ASSESSMENT AND ACTIVITIES

Embedded Linux

LTTS GLOBAL ENGINEERING ACADEMY

L&T Technology Services

Group 1

Poojashri N	99003660
Manisha Chandra	99003684
Reeshav Rout	99003696

Document History

Ver. Rel. No.	Release Date	Prepared. By	Reviewed By	Approved By
1	15-03-2021	Poojashri N, Manisha Chandra Reeshav Rout		
2	16-03-2021	Poojashri N, Manisha Chandra Reeshav Rout		
3	17-03-2021	Poojashri N, Manisha Chandra Reeshav Rout		
4	18-03-2021	Poojashri N, Manisha Chandra Reeshav Rout		
5	19-03-2021	Poojashri N, Manisha Chandra Reeshav Rout		
6	20-03-2021	Poojashri N, Manisha Chandra Reeshav Rout		

CONTENTS

TASK COMPLETION RECORD

ACTIVITY AND CHALLENGES	COMPLETED BY
➢ Activity 1 - Configuration Of The Beaglebone Black Board	Poojashri N
➢ Activity 2 - Comparison Between Raspberry Pie , Dragon, Imx7 Sabre, Bbb	Reeshav Rout
➢ Activity 3 - Comparison In Different Versions Of Bbb And Write The Evolution Of The Beagle Bone.	Reeshav Rout
➢ Activity 4 - Pin Expansion Header Of Bbb	Manisha Chandra
➢ Activity 5 - Testing Mlo And Uboot Sequence	Poojashri N
➢ Activity 6 - Linux Boot Sequence	Manisha Chandra
➢ Activity 7 Changing Banner Name	Reeshav Rout
➢ Activity 8 – Serial Booting Via Uart	Manisha Chandra
➢ Activity 9 - Working With External Peripherals Like Gpio And Connecting Led	Poojashri N
➢ Activity 10 - Creating Our Own Mlo , Uimage And U-Boot.Img	Manisha Chandra
➢ Challenge 1 – Creating Uenv File	Reeshav Rout
➢ Challenge 2 – Uenv.Txt To Automate Tftp Boot	Poojashri N
➢ Challenge 3 –Write A Generic Uenv.Txt	Manisha Chandra
➢ Challenge 4 – Change Autoload Timing	Manisha Chandra
➢ Challenge 5 – Compiling Static Lirary	Poojashri N
➢ Challenge 6 – Compiling Dynamic Library	Reeshav Rout

Activity 1 – CONFIGURATION OF THE BEAGLEBONE BLACK BOARD

Write the step by step configuration of the boards and set up in the window as well as in Linux

- **Linux**
- Connect serial com cable.
- Open terminal and install minicom (sudo apt install minicom).
- Open a new tab, and use command “dmesg” to find the COM port.
- Use command “sudo minicom -s” to open the settings of minicom.
- Go to serial port setup.
- Change the port to the port noted.
- Ensure both software and hardware control is “NO”.
- Save setup as default
- Run minicom (sudo minicom).
- Connect power cable to see board boot up debug messages.

- **Windows**
- Install TeraTerm.
- Follow instructions on this link to allow drivers without signatures to be installed.
- Connect via USB to access and install drivers from the board.
- Remove USB Cable and attach serial cable.
- Check the port of the serial cable in device manager.
- Open Tera Term set port to the port noted down.
- Connect USB cable to see boot up debug messages.

Activity 2 – COMPARISON BETWEEN RASPBERRY PIE , DRAGON, IMX7 SABRE, BBB

SI No	Features	Raspberry pie	Snapdragon Qualcomm	imx7 Sabre	Beagle Bone black
1	Processor Type	It uses ARM11 processor.	It uses octa-core Snapdragon 835 SoC	It uses ARM Cortex-A7 and Cortex-M4	It uses ARM Cortex-A8 processor.
2	RAM	512 MB SDRAM	8GB RAM	1 GB DDR3 SDRAM 8 GB eMMC Flash	512 MB DDR3L
3	GPIO Pins	It has 12 GPIO pins	It has 12 GPIO pins	Eight PMIC GPIO pins are available	It has 69 GPIO pins
4	Min power	It requires a power supply of 700mA	Low power voice activation requires 0.65mA	5V/5A universal power supply	It requires min power of 210mA
5	Processor Speed	It uses 700 MHz for processing	It uses 2.3 GHz for processing	It uses 1 GHz for processing	It uses 1 GHz for its processing

Activity 3 – COMPARISON IN DIFFERENT VERSIONS OF BBB

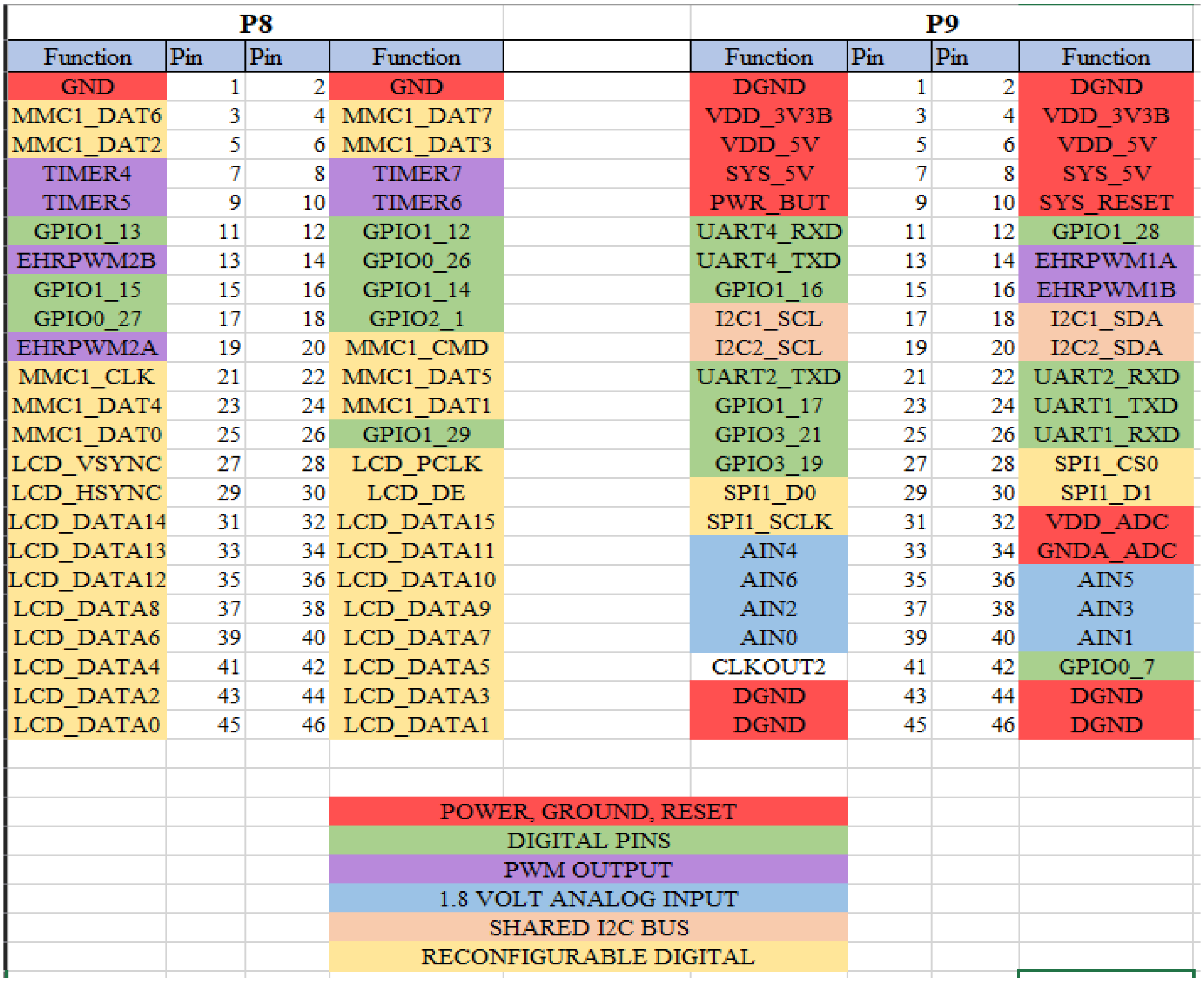
BeagleBone Black has basically 3 versions, A series, B series and at last the current updated boards which is C series.

In the below table the BeagleBone Black board's release date along with version is mentioned.

Version and Board Name	Changes	Date of release
A4	Preliminary	January 4, 2013
A5	Production release	January 8.2013
A5.1	1. Added information on Power button and the battery access points. 2. Final production released version.	April 1 2013
A5.2	1. Edited version. 2. Added numerous pictures of the Rev A5A board.	April 23 2013
A5.3	1. Updated serial number locations. 2. Corrected the feature table for 4 UARTS 3. Corrected eMMC pin table to match other tables in the manual	April 30, 2013
A5.4	1. Corrected revision listed in section 2. Rev A5A is the initial production release. 2. Added all the locations of the serial numbers. 3. Made additions to the compatibility list. 4. Corrected Table 7 for LED GPIO pins. 5. Fixed several typos. 6. Added some additional information about LDOs and StepDown converters. 7. Added short section on HDMI.	May 12, 2013
A5.5	1. Release of the A5B version. 2. The LEDS were dimmed by changing the resistors. 3. The serial termination mode was incorporated into the PCB.	May 20, 2013
A5.6	1. Added information on Rev A5C 2. Added PRU/ICSS options to tables for P8 and P9. 3. Added section on USB Host Correct modes. 4. Fixed a few typos	June 16, 2013
A5.7	1. Updated assembly revision to A6. 2. PCB change to add buffer to the reset line and ground the oscillator GND pin. 3. Added resistor on PCB for connection of OSC_GND to board GND	August 9, 2013
A6	1. Added Rev A6 changes.	October 11, 2013

Activity 4 – PIN EXPANSION HEADER OF BBB

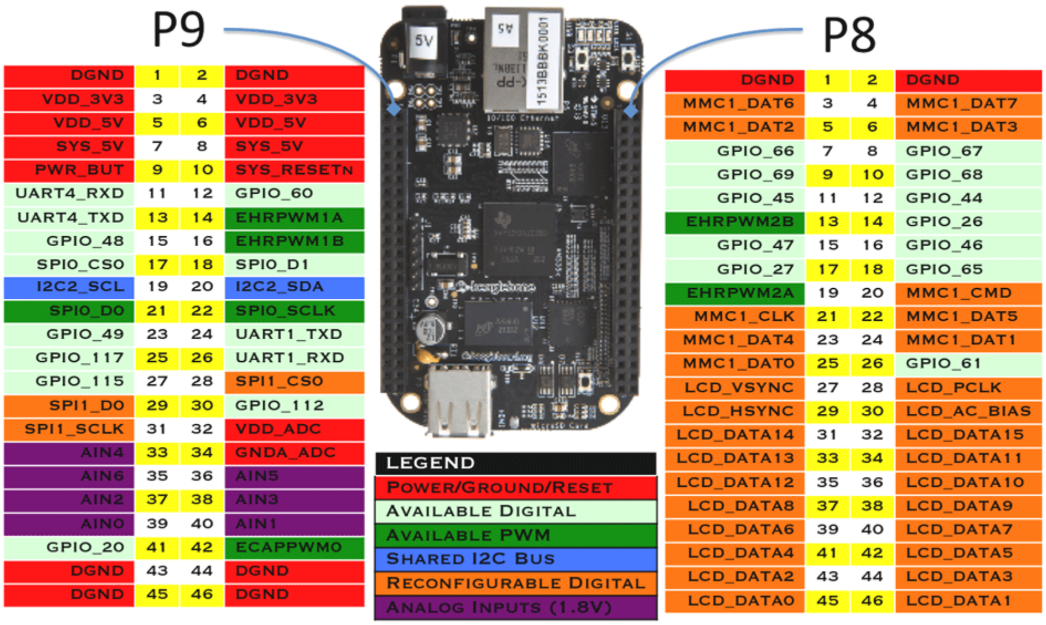
The pin extension according to the data sheet is as follows



P8					P9			
Function	Pin	Pin	Function		Function	Pin	Pin	Function
GND	1	2	GND		DGND	1	2	DGND
MMC1_DAT6	3	4	MMC1_DAT7		VDD_3V3B	3	4	VDD_3V3B
MMC1_DAT2	5	6	MMC1_DAT3		VDD_5V	5	6	VDD_5V
TIMER4	7	8	TIMER7		SYS_5V	7	8	SYS_5V
TIMER5	9	10	TIMER6		PWR_BUT	9	10	SYS_RESET
GPIO1_13	11	12	GPIO1_12		UART4_RXD	11	12	GPIO1_28
EHRPWM2B	13	14	GPIO0_26		UART4_TXD	13	14	EHRPWM1A
GPIO1_15	15	16	GPIO1_14		GPIO1_16	15	16	EHRPWM1B
GPIO0_27	17	18	GPIO2_1		I2C1_SCL	17	18	I2C1_SDA
EHRPWM2A	19	20	MMC1_CMD		I2C2_SCL	19	20	I2C2_SDA
MMC1_CLK	21	22	MMC1_DAT5		UART2_TXD	21	22	UART2_RXD
MMC1_DAT4	23	24	MMC1_DAT1		GPIO1_17	23	24	UART1_TXD
MMC1_DAT0	25	26	GPIO1_29		GPIO3_21	25	26	UART1_RXD
LCD_VSYNC	27	28	LCD_PCLK		GPIO3_19	27	28	SPI1_CS0
LCD_HSYNC	29	30	LCD_DE		SPI1_D0	29	30	SPI1_D1
LCD_DATA14	31	32	LCD_DATA15		SPI1_SCLK	31	32	VDD_ADC
LCD_DATA13	33	34	LCD_DATA11		AIN4	33	34	GNDA_ADC
LCD_DATA12	35	36	LCD_DATA10		AIN6	35	36	AIN5
LCD_DATA8	37	38	LCD_DATA9		AIN2	37	38	AIN3
LCD_DATA6	39	40	LCD_DATA7		AIN0	39	40	AIN1
LCD_DATA4	41	42	LCD_DATA5		CLKOUT2	41	42	GPIO0_7
LCD_DATA2	43	44	LCD_DATA3		DGND	43	44	DGND
LCD_DATA0	45	46	LCD_DATA1		DGND	45	46	DGND

- 🟥 POWER, GROUND, RESET
- 🟩 DIGITAL PINS
- 🟪 PWM OUTPUT
- 🟦 1.8 VOLT ANALOG INPUT
- 🟧 SHARED I2C BUS
- 🟨 RECONFIGURABLE DIGITAL

The software names for the respective pins in default mode



P9

DGND	1	2	DGND
VDD_3V3	3	4	VDD_3V3
VDD_5V	5	6	VDD_5V
SYS_5V	7	8	SYS_5V
PWR_BUT	9	10	SYS_RESETN
UART4_RXD	11	12	GPIO_60
UART4_TXD	13	14	EHRPWM1A
GPIO_48	15	16	EHRPWM1B
SPI0_CS0	17	18	SPI0_D1
I2C2_SCL	19	20	I2C2_SDA
SPI0_D0	21	22	SPI0_SCLK
GPIO_49	23	24	UART1_TXD
GPIO_117	25	26	UART1_RXD
GPIO_115	27	28	SPI1_CS0
SPI1_D0	29	30	GPIO_112
SPI1_SCLK	31	32	VDD_ADC
AIN4	33	34	GNDA_ADC
AIN6	35	36	AIN5
AIN2	37	38	AIN3
AIN0	39	40	AIN1
GPIO_20	41	42	ECAPPWM0
DGND	43	44	DGND
DGND	45	46	DGND

P8

DGND	1	2	DGND
MMC1_DAT6	3	4	MMC1_DAT7
MMC1_DAT2	5	6	MMC1_DAT3
GPIO_66	7	8	GPIO_67
GPIO_69	9	10	GPIO_68
GPIO_45	11	12	GPIO_44
EHRPWM2B	13	14	GPIO_26
GPIO_47	15	16	GPIO_46
GPIO_27	17	18	GPIO_65
EHRPWM2A	19	20	MMC1_CMD
MMC1_CLK	21	22	MMC1_DAT5
MMC1_DAT4	23	24	MMC1_DAT1
MMC1_DAT0	25	26	GPIO_61
LCD_VSYNC	27	28	LCD_PCLK
LCD_HSYNC	29	30	LCD_AC_BIAS
LCD_DATA14	31	32	LCD_DATA15
LCD_DATA13	33	34	LCD_DATA11
LCD_DATA12	35	36	LCD_DATA10
LCD_DATA8	37	38	LCD_DATA9
LCD_DATA6	39	40	LCD_DATA7
LCD_DATA4	41	42	LCD_DATA5
LCD_DATA2	43	44	LCD_DATA3
LCD_DATA0	45	46	LCD_DATA1

LEGEND

Power/Ground/Reset

Available Digital

Available PWM

Shared I2C Bus

Reconfigurable Digital

Analog Inputs (1.8V)

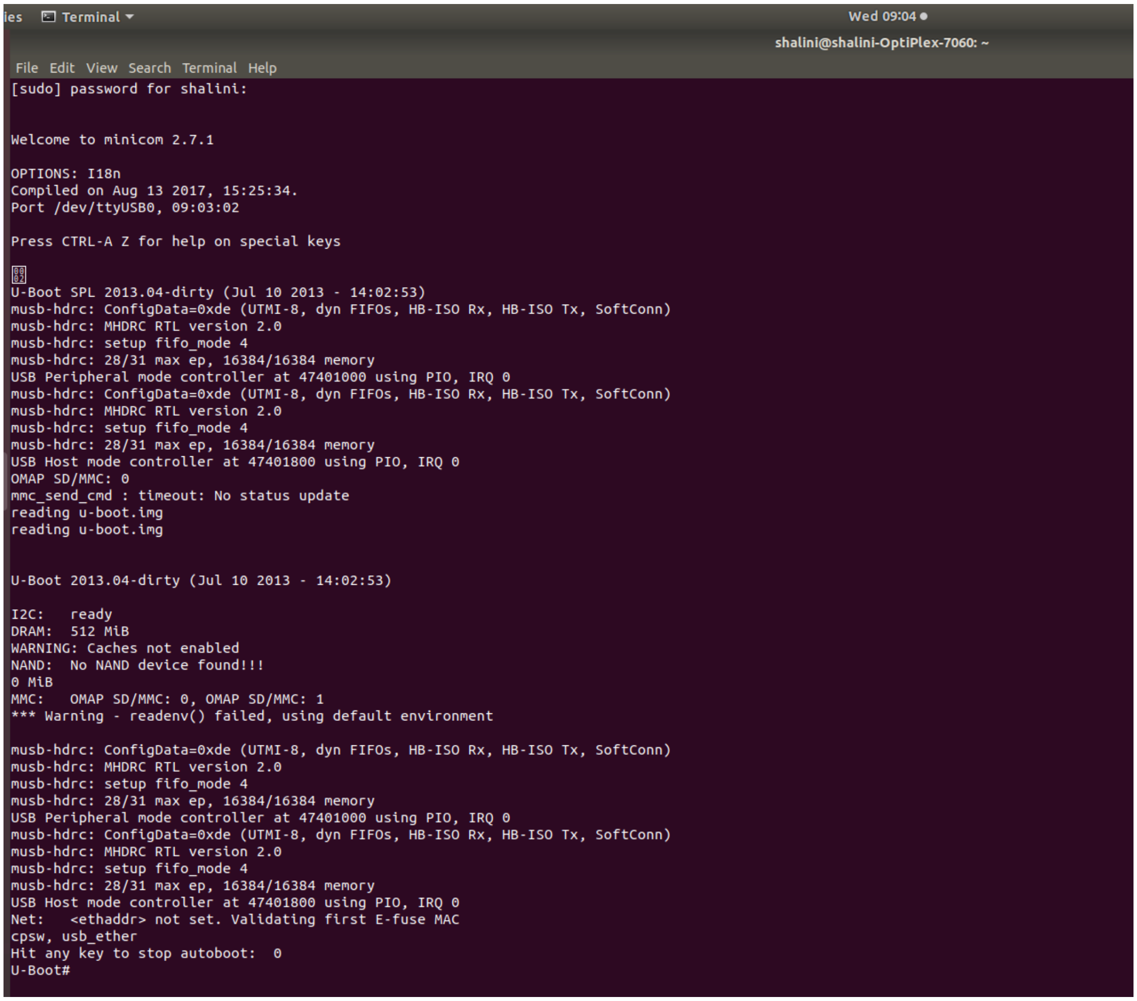
Activity 5 – TESTING MLO AND UBOOT SEQUENCE

MLO

- Partition the SD card.
- Copy the MLO file to the BOOT partition.
- Power up the board while holding s2 button.
- We can see that the boot sequence stops as it cannot find the uImage file.

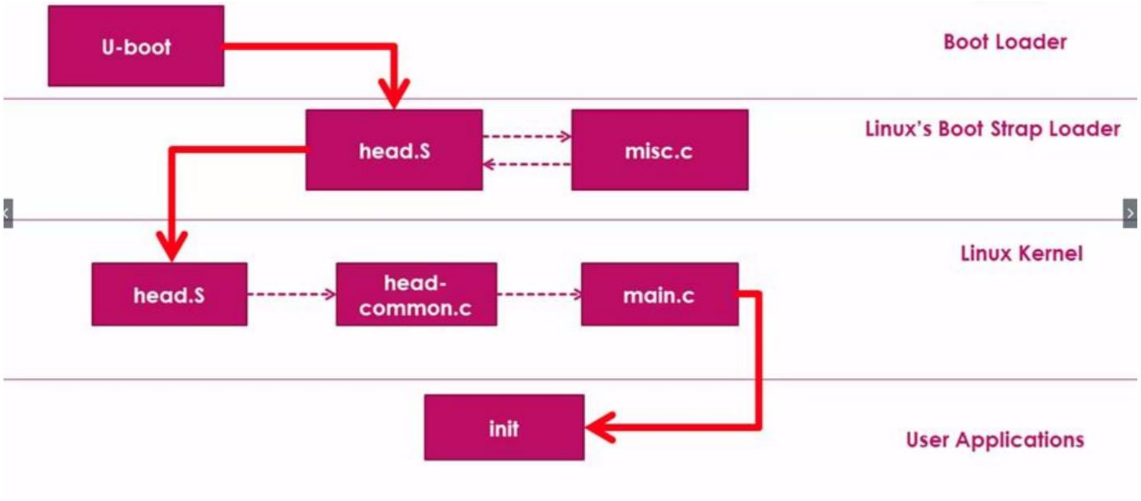
UImage

- Copy uImage to BOOT partition.
- Start booting process.
- Observe serial monitor.
- We reach the following screen.


ies
Terminal
Wed 09:04
shalini@shalini-OptiPlex-7060: ~
File Edit View Search Terminal Help

```
[sudo] password for shalini:


Welcome to minicom 2.7.1

OPTIONS: I18n
Compiled on Aug 13 2017, 15:25:34.
Port /dev/ttyUSB0, 09:03:02

Press CTRL-A Z for help on special keys


U-Boot SPL 2013.04-dirty (Jul 10 2013 - 14:02:53)
musb-hdrc: ConfigData=0xde (UTMI-8, dyn FIFOs, HB-ISO Rx, HB-ISO Tx, SoftConn)
musb-hdrc: MHDRC RTL version 2.0
musb-hdrc: setup fifo_mode 4
musb-hdrc: 28/31 max ep, 16384/16384 memory
USB Peripheral mode controller at 47401000 using PIO, IRQ 0
musb-hdrc: ConfigData=0xde (UTMI-8, dyn FIFOs, HB-ISO Rx, HB-ISO Tx, SoftConn)
musb-hdrc: MHDRC RTL version 2.0
musb-hdrc: setup fifo_mode 4
musb-hdrc: 28/31 max ep, 16384/16384 memory
USB Host mode controller at 47401800 using PIO, IRQ 0
OMAP SD/MMC: 0
mmc_send_cmd : timeout: No status update
reading u-boot.img
reading u-boot.img


U-Boot 2013.04-dirty (Jul 10 2013 - 14:02:53)

I2C:   ready
DRAM:  512 MiB
WARNING: Caches not enabled
NAND:  No NAND device found!!!
0 MiB
MMC:   OMAP SD/MMC: 0, OMAP SD/MMC: 1
*** Warning - readenv() failed, using default environment

musb-hdrc: ConfigData=0xde (UTMI-8, dyn FIFOs, HB-ISO Rx, HB-ISO Tx, SoftConn)
musb-hdrc: MHDRC RTL version 2.0
musb-hdrc: setup fifo_mode 4
musb-hdrc: 28/31 max ep, 16384/16384 memory
USB Peripheral mode controller at 47401000 using PIO, IRQ 0
musb-hdrc: ConfigData=0xde (UTMI-8, dyn FIFOs, HB-ISO Rx, HB-ISO Tx, SoftConn)
musb-hdrc: MHDRC RTL version 2.0
musb-hdrc: setup fifo_mode 4
musb-hdrc: 28/31 max ep, 16384/16384 memory
USB Host mode controller at 47401800 using PIO, IRQ 0
Net:   <ethaddr> not set. Validating first E-fuse MAC
cpsw, usb_ether
Hit any key to stop autoboot:  0
U-Boot#
```

Activity 6 – LINUX BOOT SEQUENCE



The above diagram shows the control flow of the boot process.

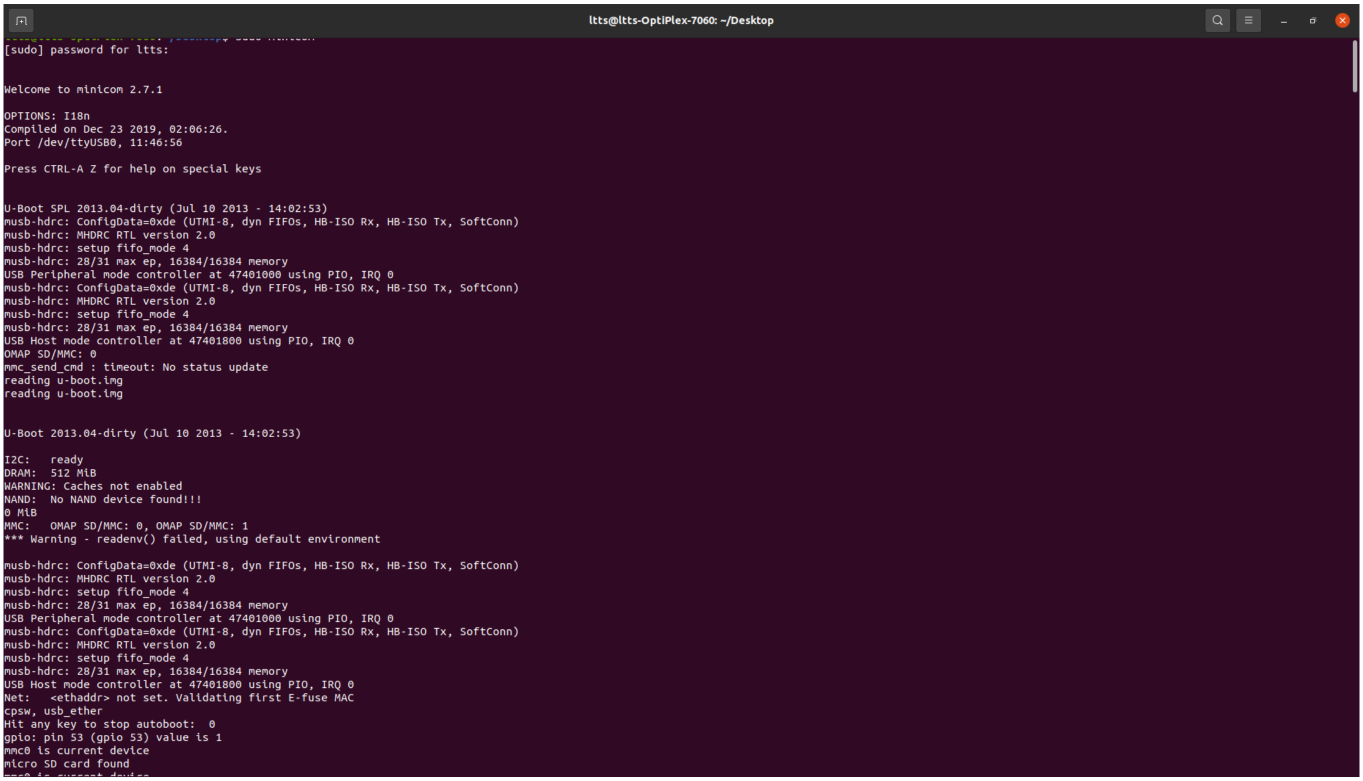
The steps involved are.

- CPU specific initializations
- Checks for valid processor architecture.
- Page table inits.
- Initialize and prepare MMU for the identified Processor Architecture.
- Enable MMU to support virtual memory.
- Calls "start_kernel" function of the main.c ("Arch" independent code).

To understand the handover from uboot to the boot strap loader we need to go through some files involved in the different stages of the boot process.

First Stage

- File Location: *u-boot-2017.05-rc2/arch/arm/lib*
- First open the file bootm.c.
- In this file we have a function "kernel_entry". This function holds 2 very important arguments i.e., r1 = machine id, r2 = DTB address.
- These values are passed on to Boot strap loader through the "kernel_entry" function.



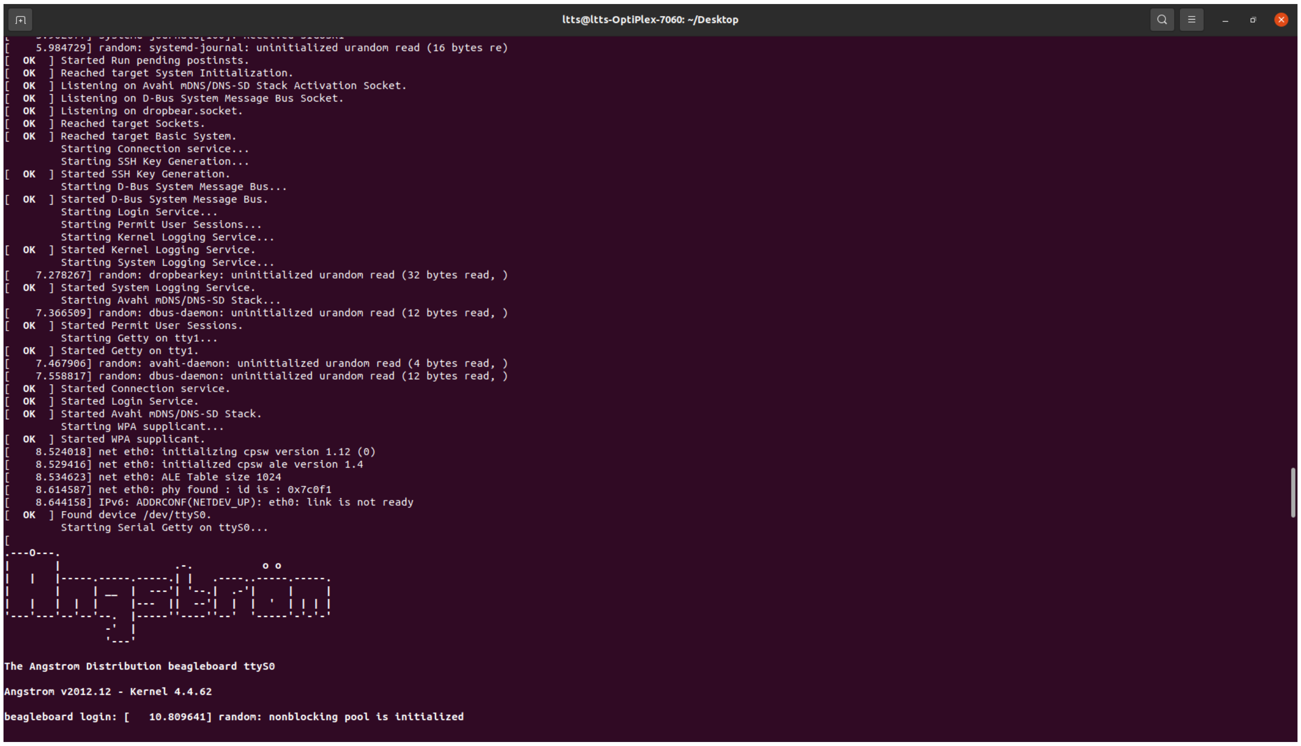
```
[sudo] password for ltts:


Welcome to minicom 2.7.1

OPTIONS: I18n
Compiled on Dec 23 2019, 02:06:26.
Port /dev/ttyUSB0, 11:46:56

Press CTRL-A Z for help on special keys


U-Boot SPL 2013.04-dirty (Jul 10 2013 - 14:02:53)
musb-hdrc: ConfigData=0xde (UTMI-8, dyn FIFOs, HB-ISO Rx, HB-ISO Tx, SoftConn)
musb-hdrc: MHDRC RTL version 2.0
musb-hdrc: setup fifo_mode 4
musb-hdrc: 28/31 max ep, 16384/16384 memory
USB Peripheral mode controller at 47401000 using PIO, IRQ 0
musb-hdrc: ConfigData=0xde (UTMI-8, dyn FIFOs, HB-ISO Rx, HB-ISO Tx, SoftConn)
musb-hdrc: MHDRC RTL version 2.0
musb-hdrc: setup fifo_mode 4
musb-hdrc: 28/31 max ep, 16384/16384 memory
USB Host mode controller at 47401800 using PIO, IRQ 0
OMAP SD/MMC: 0
mmc_send_cmd : timeout: No status update
reading u-boot.img
reading u-boot.img


U-Boot 2013.04-dirty (Jul 10 2013 - 14:02:53)

I2C:   ready
DRAM:  512 MiB
WARNING: Caches not enabled
NAND:  No NAND device found!!!
0 MiB
MMC:   OMAP SD/MMC: 0, OMAP SD/MMC: 1
*** Warning - readenv() failed, using default environment

musb-hdrc: ConfigData=0xde (UTMI-8, dyn FIFOs, HB-ISO Rx, HB-ISO Tx, SoftConn)
musb-hdrc: MHDRC RTL version 2.0
musb-hdrc: setup fifo_mode 4
musb-hdrc: 28/31 max ep, 16384/16384 memory
USB Peripheral mode controller at 47401000 using PIO, IRQ 0
musb-hdrc: ConfigData=0xde (UTMI-8, dyn FIFOs, HB-ISO Rx, HB-ISO Tx, SoftConn)
musb-hdrc: MHDRC RTL version 2.0
musb-hdrc: setup fifo_mode 4
musb-hdrc: 28/31 max ep, 16384/16384 memory
USB Host mode controller at 47401800 using PIO, IRQ 0
Net:   <ethaddr> not set. Validating first E-fuse MAC
cpsw, usb_ether
Hit any key to stop autoboot:  0
gpio: pin 53 (gpio 53) value is 1
mmc0 is current device
micro SD card found
mmc0 is current device
```

Second Stage

- File Location: *Embedded_Linux/linux-4.14$ vi arch/arm/boot/compressed/*
- Coming to the Boot strap Loader we have two file – head.S and misc.c.
- The role of head.S file is to catch the r1 and r2 values and initiate the Linux decompression.
- For this there is a branch line named "decompress_kernel" which takes the flow to the misc.c file where the decompression of the kernel is to happen.
- The flow comes back to the head. S file where the function "enter_kernel" sends the control flow to the third stage.

Third Stage

- File Location: *Embedded_Linux/linux-4.14$ vi arch/arm/kernel/*
- In this stage three files are generated – head.S, head-common.S, and the main.c.
- The role of the head.S file is to disable the MMU, setup the initial page tables to the barest amount, which is required to get the kernel running, which generally means mapping in the kernel code and reinitialize the MMU.
- The flow then goes to the head-common.S file.
- The following fragment of code is executed with the MMU on in MMU mode and uses absolute addresses; this is not position independent.
- The absolute address can be obtained by looking up in the lookup_procssor_type.



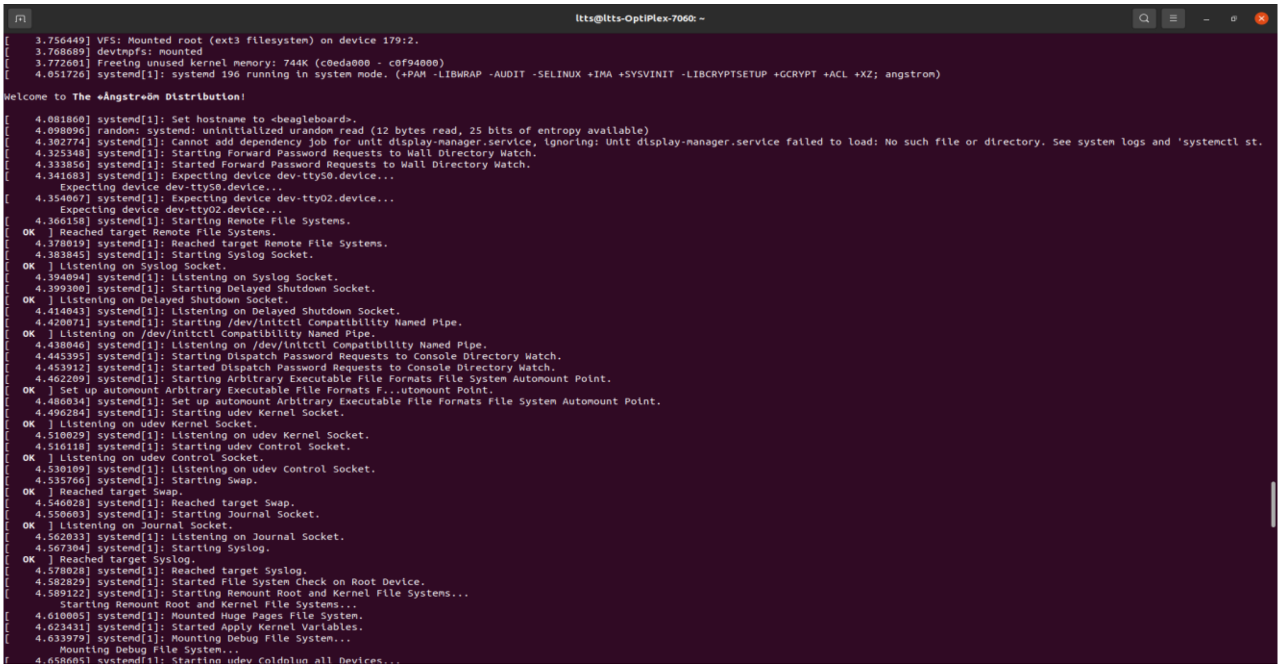
```
[    5.962077] systemd-journald[100]: Received SIGUSR1
[    5.984729] random: systemd-journal: uninitialized urandom read (16 bytes re)
[  OK  ] Started Run pending postinsts.
[  OK  ] Reached target System Initialization.
[  OK  ] Listening on Avahi mDNS/DNS-SD Stack Activation Socket.
[  OK  ] Listening on D-Bus System Message Bus Socket.
[  OK  ] Listening on dropbear.socket.
[  OK  ] Reached target Sockets.
[  OK  ] Reached target Basic System.
         Starting Connection service...
         Starting SSH Key Generation...
[  OK  ] Started SSH Key Generation.
         Starting D-Bus System Message Bus...
[  OK  ] Started D-Bus System Message Bus.
         Starting Login Service...
         Starting Permit User Sessions...
         Starting Kernel Logging Service...
[  OK  ] Started Kernel Logging Service.
         Starting System Logging Service...
[    7.278267] random: dropbearkey: uninitialized urandom read (32 bytes read, )
[  OK  ] Started System Logging Service.
         Starting Avahi mDNS/DNS-SD Stack...
[    7.366509] random: dbus-daemon: uninitialized urandom read (12 bytes read, )
[  OK  ] Started Permit User Sessions.
         Starting Getty on tty1...
[  OK  ] Started Getty on tty1.
[    7.467906] random: avahi-daemon: uninitialized urandom read (4 bytes read, )
[    7.558817] random: dbus-daemon: uninitialized urandom read (12 bytes read, )
[  OK  ] Started Connection service.
[  OK  ] Started Login Service.
[  OK  ] Started Avahi mDNS/DNS-SD Stack.
         Starting WPA supplicant...
[  OK  ] Started WPA supplicant.
[    8.524018] net eth0: initializing cpsw version 1.12 (0)
[    8.529416] net eth0: initialized cpsw ale version 1.4
[    8.534623] net eth0: ALE Table size 1024
[    8.614587] net eth0: phy found : id is : 0x7c0f1
[    8.644158] IPv6: ADDRCONF(NETDEV_UP): eth0: link is not ready
[  OK  ] Found device /dev/ttyS0.
         Starting Serial Getty on ttyS0...
[
.---O---.
|       |                  .-.           o o
|   |   |-----.-----.-----.| |   .----..-----.-----.
|       |     | __  |  ---'| '--.|  .-'|     |     |
|   |   |  |  |     |---  ||  --'|  |  |  '  | | | |
'---'---'--'--'--.  |-----''----''--'  '-----'-'-'-'
                -'  |
                '---'

The Angstrom Distribution beagleboard ttyS0

Angstrom v2012.12 - Kernel 4.4.62

beagleboard login: [   10.809641] random: nonblocking pool is initialized
```

Figure : Boot Completion

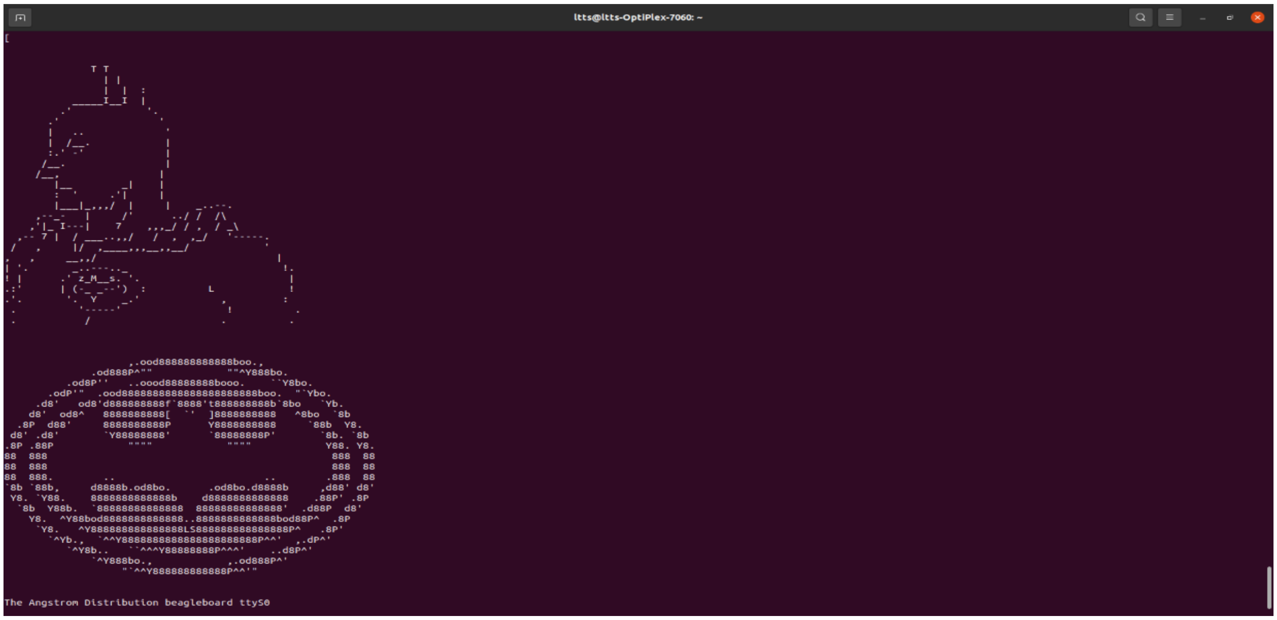
Activity 7 – CHANGING BANNER NAME

- **Changing Banner**

The banner can be changed by locating a file called ‘issue’ inside the folder ‘etc’ in the root file system.



```
[    3.756449] VFS: Mounted root (ext3 filesystem) on device 179:2.
[    3.768689] devtmpfs: mounted
[    3.772601] Freeing unused kernel memory: 744K (c0eda000 - c0f94000)
[    4.051726] systemd[1]: systemd 196 running in system mode. (+PAM -LIBWRAP -AUDIT -SELINUX +IMA +SYSVINIT -LIBCRYPTSETUP +GCRYPT +ACL +XZ; angstrom)

Welcome to The ◆Ångstr◆öm Distribution!

[    4.081860] systemd[1]: Set hostname to <beagleboard>.
[    4.098096] random: systemd: uninitialized urandom read (12 bytes read, 25 bits of entropy available)
[    4.302774] systemd[1]: Cannot add dependency job for unit display-manager.service, ignoring: Unit display-manager.service failed to load: No such file or directory. See system logs and 'systemctl st.
[    4.325348] systemd[1]: Starting Forward Password Requests to Wall Directory Watch.
[    4.333856] systemd[1]: Started Forward Password Requests to Wall Directory Watch.
[    4.341683] systemd[1]: Expecting device dev-ttyS0.device...
         Expecting device dev-ttyS0.device...
[    4.354067] systemd[1]: Expecting device dev-ttyO2.device...
         Expecting device dev-ttyO2.device...
[    4.366158] systemd[1]: Starting Remote File Systems.
[  OK  ] Reached target Remote File Systems.
[    4.378019] systemd[1]: Reached target Remote File Systems.
[    4.383845] systemd[1]: Starting Syslog Socket.
[  OK  ] Listening on Syslog Socket.
[    4.394094] systemd[1]: Listening on Syslog Socket.
[    4.399300] systemd[1]: Starting Delayed Shutdown Socket.
[  OK  ] Listening on Delayed Shutdown Socket.
[    4.414043] systemd[1]: Listening on Delayed Shutdown Socket.
[    4.420071] systemd[1]: Starting /dev/initctl Compatibility Named Pipe.
[  OK  ] Listening on /dev/initctl Compatibility Named Pipe.
[    4.438046] systemd[1]: Listening on /dev/initctl Compatibility Named Pipe.
[    4.445395] systemd[1]: Starting Dispatch Password Requests to Console Directory Watch.
[    4.453912] systemd[1]: Started Dispatch Password Requests to Console Directory Watch.
[    4.462209] systemd[1]: Starting Arbitrary Executable File Formats File System Automount Point.
[  OK  ] Set up automount Arbitrary Executable File Formats F...utomount Point.
[    4.486034] systemd[1]: Set up automount Arbitrary Executable File Formats File System Automount Point.
[    4.496284] systemd[1]: Starting udev Kernel Socket.
[  OK  ] Listening on udev Kernel Socket.
[    4.510029] systemd[1]: Listening on udev Kernel Socket.
[    4.516118] systemd[1]: Starting udev Control Socket.
[  OK  ] Listening on udev Control Socket.
[    4.530109] systemd[1]: Listening on udev Control Socket.
[    4.535766] systemd[1]: Starting Swap.
[  OK  ] Reached target Swap.
[    4.546028] systemd[1]: Reached target Swap.
[    4.550603] systemd[1]: Starting Journal Socket.
[  OK  ] Listening on Journal Socket.
[    4.562033] systemd[1]: Listening on Journal Socket.
[    4.567304] systemd[1]: Starting Syslog.
[  OK  ] Reached target Syslog.
[    4.578028] systemd[1]: Reached target Syslog.
[    4.582829] systemd[1]: Started File System Check on Root Device.
[    4.589122] systemd[1]: Starting Remount Root and Kernel File Systems...
         Starting Remount Root and Kernel File Systems...
[    4.610005] systemd[1]: Mounted Huge Pages File System.
[    4.623431] systemd[1]: Started Apply Kernel Variables.
[    4.633979] systemd[1]: Mounting Debug File System...
         Mounting Debug File System...
[    4.658605] systemd[1]: Starting udev Coldplug all Devices...
```

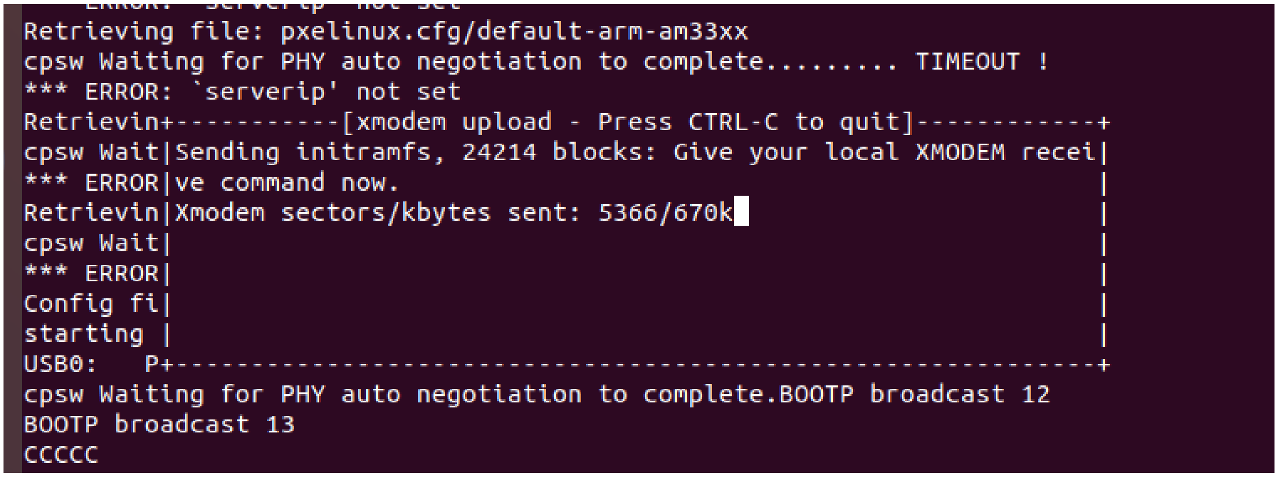


```
[


           T T
             | |
             | |   :
         _____I__I  |
       .'           '.
     .'               '
     |    ..          '
     |  /__.          |
     :.' -'           |
    /__.              |
   /__,              |
      |__.       _|  |
      :  '     .'|   |
      |___|_,,,/  |    |    _..--.
   ,--_-    |     /'     ../ /  /\
  ,'|_ I---|    7     ,,,_/ / ,  / _\
 ,-- 7 |  / ____..,,/   /  ,  ,_/   '-----.
/    ,    |/  ,____,,,__,,__/              '
,  ,     __,,/                            |
| '.     _..---.._                         !.
! |     .' z_M__s. '.                       |
.:'     | (-_ _--')  :         L            !
.'.      '.  Y    _.'            ,          :
 .        '-----'                 !          .
 .          /                    .          .



                ,.ood888888888888boo.,
           .od888P^""                ""^Y888bo.
        .od8P''   ..oood88888888booo.    ``Y8bo.
      .odP'"  .ood8888888888888888888888boo.  "`Ybo.
    .d8'   od8'd888888888f`8888't888888888b`8bo   `Yb.
   d8'  od8^   8888888888[  `'  ]8888888888   ^8bo  `8b
  .8P  d88'     8888888888P      Y8888888888     `88b  Y8.
 d8' .d8'       `Y88888888'      `88888888P'       `8b. `8b
.8P .88P            """"            """"            Y88. Y8.
88  888                                              888  88
88  888                                              888  88
88  888.        ..                        ..        .888  88
`8b `88b,     d8888b.od8bo.      .od8bo.d8888b     ,d88' d8'
 Y8. `Y88.    8888888888888b    d8888888888888    .88P' .8P
  `8b  Y88b.  `88888888888888  88888888888888'  .d88P  d8'
    Y8.  ^Y88bod8888888888888..8888888888888bod88P^  .8P
     `Y8.   ^Y888888888888888LS888888888888888P^   .8P'
       `^Yb.,  `^^Y8888888888888888888888P^^'  ,.dP^'
          `^Y8b..   ``^^^Y88888888P^^^'    ..d8P^'
              `^Y888bo.,            ,.od888P^'
                   "`^^Y888888888888P^^'"


The Angstrom Distribution beagleboard ttyS0
```

Figure New logo

Activity 8 – SERIAL BOOTING VIA UART

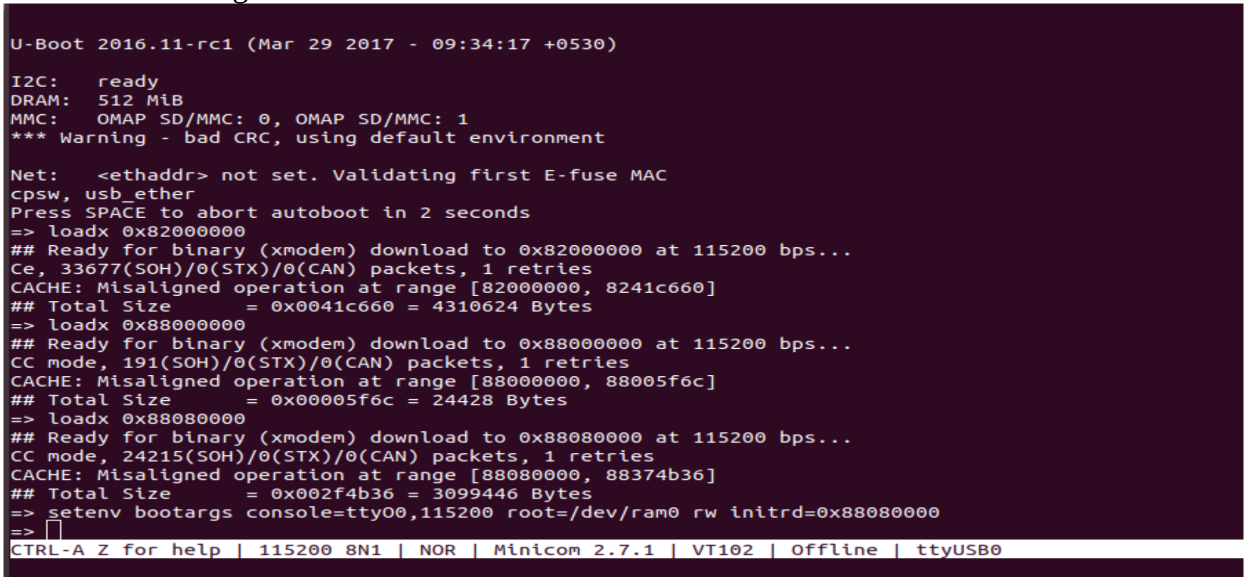
- Connect board using power cable. Without sd card,holding the s2 button will make the board boot up using UART.
- Once booted up holding s2 button, you will reach the loading screen for UART.
- Press Ctrl + A, then S. Select loadx and select the appropriate file to load.
- We first load the uboot.bin file and the uboot.img file.



```
*** ERROR: `serverip' not set
Retrieving file: pxelinux.cfg/default-arm-am33xx
cpsw Waiting for PHY auto negotiation to complete......... TIMEOUT !
*** ERROR: `serverip' not set
Retrievin+-----------[xmodem upload - Press CTRL-C to quit]------------+
cpsw Wait|Sending initramfs, 24214 blocks: Give your local XMODEM recei|
*** ERROR|ve command now.                                              |
Retrievin|Xmodem sectors/kbytes sent: 5366/670k                        |
cpsw Wait|                                                             |
*** ERROR|                                                             |
Config fi|                                                             |
starting |                                                             |
USB0:   P+-------------------------------------------------------------+
cpsw Waiting for PHY auto negotiation to complete.BOOTP broadcast 12
BOOTP broadcast 13
CCCCC
```

Figure 1:Loading files using UART

- The following commands are visible in the screenshot below.



```
U-Boot 2016.11-rc1 (Mar 29 2017 - 09:34:17 +0530)

I2C:   ready
DRAM:  512 MiB
MMC:   OMAP SD/MMC: 0, OMAP SD/MMC: 1
*** Warning - bad CRC, using default environment

Net:   <ethaddr> not set. Validating first E-fuse MAC
cpsw, usb_ether
Press SPACE to abort autoboot in 2 seconds
=> loadx 0x82000000
## Ready for binary (xmodem) download to 0x82000000 at 115200 bps...
Ce, 33677(SOH)/0(STX)/0(CAN) packets, 1 retries
CACHE: Misaligned operation at range [82000000, 8241c660]
## Total Size      = 0x0041c660 = 4310624 Bytes
=> loadx 0x88000000
## Ready for binary (xmodem) download to 0x88000000 at 115200 bps...
CC mode, 191(SOH)/0(STX)/0(CAN) packets, 1 retries
CACHE: Misaligned operation at range [88000000, 88005f6c]
## Total Size      = 0x00005f6c = 24428 Bytes
=> loadx 0x88080000
## Ready for binary (xmodem) download to 0x88080000 at 115200 bps...
CC mode, 24215(SOH)/0(STX)/0(CAN) packets, 1 retries
CACHE: Misaligned operation at range [88080000, 88374b36]
## Total Size      = 0x002f4b36 = 3099446 Bytes
=> setenv bootargs console=ttyO0,115200 root=/dev/ram0 rw initrd=0x88080000
=> 
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | ttyUSB0
```

Figure 2:Serial booting

- We load the uImage, .dtb file and initramfs file into their respective addresses.
- Then we type the following commands
  - setenv bootargs console=ttyo0,115200 root=/dev/ram0 rw initrd=0x88080000
  - bootm 0x82000000 0x88080000 0x88000000

We can boot the board using serial boot in this way.

Activity 9 – WORKING WITH EXTERNAL PERIPHERALS LIKE GPIO AND CONNECTING LED

Steps to connect external led

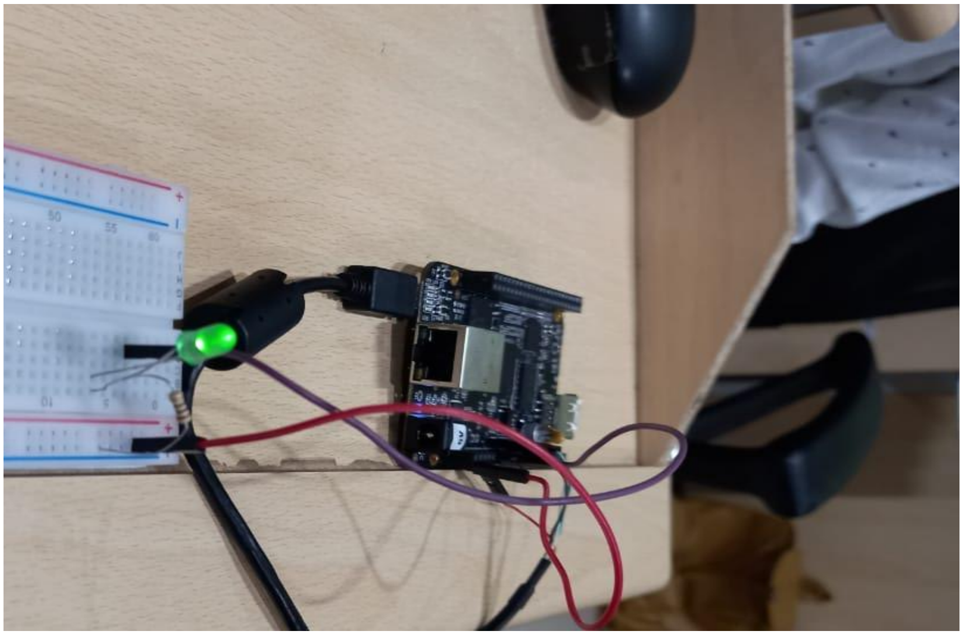
1. Power up BeagleBone Black and login
2. Then go to /sys/class/gpio
3. In gpio if you enter ‘ls’ you will see the following
   ```
   export  gpiochip0  gpiochip32  gpiochip64  gpiochip96  unexport
   ```
4. now export your own pin according to the pin diagram
   ```
   echo 49 > export //gpio_49 is in pin number 23 under pinheader 9
   ```
5. now if you press ‘ls’ you will see:
   ```
   export  gpio49 gpiochip0  gpiochip32  gpiochip64  gpiochip96  unexport
   ```
6. now do cd gpio49
7. now do ‘ls’ to see the following:
   ```
   active_low   direction   power        uevent
   device         edge        subsystem   value
   ```
8. now change direction to ‘out’ and change value to ‘1’
   ```
   echo out > direction
   echo 1 > value
   ```

Steps to connect internal led :

1. Power up BeagleBone Black and login
2. Then go to /sys/class/leds
3. then insert ‘ls’ you will see:
   ```
   beaglebone:green:usr0    beaglebone:green:usr2
   beaglebone:green:usr1    beaglebone:green:usr3
   ```
4. then cd beaglebone:green:usr0
5. then type → echo “heartbeat” > trigger

You will see one of the internal leds of the BeagleBone Black flickering like the heartbeat.

Working with external peripherals like GPIO and Connecting LED



Activity 10 – CREATING OUR OWN MLO , UIMAGE AND U-BOOT.IMG

1. cross tool-chain installation and settings for linux

- -export path of the cross compilation toolchain.
- -Open nano .bashrc
- -add path :
- export PATH=/home/ltts/EMB_LINUX/Backup/crosstoolchain/gcc-linaro-5.5.0-2017.10-x86_64_arm-linux-gnueabihf/bin:$PATH
- -then run this command
- ltts@ltts-OptiPlex-7050:~$ source /home/ltts/.bashrc

2. U-boot Compilation

- open /home/ltts/EMB_LINUX/Backup/u-boot-2017.05-rc2 in terminal
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- distclean
  //distclean : deletes all the previously compiled/generated object files.
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- am335x_boneblack_defconfig
  //apply board default configuration for uboot
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- menuconfig
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- -j6
  //compile
  //use nproc to check the number of processors and the use them accordingly for faster compilation (-j6)

3. Linux compilation

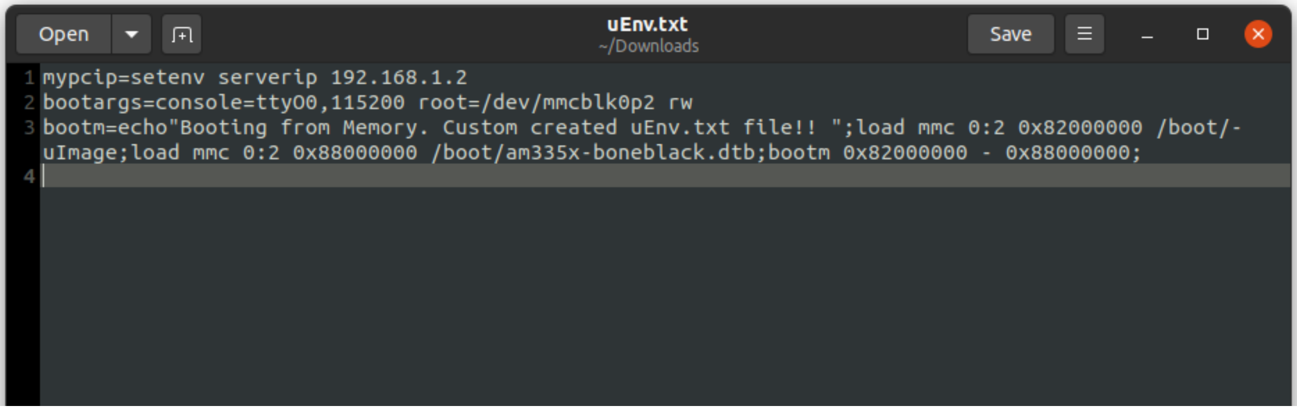
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- distclean
  //Download the kernel from here: https://github.com/beagleboard/linux
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- bb.org_defconfig
  //(4.4) for 4.11 use omap2plus_defconfig
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- menuconfig
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- uImage dtbs LOADADDR=0x80008000 -j6
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- modules -j6
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- INSTALL_MOD_PATH=<path of the RFS> modules_install (path of RFS:/home/ltts/EMB_LINUX/TestBoot/ROOTFS_Static_Lib)

4. Creating our own Root file system using Busy box

- https://busybox.net/
  // download busybox
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- defconfig
  //Apply default configuration
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- menuconfig
  //change default settings if you want
- make ARCH=arm CROSS_COMPILE=arm-linux-gnueabi- CONFIG_PREFIX=<install_path> install
- in my case the <install_path> : /home/ltts/EMB_LINUX/TestBoot/ROOTFS_Static_Lib
  // generate the busy box binary and minimal file system

Challenge 1 - MAKE UENV.TXT TO BOOT FROM MMC0

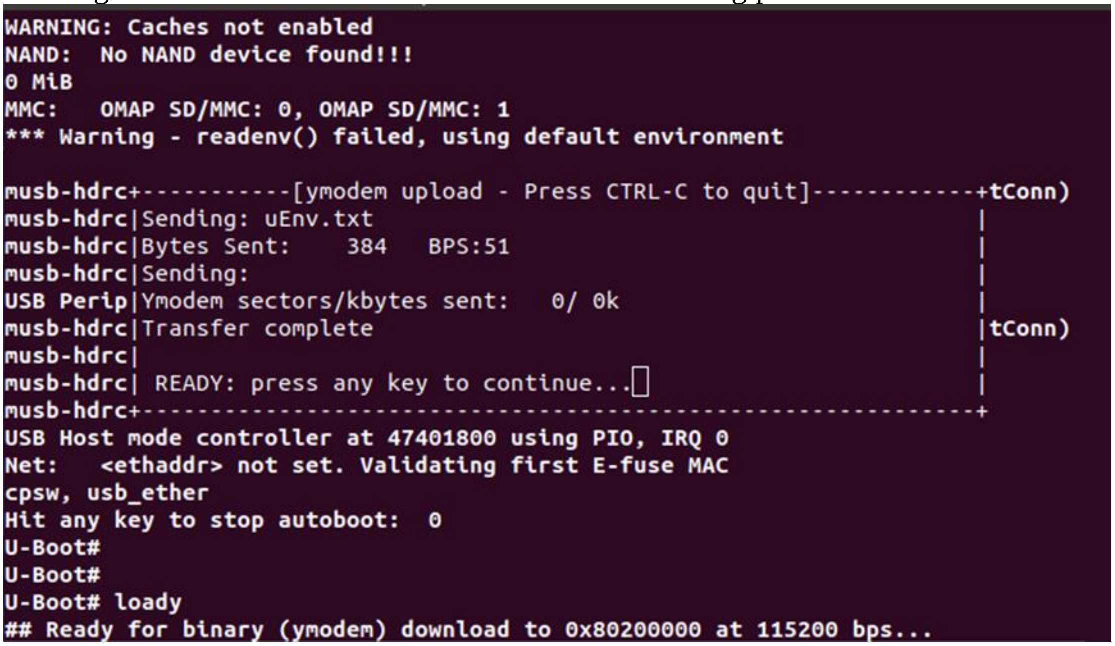
Here is the sample uEnv.txt file that we created.



```
mypcip=setenv serverip 192.168.1.2
bootargs=console=ttyO0,115200 root=/dev/mmcblk0p2 rw
bootm=echo"Booting from Memory. Custom created uEnv.txt file!! ";load mmc 0:2 0x82000000 /boot/-
uImage;load mmc 0:2 0x88000000 /boot/am335x-boneblack.dtb;bootm 0x82000000 - 0x88000000;
```

Figure :uEnv.txt file

- In this file we first store a command into a variable(mypcip), i.e., setting the variable serverip to 192.168.1.2.
- We then stored values to bootargs for the booting process and the entire booting process into a variable.
- The bootm variable loads the .dtb and the uImage file onto the sd card.
- Running the command load bootm would start the booting process from mmc0.



```
WARNING: Caches not enabled
NAND:  No NAND device found!!!
0 MiB
MMC:   OMAP SD/MMC: 0, OMAP SD/MMC: 1
*** Warning - readenv() failed, using default environment

musb-hdrc+----------[ymodem upload - Press CTRL-C to quit]-----------+tConn)
musb-hdrc|Sending: uEnv.txt                                          |
musb-hdrc|Bytes Sent:    384   BPS:51                                |
musb-hdrc|Sending:                                                   |
USB Perip|Ymodem sectors/kbytes sent:   0/ 0k                        |
musb-hdrc|Transfer complete                                          |tConn)
musb-hdrc|                                                           |
musb-hdrc| READY: press any key to continue...                       |
musb-hdrc+-----------------------------------------------------------+
USB Host mode controller at 47401800 using PIO, IRQ 0
Net:   <ethaddr> not set. Validating first E-fuse MAC
cpsw, usb_ether
Hit any key to stop autoboot:  0
U-Boot#
U-Boot#
U-Boot# loady
## Ready for binary (ymodem) download to 0x80200000 at 115200 bps...
```

Figure 3:Loading uENv.txt using loady

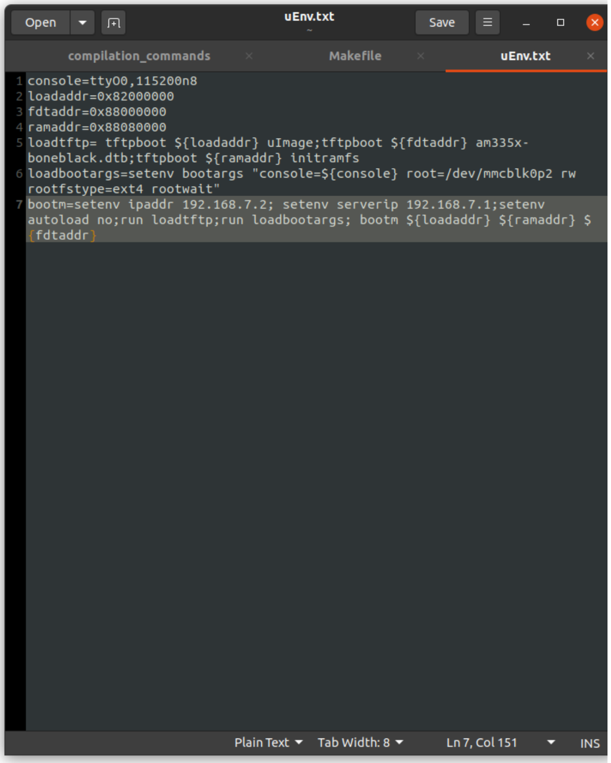
Challenge 2 - UENV.TXT TO AUTOMATE TFTP BOOT

Host

- Connect the tty cable and LAN cable from BBB to the host PC.
- Run “ifconfig” and note down the ethernet port name or set up the connection using “sudo ifconfig <ethernet port> 192.168.7.1 up”.
- Install tftpd on a new terminal.
- Follow the steps for Ubuntu on this link. Instead of the file specified in this link, copy uImage, am335x-boneblack.dtb, initramfs and u-boot.bin to the created directory.

BBB

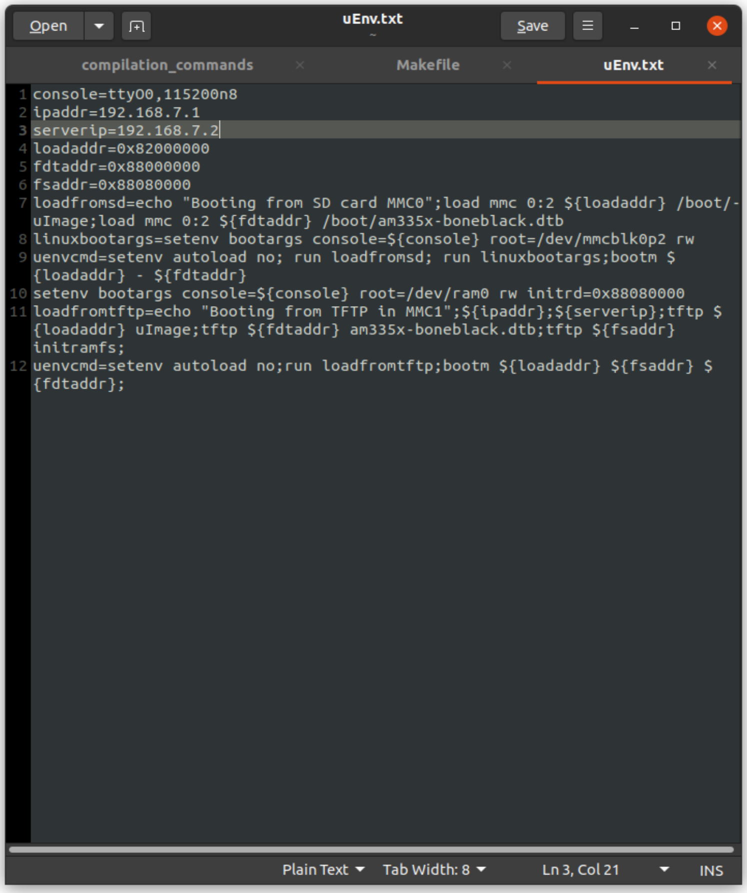
F



```
console=ttyO0,115200n8
loadaddr=0x82000000
fdtaddr=0x88000000
ramaddr=0x88080000
loadtftp= tftpboot ${loadaddr} uImage;tftpboot ${fdtaddr} am335x-boneblack.dtb;tftpboot ${ramaddr} initramfs
loadbootargs=setenv bootargs "console=${console} root=/dev/mmcblk0p2 rw rootfstype=ext4 rootwait"
bootm=setenv ipaddr 192.168.7.2; setenv serverip 192.168.7.1;setenv autoload no;run loadtftp;run loadbootargs; bootm ${loadaddr} ${ramaddr} ${fdtaddr}
```

- Create a uEnv.txt file as shown below.
- Start up the board and stop the booting process at u-boot.
- Load the uEnv.txt file using loady.
- This will load automate the tftp booting

Challenge 3 - WRITE A GENERIC UENV.TXT

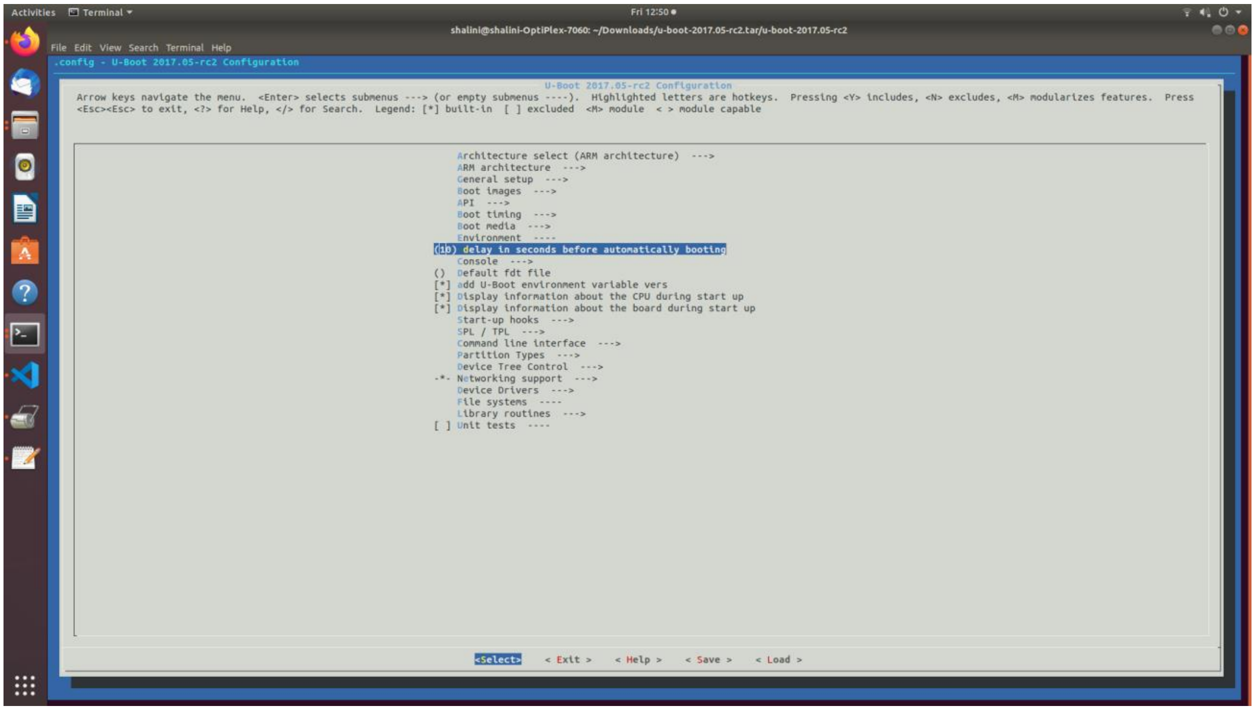


```
console=tty00,115200n8
ipaddr=192.168.7.1
serverip=192.168.7.2
loadaddr=0x82000000
fdtaddr=0x88000000
fsaddr=0x88080000
loadfromsd=echo "Booting from SD card MMC0";load mmc 0:2 ${loadaddr} /boot/-
uImage;load mmc 0:2 ${fdtaddr} /boot/am335x-boneblack.dtb
linuxbootargs=setenv bootargs console=${console} root=/dev/mmcblk0p2 rw
uenvcmd=setenv autoload no; run loadfromsd; run linuxbootargs;bootm $
{loadaddr} - ${fdtaddr}
setenv bootargs console=${console} root=/dev/ram0 rw initrd=0x88080000
loadfromtftp=echo "Booting from TFTP in MMC1";${ipaddr};${serverip};tftp $
{loadaddr} uImage;tftp ${fdtaddr} am335x-boneblack.dtb;tftp ${fsaddr}
initramfs;
uenvcmd=setenv autoload no;run loadfromtftp;bootm ${loadaddr} ${fsaddr} $
{fdtaddr};
```

- Write a uEnv.txt as above.
- Load it onto the BBB boot folder.
- This will generate an generic boot process.

Challenge 4 - CHANGE AUTOLOAD TIMING

- Go to the menuconfig of u-boot.
- Find the option below and change the boot time.



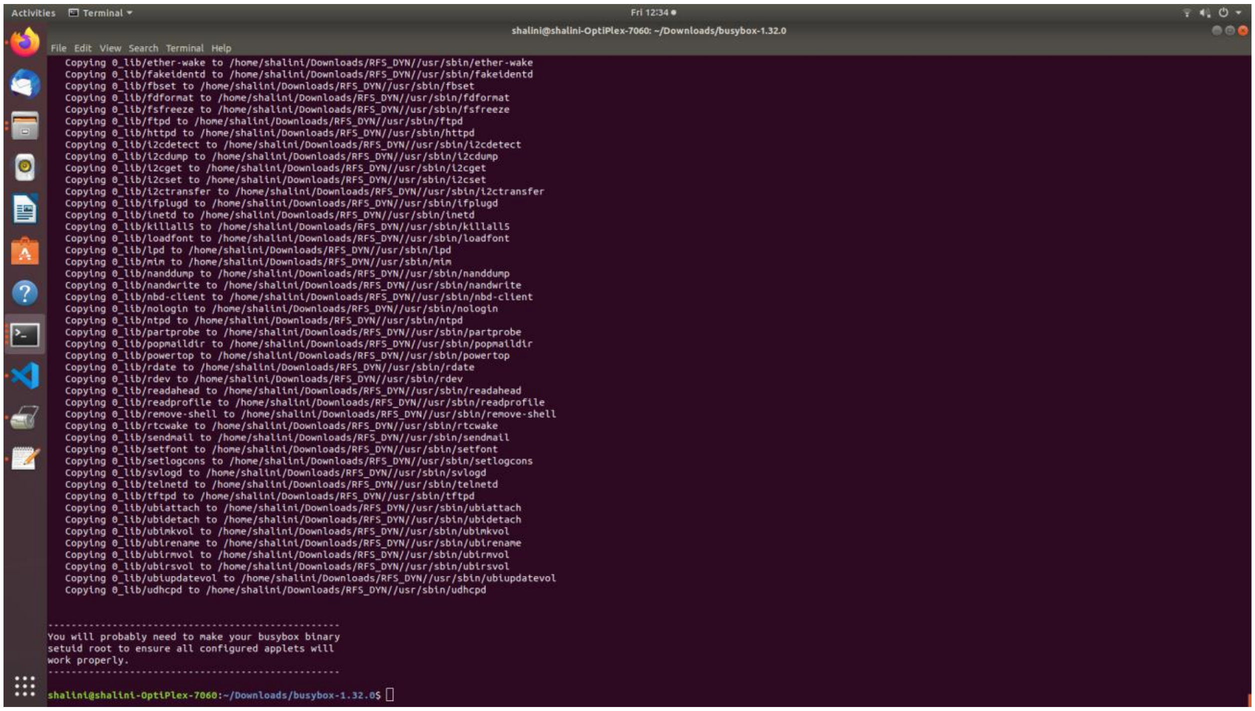
Activities
Terminal
Fri 12:50
shalini@shalini-OptiPlex-7060: ~/Downloads/u-boot-2017.05-rc2.tar/u-boot-2017.05-rc2
File Edit View Search Terminal Help

```
.config - U-Boot 2017.05-rc2 Configuration
                              U-Boot 2017.05-rc2 Configuration
 Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press
 <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

        Architecture select (ARM architecture)  --->
        ARM architecture  --->
        General setup  --->
        Boot images  --->
        API  --->
        Boot timing  --->
        Boot media  --->
        Environment  ----
    (10) delay in seconds before automatically booting
        Console  --->
    ()  Default fdt file
    [*] add U-Boot environment variable vers
    [*] Display information about the CPU during start up
    [*] Display information about the board during start up
        Start-up hooks  --->
        SPL / TPL  --->
        Command line interface  --->
        Partition Types  --->
        Device Tree Control  --->
    -*- Networking support  --->
        Device Drivers  --->
        File systems  ----
        Library routines  --->
    [ ] Unit tests  ----

              <Select>    < Exit >    < Help >    < Save >    < Load >
```

- Using this u-boot.bin file to boot will have a changed auto load timing.

Challenge 5 –COMPILING STATIC LIBRARY

- Follow the same steps for compilation of kernel and u-boot.
- In the menuconfig of busybox select the static option and the busybox will automatically compile for a dynamic library file system.

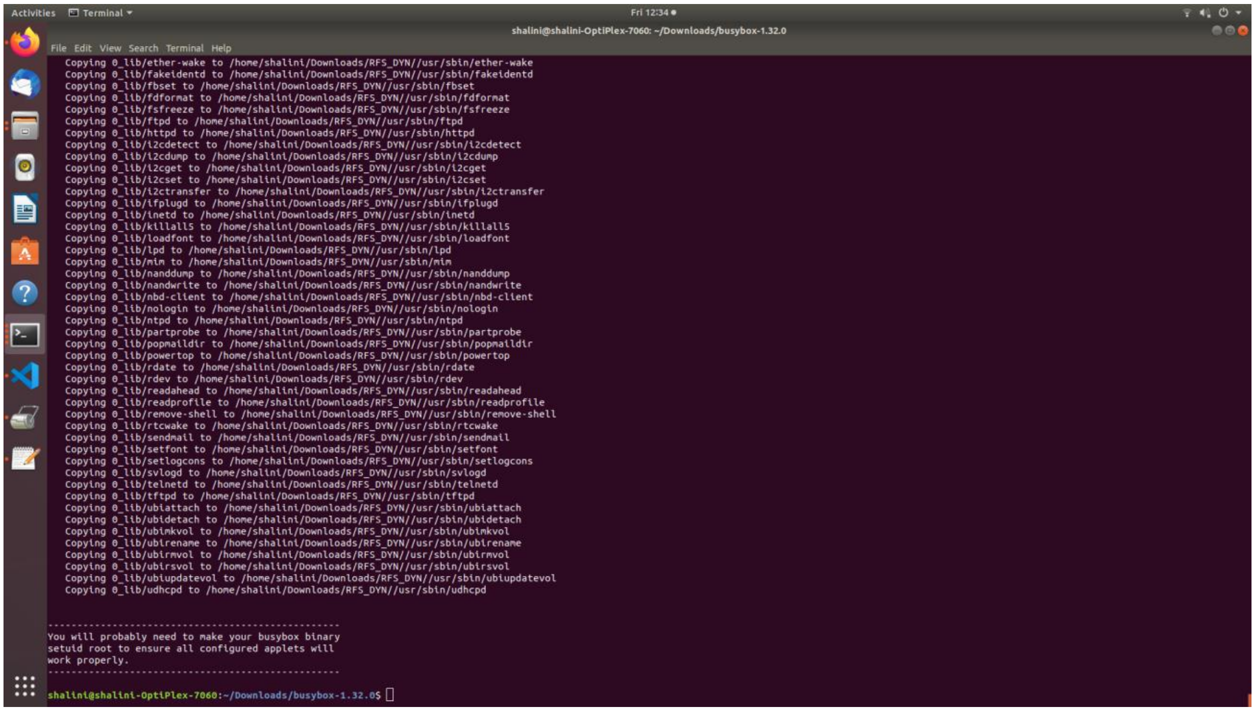


Activities
Terminal
Fri 12:34 ●
shalini@shalini-OptiPlex-7060: ~/Downloads/busybox-1.32.0
File Edit View Search Terminal Help

```
   Copying 0_lib/ether-wake to /home/shalini/Downloads/RFS_DYN//usr/sbin/ether-wake
   Copying 0_lib/fakeidentd to /home/shalini/Downloads/RFS_DYN//usr/sbin/fakeidentd
   Copying 0_lib/fbset to /home/shalini/Downloads/RFS_DYN//usr/sbin/fbset
   Copying 0_lib/fdformat to /home/shalini/Downloads/RFS_DYN//usr/sbin/fdformat
   Copying 0_lib/fsfreeze to /home/shalini/Downloads/RFS_DYN//usr/sbin/fsfreeze
   Copying 0_lib/ftpd to /home/shalini/Downloads/RFS_DYN//usr/sbin/ftpd
   Copying 0_lib/httpd to /home/shalini/Downloads/RFS_DYN//usr/sbin/httpd
   Copying 0_lib/i2cdetect to /home/shalini/Downloads/RFS_DYN//usr/sbin/i2cdetect
   Copying 0_lib/i2cdump to /home/shalini/Downloads/RFS_DYN//usr/sbin/i2cdump
   Copying 0_lib/i2cget to /home/shalini/Downloads/RFS_DYN//usr/sbin/i2cget
   Copying 0_lib/i2cset to /home/shalini/Downloads/RFS_DYN//usr/sbin/i2cset
   Copying 0_lib/i2ctransfer to /home/shalini/Downloads/RFS_DYN//usr/sbin/i2ctransfer
   Copying 0_lib/ifplugd to /home/shalini/Downloads/RFS_DYN//usr/sbin/ifplugd
   Copying 0_lib/inetd to /home/shalini/Downloads/RFS_DYN//usr/sbin/inetd
   Copying 0_lib/killall5 to /home/shalini/Downloads/RFS_DYN//usr/sbin/killall5
   Copying 0_lib/loadfont to /home/shalini/Downloads/RFS_DYN//usr/sbin/loadfont
   Copying 0_lib/lpd to /home/shalini/Downloads/RFS_DYN//usr/sbin/lpd
   Copying 0_lib/mim to /home/shalini/Downloads/RFS_DYN//usr/sbin/mim
   Copying 0_lib/nanddump to /home/shalini/Downloads/RFS_DYN//usr/sbin/nanddump
   Copying 0_lib/nandwrite to /home/shalini/Downloads/RFS_DYN//usr/sbin/nandwrite
   Copying 0_lib/nbd-client to /home/shalini/Downloads/RFS_DYN//usr/sbin/nbd-client
   Copying 0_lib/nologin to /home/shalini/Downloads/RFS_DYN//usr/sbin/nologin
   Copying 0_lib/ntpd to /home/shalini/Downloads/RFS_DYN//usr/sbin/ntpd
   Copying 0_lib/partprobe to /home/shalini/Downloads/RFS_DYN//usr/sbin/partprobe
   Copying 0_lib/popmaildir to /home/shalini/Downloads/RFS_DYN//usr/sbin/popmaildir
   Copying 0_lib/powertop to /home/shalini/Downloads/RFS_DYN//usr/sbin/powertop
   Copying 0_lib/rdate to /home/shalini/Downloads/RFS_DYN//usr/sbin/rdate
   Copying 0_lib/rdev to /home/shalini/Downloads/RFS_DYN//usr/sbin/rdev
   Copying 0_lib/readahead to /home/shalini/Downloads/RFS_DYN//usr/sbin/readahead
   Copying 0_lib/readprofile to /home/shalini/Downloads/RFS_DYN//usr/sbin/readprofile
   Copying 0_lib/remove-shell to /home/shalini/Downloads/RFS_DYN//usr/sbin/remove-shell
   Copying 0_lib/rtcwake to /home/shalini/Downloads/RFS_DYN//usr/sbin/rtcwake
   Copying 0_lib/sendmail to /home/shalini/Downloads/RFS_DYN//usr/sbin/sendmail
   Copying 0_lib/setfont to /home/shalini/Downloads/RFS_DYN//usr/sbin/setfont
   Copying 0_lib/setlogcons to /home/shalini/Downloads/RFS_DYN//usr/sbin/setlogcons
   Copying 0_lib/svlogd to /home/shalini/Downloads/RFS_DYN//usr/sbin/svlogd
   Copying 0_lib/telnetd to /home/shalini/Downloads/RFS_DYN//usr/sbin/telnetd
   Copying 0_lib/tftpd to /home/shalini/Downloads/RFS_DYN//usr/sbin/tftpd
   Copying 0_lib/ubiattach to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubiattach
   Copying 0_lib/ubidetach to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubidetach
   Copying 0_lib/ubimkvol to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubimkvol
   Copying 0_lib/ubirename to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubirename
   Copying 0_lib/ubirmvol to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubirmvol
   Copying 0_lib/ubirsvol to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubirsvol
   Copying 0_lib/ubiupdatevol to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubiupdatevol
   Copying 0_lib/udhcpd to /home/shalini/Downloads/RFS_DYN//usr/sbin/udhcpd


--------------------------------------------------
You will probably need to make your busybox binary
setuid root to ensure all configured applets will
work properly.
--------------------------------------------------

shalini@shalini-OptiPlex-7060:~/Downloads/busybox-1.32.0$ 
```

Challenge 6 COMPILING DYNAMIC LIBRARY

- Follow the same steps for compilation of kernel and u-boot.
- In the menuconfig of busybox do not select the static option and the busybox will automatically compile for a dynamic library file system.


Activities
Terminal
Fri 12:34
shalini@shalini-OptiPlex-7060: ~/Downloads/busybox-1.32.0
File Edit View Search Terminal Help

```
   Copying 0_lib/ether-wake to /home/shalini/Downloads/RFS_DYN//usr/sbin/ether-wake
   Copying 0_lib/fakeidentd to /home/shalini/Downloads/RFS_DYN//usr/sbin/fakeidentd
   Copying 0_lib/fbset to /home/shalini/Downloads/RFS_DYN//usr/sbin/fbset
   Copying 0_lib/fdformat to /home/shalini/Downloads/RFS_DYN//usr/sbin/fdformat
   Copying 0_lib/fsfreeze to /home/shalini/Downloads/RFS_DYN//usr/sbin/fsfreeze
   Copying 0_lib/ftpd to /home/shalini/Downloads/RFS_DYN//usr/sbin/ftpd
   Copying 0_lib/httpd to /home/shalini/Downloads/RFS_DYN//usr/sbin/httpd
   Copying 0_lib/i2cdetect to /home/shalini/Downloads/RFS_DYN//usr/sbin/i2cdetect
   Copying 0_lib/i2cdump to /home/shalini/Downloads/RFS_DYN//usr/sbin/i2cdump
   Copying 0_lib/i2cget to /home/shalini/Downloads/RFS_DYN//usr/sbin/i2cget
   Copying 0_lib/i2cset to /home/shalini/Downloads/RFS_DYN//usr/sbin/i2cset
   Copying 0_lib/i2ctransfer to /home/shalini/Downloads/RFS_DYN//usr/sbin/i2ctransfer
   Copying 0_lib/ifplugd to /home/shalini/Downloads/RFS_DYN//usr/sbin/ifplugd
   Copying 0_lib/inetd to /home/shalini/Downloads/RFS_DYN//usr/sbin/inetd
   Copying 0_lib/killall5 to /home/shalini/Downloads/RFS_DYN//usr/sbin/killall5
   Copying 0_lib/loadfont to /home/shalini/Downloads/RFS_DYN//usr/sbin/loadfont
   Copying 0_lib/lpd to /home/shalini/Downloads/RFS_DYN//usr/sbin/lpd
   Copying 0_lib/mim to /home/shalini/Downloads/RFS_DYN//usr/sbin/mim
   Copying 0_lib/nanddump to /home/shalini/Downloads/RFS_DYN//usr/sbin/nanddump
   Copying 0_lib/nandwrite to /home/shalini/Downloads/RFS_DYN//usr/sbin/nandwrite
   Copying 0_lib/nbd-client to /home/shalini/Downloads/RFS_DYN//usr/sbin/nbd-client
   Copying 0_lib/nologin to /home/shalini/Downloads/RFS_DYN//usr/sbin/nologin
   Copying 0_lib/ntpd to /home/shalini/Downloads/RFS_DYN//usr/sbin/ntpd
   Copying 0_lib/partprobe to /home/shalini/Downloads/RFS_DYN//usr/sbin/partprobe
   Copying 0_lib/popmaildir to /home/shalini/Downloads/RFS_DYN//usr/sbin/popmaildir
   Copying 0_lib/powertop to /home/shalini/Downloads/RFS_DYN//usr/sbin/powertop
   Copying 0_lib/rdate to /home/shalini/Downloads/RFS_DYN//usr/sbin/rdate
   Copying 0_lib/rdev to /home/shalini/Downloads/RFS_DYN//usr/sbin/rdev
   Copying 0_lib/readahead to /home/shalini/Downloads/RFS_DYN//usr/sbin/readahead
   Copying 0_lib/readprofile to /home/shalini/Downloads/RFS_DYN//usr/sbin/readprofile
   Copying 0_lib/remove-shell to /home/shalini/Downloads/RFS_DYN//usr/sbin/remove-shell
   Copying 0_lib/rtcwake to /home/shalini/Downloads/RFS_DYN//usr/sbin/rtcwake
   Copying 0_lib/sendmail to /home/shalini/Downloads/RFS_DYN//usr/sbin/sendmail
   Copying 0_lib/setfont to /home/shalini/Downloads/RFS_DYN//usr/sbin/setfont
   Copying 0_lib/setlogcons to /home/shalini/Downloads/RFS_DYN//usr/sbin/setlogcons
   Copying 0_lib/svlogd to /home/shalini/Downloads/RFS_DYN//usr/sbin/svlogd
   Copying 0_lib/telnetd to /home/shalini/Downloads/RFS_DYN//usr/sbin/telnetd
   Copying 0_lib/tftpd to /home/shalini/Downloads/RFS_DYN//usr/sbin/tftpd
   Copying 0_lib/ubiattach to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubiattach
   Copying 0_lib/ubidetach to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubidetach
   Copying 0_lib/ubimkvol to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubimkvol
   Copying 0_lib/ubirename to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubirename
   Copying 0_lib/ubirmvol to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubirmvol
   Copying 0_lib/ubirsvol to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubirsvol
   Copying 0_lib/ubiupdatevol to /home/shalini/Downloads/RFS_DYN//usr/sbin/ubiupdatevol
   Copying 0_lib/udhcpd to /home/shalini/Downloads/RFS_DYN//usr/sbin/udhcpd


--------------------------------------------------
You will probably need to make your busybox binary
setuid root to ensure all configured applets will
work properly.
--------------------------------------------------

shalini@shalini-OptiPlex-7060:~/Downloads/busybox-1.32.0$ 
```

REFERENCES

1. https://www.youtube.com/watch?v=UMEUo6Wm6u4&list=PLGs0VKk2DiYyThNvj6VyDFmOnQ8ncXk8b&index=2
2. https://www.youtube.com/watch?v=c81tmb7WJxw&list=PLGs0VKk2DiYyThNvj6VyDFmOnQ8ncXk8b&index=3
3. https://elinux.org/EBC_Flashing_an_LED
4. https://toptechboy.com/beaglebone-black-lesson-1-understanding-beaglebone-black-pinout/
5. https://beagleboard.org/getting-started
6. https://learn.adafruit.com/connecting-a-push-button-to-beaglebone-black
7. https://www.google.com/search?channel=fs&client=ubuntu&q=datasheet+of+arm335x+processor
8. https://cdn-shop.adafruit.com/datasheets/BBB_SRM.pdf
9. https://github.com/beagleboard/beaglebone-black/wiki/System-Reference-Manual
10. https://www.ti.com/lit/ds/symlink/am3359.pdf?ts=1615745141545&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FAM3359
11. https://www.ti.com/lit/ug/spruh73q/spruh73q.pdf?ts=1615745144717&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FAM3359
12. https://www.ti.com/product/AM3359