00
find_tan.a                                                100%   24KB   2.3MB/s   00:00
find_tan.o                                                100%   24KB   2.4MB/s   00:00
find_tan.c                                                100%   96    24.5KB/s   00:00
main.c                                                    100%  375    87.3KB/s   00:00
main.o                                                    100%   51KB   2.8MB/s   00:00
a.out                                                     100%  485KB   3.6MB/s   00:00
find_tan.a                                                100%   21KB   2.4MB/s   00:00
find_tan.o                                                100%   20KB   2.2MB/s   00:00
Makefile                                                  100%  328    84.7KB/s   00:00
find_tan.c                                                100%   96    21.8KB/s   00:00
Makefile                                                  100%   78    18.2KB/s   00:00
makefile                                                  100%  154    35.6KB/s   00:00
```

- Copy file from host machine to BBB board

  $ scp -r <file name> desd@192.168.7.1



```
debian@BeagleBone:~$ scp -r main.c desd@192.168.7.1:~/.
ssh: connect to host 192.168.7.1 port 22: Connection refused
scp: Connection closed
```

7. Cross-Compile a simple C program on the Host machine and try to execute it on the host machine. Note the observations.

- Create a simple C program or a directory

  $ vim <filename>.c   (for file)

  $ mkdir <name> (for creating directory)

- Run the program on gcc compiler on host machine (intel x86)

  $ gcc <filename>.c -o <filename>.out

- Execute the file

  $ ./<filename>.out

  (Print the output message successfully)

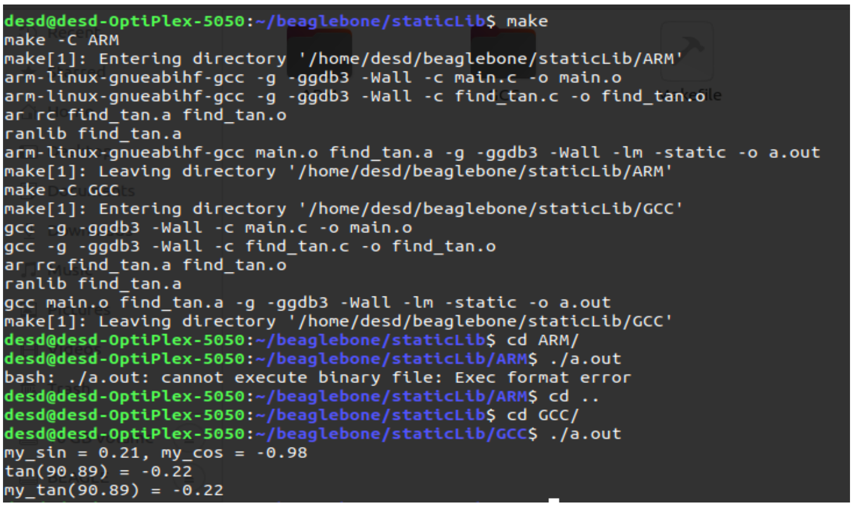
- Now run the program on arm linux compiler on host machine

  $ arm-linux-gnueabihf-gcc <filename>.c -o <filename>.out

- Execute the file

  $ ./<filename>.out

  (Print the error message : Exec format error)



```
desd@desd-OptiPlex-5050:~/beaglebone/staticLib$ make
make -C ARM
make[1]: Entering directory '/home/desd/beaglebone/staticLib/ARM'
arm-linux-gnueabihf-gcc -g -ggdb3 -Wall -c main.c -o main.o
arm-linux-gnueabihf-gcc -g -ggdb3 -Wall -c find_tan.c -o find_tan.o
ar rc find_tan.a find_tan.o
ranlib find_tan.a
arm-linux-gnueabihf-gcc main.o find_tan.a -g -ggdb3 -Wall -lm -static -o a.out
make[1]: Leaving directory '/home/desd/beaglebone/staticLib/ARM'
make -C GCC
make[1]: Entering directory '/home/desd/beaglebone/staticLib/GCC'
gcc -g -ggdb3 -Wall -c main.c -o main.o
gcc -g -ggdb3 -Wall -c find_tan.c -o find_tan.o
ar rc find_tan.a find_tan.o
ranlib find_tan.a
gcc main.o find_tan.a -g -ggdb3 -Wall -lm -static -o a.out
make[1]: Leaving directory '/home/desd/beaglebone/staticLib/GCC'
desd@desd-OptiPlex-5050:~/beaglebone/staticLib$ cd ARM/
desd@desd-OptiPlex-5050:~/beaglebone/staticLib/ARM$ ./a.out
bash: ./a.out: cannot execute binary file: Exec format error
desd@desd-OptiPlex-5050:~/beaglebone/staticLib/ARM$ cd ..
desd@desd-OptiPlex-5050:~/beaglebone/staticLib$ cd GCC/
desd@desd-OptiPlex-5050:~/beaglebone/staticLib/GCC$ ./a.out
my_sin = 0.21, my_cos = -0.98
tan(90.89) = -0.22
my_tan(90.89) = -0.22
```

8. Cross-Compile a simple C program on the Host machine and transfer it to the Embedded Linux Board (target machine). Execute it on the board.

CREATING, COMPILING and EXECUTING FILE ON HOST

- Create a simple C program or a directory

```
$ vim <filename>.c   (for file)

$ mkdir <name>  (for creating directory)
```

- Run the program on gcc compiler on host machine (intel x86)

```
$ gcc <filename>.c  -o  <filename>.out
```

- Execute the file

```
$ ./<filename>.out
```

(Print the output message successfully)

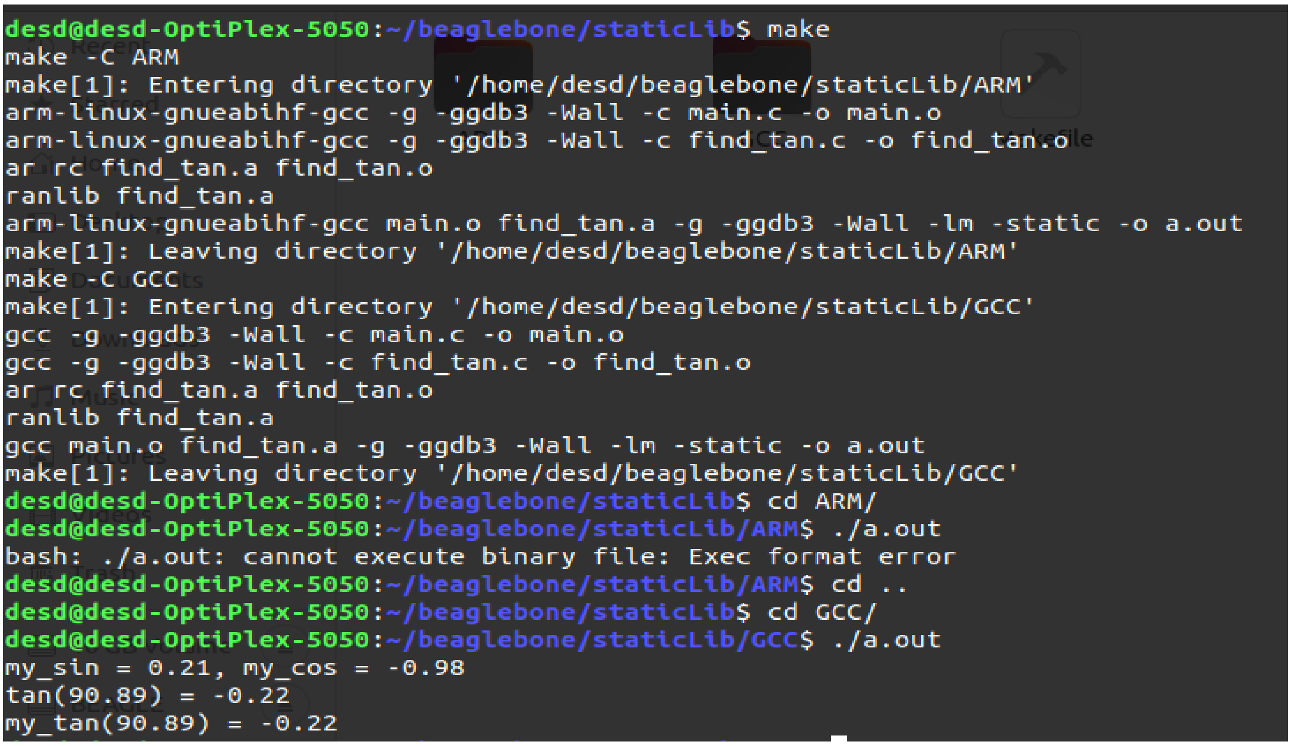
- Now run the program on arm linux compiler on host machine

```
$ arm-linux-gnueabihf-gcc <filename>.c  -o  <filename>.out
```

- Execute the file

```
$ ./<filename>.out
```

(Print the error message : Exec format error)

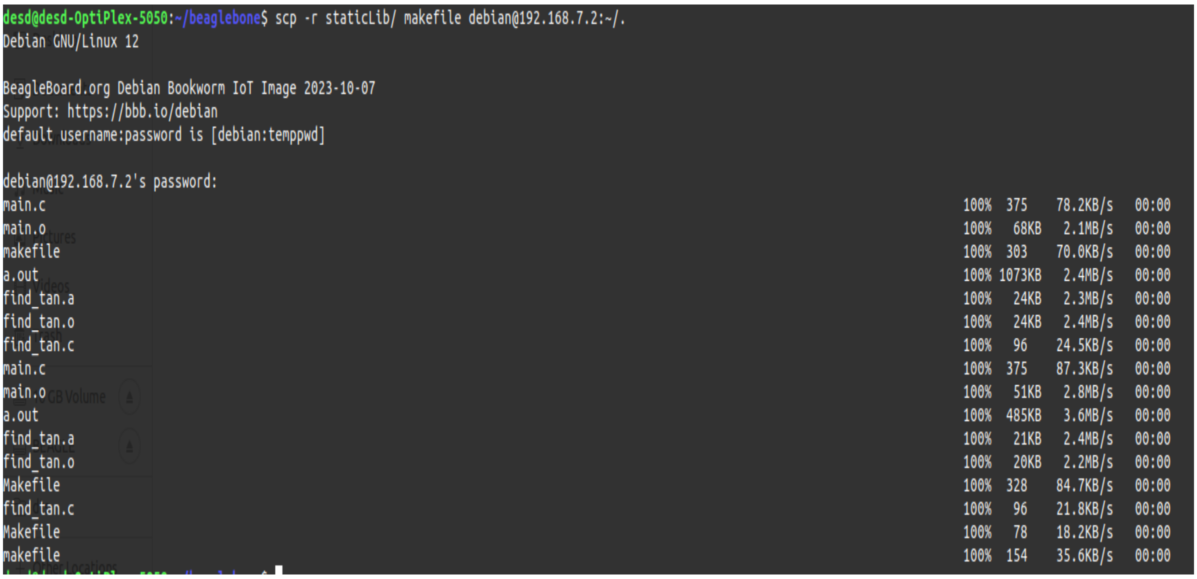


```
desd@desd-OptiPlex-5050:~/beaglebone/staticLib$ make
make -C ARM
make[1]: Entering directory '/home/desd/beaglebone/staticLib/ARM'
arm-linux-gnueabihf-gcc -g -ggdb3 -Wall -c main.c -o main.o
arm-linux-gnueabihf-gcc -g -ggdb3 -Wall -c find_tan.c -o find_tan.o
ar rc find_tan.a find_tan.o
ranlib find_tan.a
arm-linux-gnueabihf-gcc main.o find_tan.a -g -ggdb3 -Wall -lm -static -o a.out
make[1]: Leaving directory '/home/desd/beaglebone/staticLib/ARM'
make -C GCC
make[1]: Entering directory '/home/desd/beaglebone/staticLib/GCC'
gcc -g -ggdb3 -Wall -c main.c -o main.o
gcc -g -ggdb3 -Wall -c find_tan.c -o find_tan.o
ar rc find_tan.a find_tan.o
ranlib find_tan.a
gcc main.o find_tan.a -g -ggdb3 -Wall -lm -static -o a.out
make[1]: Leaving directory '/home/desd/beaglebone/staticLib/GCC'
desd@desd-OptiPlex-5050:~/beaglebone/staticLib$ cd ARM/
desd@desd-OptiPlex-5050:~/beaglebone/staticLib/ARM$ ./a.out
bash: ./a.out: cannot execute binary file: Exec format error
desd@desd-OptiPlex-5050:~/beaglebone/staticLib/ARM$ cd ..
desd@desd-OptiPlex-5050:~/beaglebone/staticLib$ cd GCC/
desd@desd-OptiPlex-5050:~/beaglebone/staticLib/GCC$ ./a.out
my_sin = 0.21, my_cos = -0.98
tan(90.89) = -0.22
my_tan(90.89) = -0.22
```

TRANSFER DATA TO BBB BOARD

- Command to transfer secure copy of data

  $ scp -r <filename> debian@192.168.7.2:~/.



```
desd@desd-OptiPlex-5050:~/beaglebone$ scp -r staticLib/ makefile debian@192.168.7.2:~/.
Debian GNU/Linux 12

BeagleBoard.org Debian Bookworm IoT Image 2023-10-07
Support: https://bbb.io/debian
default username:password is [debian:temppwd]

debian@192.168.7.2's password:
main.c                                                                        100%  375    78.2KB/s   00:00
main.o                                                                        100%   68KB   2.1MB/s   00:00
makefile                                                                      100%  303    70.0KB/s   00:00
a.out                                                                         100% 1073KB   2.4MB/s   00:00
find_tan.a                                                                    100%   24KB   2.3MB/s   00:00
find_tan.o                                                                    100%   24KB   2.4MB/s   00:00
find_tan.c                                                                    100%   96    24.5KB/s   00:00
main.c                                                                        100%  375    87.3KB/s   00:00
main.o                                                                        100%   51KB   2.8MB/s   00:00
a.out                                                                         100%  485KB   3.6MB/s   00:00
find_tan.a                                                                    100%   21KB   2.4MB/s   00:00
find_tan.o                                                                    100%   20KB   2.2MB/s   00:00
Makefile                                                                      100%  328    84.7KB/s   00:00
find_tan.c                                                                    100%   96    21.8KB/s   00:00
Makefile                                                                      100%   78    18.2KB/s   00:00
makefile                                                                      100%  154    35.6KB/s   00:00
```

CHECK and EXECUTING FILE ON BBB board

- Checking if file/directory received or not

  $ ls

- Execute the file compile by gcc compiler

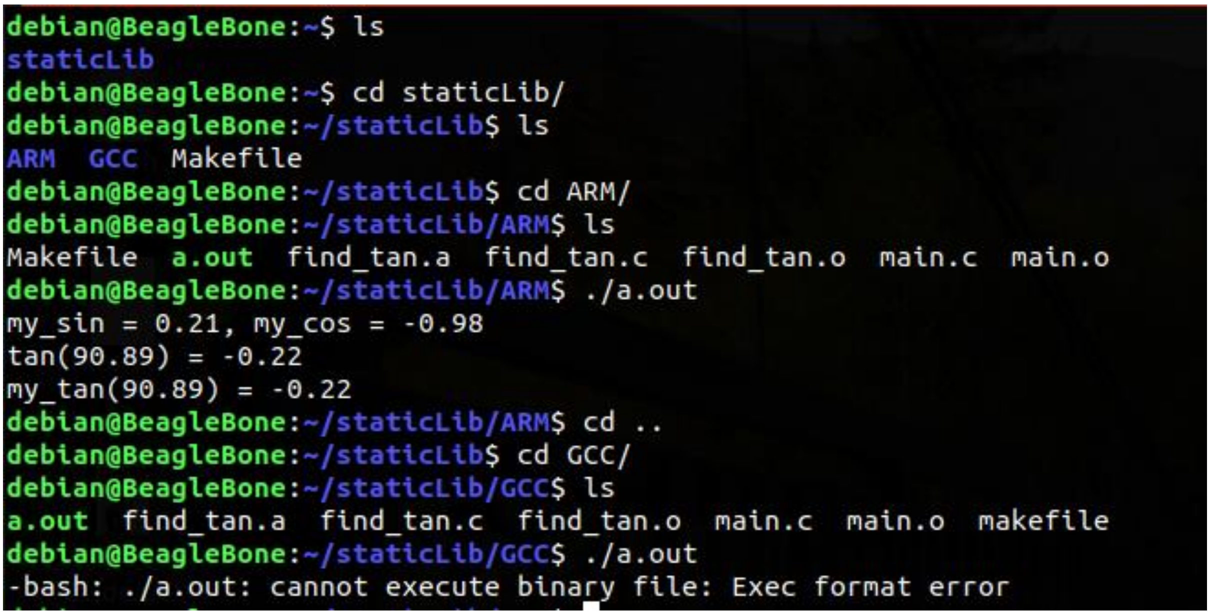
  $ ./<filename>.out

  (Print the error message : Exec format error)

- Execute the file compile by arm-linux compiler

  $ ./<filename>.out

  (Print the output message successfully)

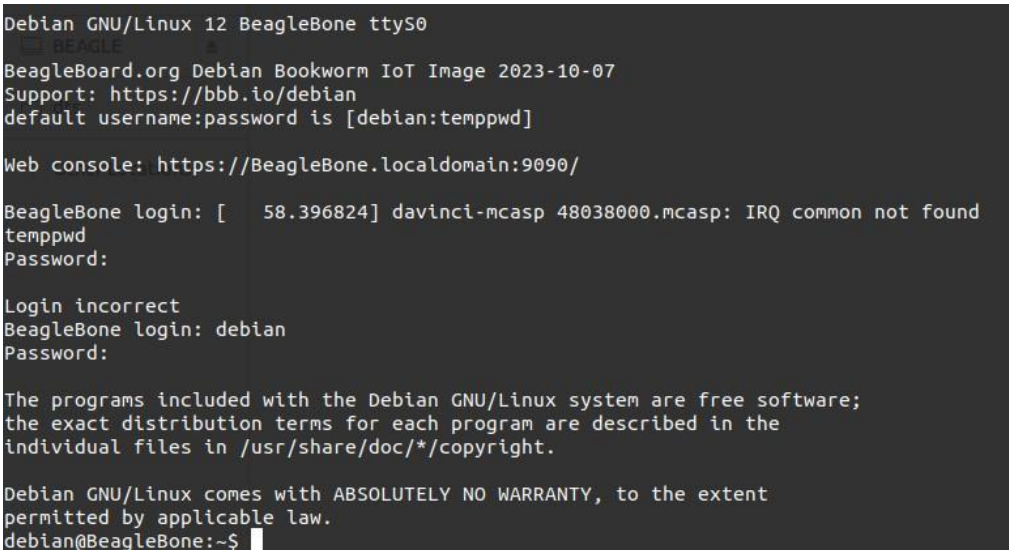


```
debian@BeagleBone:~$ ls
staticLib
debian@BeagleBone:~$ cd staticLib/
debian@BeagleBone:~/staticLib$ ls
ARM  GCC  Makefile
debian@BeagleBone:~/staticLib$ cd ARM/
debian@BeagleBone:~/staticLib/ARM$ ls
Makefile  a.out  find_tan.a  find_tan.c  find_tan.o  main.c  main.o
debian@BeagleBone:~/staticLib/ARM$ ./a.out
my_sin = 0.21, my_cos = -0.98
tan(90.89) = -0.22
my_tan(90.89) = -0.22
debian@BeagleBone:~/staticLib/ARM$ cd ..
debian@BeagleBone:~/staticLib$ cd GCC/
debian@BeagleBone:~/staticLib/GCC$ ls
a.out  find_tan.a  find_tan.c  find_tan.o  main.c  main.o  makefile
debian@BeagleBone:~/staticLib/GCC$ ./a.out
-bash: ./a.out: cannot execute binary file: Exec format error
```

9. Compile a simple C program on the Embedded Linux Board and execute it on the board

- Login to BBB board

  $ ssh debian@192.168.7.2



```
Debian GNU/Linux 12 BeagleBone ttyS0

BeagleBoard.org Debian Bookworm IoT Image 2023-10-07
Support: https://bbb.io/debian
default username:password is [debian:temppwd]

Web console: https://BeagleBone.localdomain:9090/

BeagleBone login: [   58.396824] davinci-mcasp 48038000.mcasp: IRQ common not found
temppwd
Password:

Login incorrect
BeagleBone login: debian
Password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
debian@BeagleBone:~$
```

- Create a simple C program

  $ vim <filename>.c

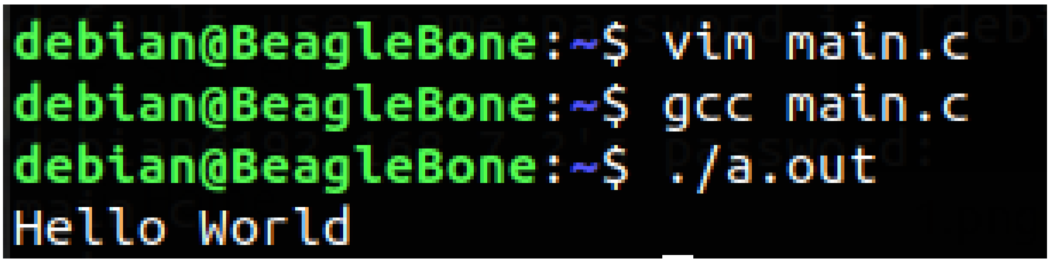
- Run the program

  $ gcc <filename>.c

- Execute the file

  $ ./<filename>.out

  (Print the output message successfully)



```
debian@BeagleBone:~$ vim main.c
debian@BeagleBone:~$ gcc main.c
debian@BeagleBone:~$ ./a.out
Hello World
```

12.Investigate the man pages of open, read, write and close systems calls. Note the function prototypes and the significance of each argument.

1. open System Call:

Man Page:

```
$ man open
```

Function Prototype:

```
#include <sys/types.h>

#include <sys/stat.h>

 #include <fcntl.h>
```

- int open(const char *pathname, int flags);
- int open(const char *pathname, int flags, mode_t mode);

Arguments:

- **pathname**: The path of the file to be opened.
- **flags**: The access mode of the file (read-only, write-only, read/write, etc.), and additional options (e.g., creation flags).
- **mode** (optional): The file permissions to be set if the file is created. It is only used when **O_CREAT** flag is present in **flags**.

2. read System Call:

Man Page:

```
$ man read
```

Function Prototype:

```
#include <unistd.h>

ssize_t read(int fd, void *buf, size_t count);
```

Arguments:

- **fd**: File descriptor of the file or socket from which to read.
- **buf**: Buffer where the data will be read into.
- **count**: Number of bytes to read.

Return Value:

- On success, the number of bytes read is returned.
- On error, -1 is returned, and **errno** is set to indicate the error.

3. write System Call:

Man Page:

```
$ man write
```

Function Prototype:

```
#include <unistd.h>
```

- ssize_t write(int fd, const void *buf, size_t count);

Arguments:

- **fd**: File descriptor of the file or socket to which to write.
- **buf**: Buffer containing the data to be written.
- **count**: Number of bytes to write.

Return Value:

- On success, the number of bytes written is returned.
- On error, -1 is returned, and **errno** is set to indicate the error.

4. close System Call:

Man Page:

```
$ man close
```

Function Prototype:

#include <unistd.h>

- int close(int fd);

Arguments:

- **fd**: File descriptor to be closed.

Return Value:

- On success, 0 is returned.
- On error, -1 is returned, and **errno** is set to indicate the error.