**ESE-3014 EMBEDDED SYSTEMS COMMUNICATION PROTOCOLS AND SECURITY**

**LAB 7 Report**

**GROUP No. 2**

|  |  |
| --- | --- |
| **Group Members** | **Student ID** |
| Christy Rachel Philip | C0765535 |
| Fahad Rahman | C0769871 |
| James M Chacko | C0777192 |
| Premil Presannan | C0777191 |
| Vy Nguyen | C0776242 |

**Step 1: Booting the Host maching and insert the FTDI cable to the USB port, then using the command dmesg | tail:**

Notice that the command has been repeated twice, once when the cable is disconnected and one when it is re-establish:

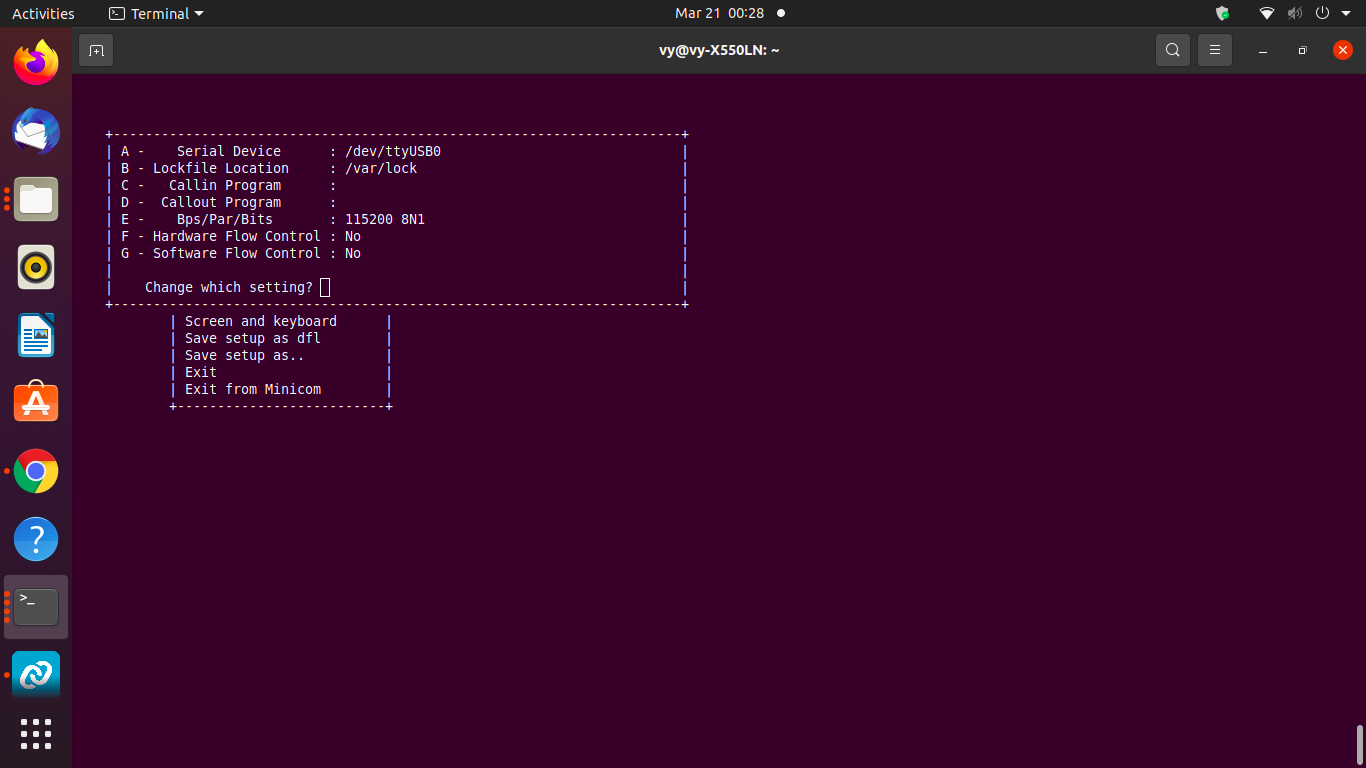
|  |
| --- |
| vy@vy-X550LN:~$ dmesg | tail [29545.182028] cdc\_acm 2-1:1.4: ttyACM0: USB ACM device [29545.735642] cdc\_ether 2-1:1.2 enx0cb2b7d5757e: renamed from eth1 [29545.758657] rndis\_host 2-1:1.0 enx0cb2b7d5757c: renamed from eth0 [29804.104888] usb 2-1: USB disconnect, device number 6 [29804.105033] rndis\_host 2-1:1.0 enx0cb2b7d5757c: unregister 'rndis\_host' usb-0000:00:14.0-1, RNDIS device [29804.127067] cdc\_ether 2-1:1.2 enx0cb2b7d5757e: unregister 'cdc\_ether' usb-0000:00:14.0-1, CDC Ethernet Device [29962.674029] usb 2-2: USB disconnect, device number 4 [29962.674184] ftdi\_sio ttyUSB0: error from flowcontrol urb [29962.676634] ftdi\_sio ttyUSB0: FTDI USB Serial Device converter now disconnected from ttyUSB0 [29962.676689] ftdi\_sio 2-2:1.0: device disconnected  vy@vy-X550LN:~$ dmesg | tail [29962.676689] ftdi\_sio 2-2:1.0: device disconnected [29976.067797] usb 2-2: new full-speed USB device number 7 using xhci\_hcd [29976.221328] usb 2-2: New USB device found, idVendor=0403, idProduct=6001, bcdDevice= 6.00 [29976.221334] usb 2-2: New USB device strings: Mfr=1, Product=2, SerialNumber=3 [29976.221337] usb 2-2: Product: TTL232R-3V3 [29976.221339] usb 2-2: Manufacturer: FTDI [29976.221341] usb 2-2: SerialNumber: FTBNS1D3 [29976.224155] ftdi\_sio 2-2:1.0: FTDI USB Serial Device converter detected [29976.224250] usb 2-2: Detected FT232RL [29976.225005] usb 2-2: FTDI USB Serial Device converter now attached to ttyUSB0 |

A new device is TTL232R-3V3 at ttyUSB0. It is a FTDI USB Serial Device converter

**Step 2: Now connect to beaglebone, we can enable UART4 in uEnv.txt file:**

|  |
| --- |
| #Docs: http://elinux.org/Beagleboard:U-boot\_partitioning\_layout\_2.0  uname\_r=4.14.108-ti-r131 #uuid= #dtb=  ###U-Boot Overlays### ###Documentation: http://elinux.org/Beagleboard:BeagleBoneBlack\_Debian#U-Boot\_Overlays ###Master Enable enable\_uboot\_overlays=1 ### ###Overide capes with eeprom #uboot\_overlay\_addr0=/lib/firmware/<file0>.dtbo #uboot\_overlay\_addr1=/lib/firmware/<file1>.dtbo #uboot\_overlay\_addr2=/lib/firmware/<file2>.dtbo #uboot\_overlay\_addr3=/lib/firmware/<file3>.dtbo ### ###Additional custom capes uboot\_overlay\_addr4=/lib/firmware/BB-UART4-00A0.dtbo #uboot\_overlay\_addr5=/lib/firmware/<file5>.dtbo #uboot\_overlay\_addr6=/lib/firmware/<file6>.dtbo #uboot\_overlay\_addr7=/lib/firmware/<file7>.dtbo ### ###Custom Cape #dtb\_overlay=/lib/firmware/<file8>.dtbo ### ###Disable auto loading of virtual capes (emmc/video/wireless/adc) #disable\_uboot\_overlay\_emmc=1 #disable\_uboot\_overlay\_emmc=1 disable\_uboot\_overlay\_video=1 disable\_uboot\_overlay\_audio=1 #disable\_uboot\_overlay\_wireless=1 #disable\_uboot\_overlay\_adc=1 ### ###PRUSS OPTIONS ###pru\_rproc (4.14.x-ti kernel) uboot\_overlay\_pru=/lib/firmware/AM335X-PRU-RPROC-4-14-TI-00A0.dtbo ###pru\_rproc (4.19.x-ti kernel) #uboot\_overlay\_pru=/lib/firmware/AM335X-PRU-RPROC-4-19-TI-00A0.dtbo ###pru\_uio (4.14.x-ti, 4.19.x-ti & mainline/bone kernel) uboot\_overlay\_pru=/lib/firmware/AM335X-PRU-UIO-00A0.dtbo ### ###Cape Universal Enable enable\_uboot\_cape\_universal=1 ### ###Debug: disable uboot autoload of Cape #disable\_uboot\_overlay\_addr0=1 #disable\_uboot\_overlay\_addr1=1 #disable\_uboot\_overlay\_addr2=1 #disable\_uboot\_overlay\_addr3=1 ### ###U-Boot fdt tweaks... (60000 = 384KB) #uboot\_fdt\_buffer=0x60000 ###U-Boot Overlays### cmdline=coherent\_pool=1M net.ifnames=0 lpj=1990656 rng\_core.default\_quality=100 quiet  cape\_enable=bone\_capemgr.enable\_partno=BB-UART4  #In the event of edid real failures, uncomment this next line: #cmdline=coherent\_pool=1M net.ifnames=0 lpj=1990656 rng\_core.default\_quality=100 quiet video=HDMI-A-1:1024x768@60e  ##enable Generic eMMC Flasher: ##make sure, these tools are installed: dosfstools rsync #cmdline=init=/opt/scripts/tools/eMMC/init-eMMC-flasher-v3.sh |

**Step 3: Setting the minicom setting like this, then reboot the beaglebone to see the booting process.**



Booting using VCP:

|  |
| --- |
| [ \*\*\*] (2 of 3) A start job is running for dev-ttyS0.device (32s / 1min 30s)  U-Boot SPL 2019.04-00002-g07d5700e21 (Mar 06 2020 - 11:24:55 -0600)  Trying to boot from MMC2  Loading Environment from EXT4... Card did not respond to voltage select!      U-Boot 2019.04-00002-g07d5700e21 (Mar 06 2020 - 11:24:55 -0600), Build: jenkin7   CPU : AM335X-GP rev 2.1  I2C: ready  DRAM: 512 MiB  No match for driver 'omap\_hsmmc'  No match for driver 'omap\_hsmmc'  Some drivers were not found  Reset Source: Global external warm reset has occurred.  Reset Source: Power-on reset has occurred.  RTC 32KCLK Source: External.  MMC: OMAP SD/MMC: 0, OMAP SD/MMC: 1  Loading Environment from EXT4... Card did not respond to voltage select!  Board: BeagleBone Black  <ethaddr> not set. Validating first E-fuse MAC  BeagleBone Black:  BeagleBone: cape eeprom: i2c\_probe: 0x54:  BeagleBone: cape eeprom: i2c\_probe: 0x55:  BeagleBone: cape eeprom: i2c\_probe: 0x56:  BeagleBone: cape eeprom: i2c\_probe: 0x57:  Net: eth0: MII MODE  cpsw, usb\_ether  Press SPACE to abort autoboot in 0 seconds  board\_name=[A335BNLT] ...  board\_rev=[000C] ...  Card did not respond to voltage select!  Card did not respond to voltage select!  Card did not respond to voltage select!  gpio: pin 56 (gpio 56) value is 0  gpio: pin 55 (gpio 55) value is 0  gpio: pin 54 (gpio 54) value is 0  gpio: pin 53 (gpio 53) value is 1  Card did not respond to voltage select!  Card did not respond to voltage select!  switch to partitions #0, OK  mmc1(part 0) is current device  Scanning mmc 1:1...  gpio: pin 56 (gpio 56) value is 0  gpio: pin 55 (gpio 55) value is 0  gpio: pin 54 (gpio 54) value is 0  gpio: pin 53 (gpio 53) value is 1  switch to partitions #0, OK  mmc1(part 0) is current device  gpio: pin 54 (gpio 54) value is 1  Checking for: /uEnv.txt ...  Checking for: /boot.scr ...  Checking for: /boot/boot.scr ...  Checking for: /boot/uEnv.txt ...  gpio: pin 55 (gpio 55) value is 1  2114 bytes read in 12 ms (171.9 KiB/s)  Loaded environment from /boot/uEnv.txt  Checking if uname\_r is set in /boot/uEnv.txt...  gpio: pin 56 (gpio 56) value is 1  Running uname\_boot ...  loading /boot/vmlinuz-4.14.108-ti-r131 ...  9654768 bytes read in 623 ms (14.8 MiB/s)  debug: [enable\_uboot\_overlays=1] ...  debug: [enable\_uboot\_cape\_universal=1] ...  debug: [uboot\_base\_dtb\_univ=am335x-boneblack-uboot-univ.dtb] ...  uboot\_overlays: [uboot\_base\_dtb=am335x-boneblack-uboot-univ.dtb] ...  uboot\_overlays: Switching too: dtb=am335x-boneblack-uboot-univ.dtb ...  loading /boot/dtbs/4.14.108-ti-r131/am335x-boneblack-uboot-univ.dtb ...  162971 bytes read in 39 ms (4 MiB/s)  uboot\_overlays: [fdt\_buffer=0x60000] ...  uboot\_overlays: loading /lib/firmware/BB-ADC-00A0.dtbo ...  867 bytes read in 31 ms (26.4 KiB/s)  uboot\_overlays: loading /lib/firmware/BB-UART4-00A0.dtbo ...  1233 bytes read in 52 ms (22.5 KiB/s)  uboot\_overlays: loading /lib/firmware/BB-BONE-eMMC1-01-00A0.dtbo ...  1606 bytes read in 128 ms (11.7 KiB/s)  uboot\_overlays: uboot loading of [/lib/firmware/BB-HDMI-TDA998x-00A0.dtbo] dis. uboot\_overlays: loading /lib/firmware/AM335X-PRU-UIO-00A0.dtbo ...  1035 bytes read in 32 ms (31.3 KiB/s)  loading /boot/initrd.img-4.14.108-ti-r131 ...  3297330 bytes read in 222 ms (14.2 MiB/s)  debug: [console=ttyO0,115200n8 bone\_capemgr.enable\_partno=BB-UART4 bone\_capemg. debug: [bootz 0x82000000 0x88080000:325032 88000000] ...  ## Flattened Device Tree blob at 88000000   Booting using the fdt blob at 0x88000000   Loading Ramdisk to 8fcda000, end 8ffff032 ... OK   Loading Device Tree to 8fc4e000, end 8fcd9fff ... OK    Starting kernel ...    [ 0.002117] timer\_probe: no matching timers found  [ 0.697252] wkup\_m3\_ipc 44e11324.wkup\_m3\_ipc: could not get rproc handle  [ 1.045173] omap\_voltage\_late\_init: Voltage driver support not added  [ 1.052370] PM: Cannot get wkup\_m3\_ipc handle  [ OK ] Listening on Load/Save RF Kill Switch Status /dev/rfkill Watch.  [ OK ] Found device /dev/ttyGS0.  [ OK ] Started Serial Getty on ttyGS0.  [ OK ] Started Generic Board Startup.  [ OK ] Found device /dev/ttyS0.  [ OK ] Started Serial Getty on ttyS0.  [ OK ] Reached target Login Prompts.  [ OK ] Reached target Multi-User System.  [ OK ] Reached target Graphical Interface.   Starting Update UTMP about System Runlevel Changes...  [ OK ] Started Update UTMP about System Runlevel Changes. |

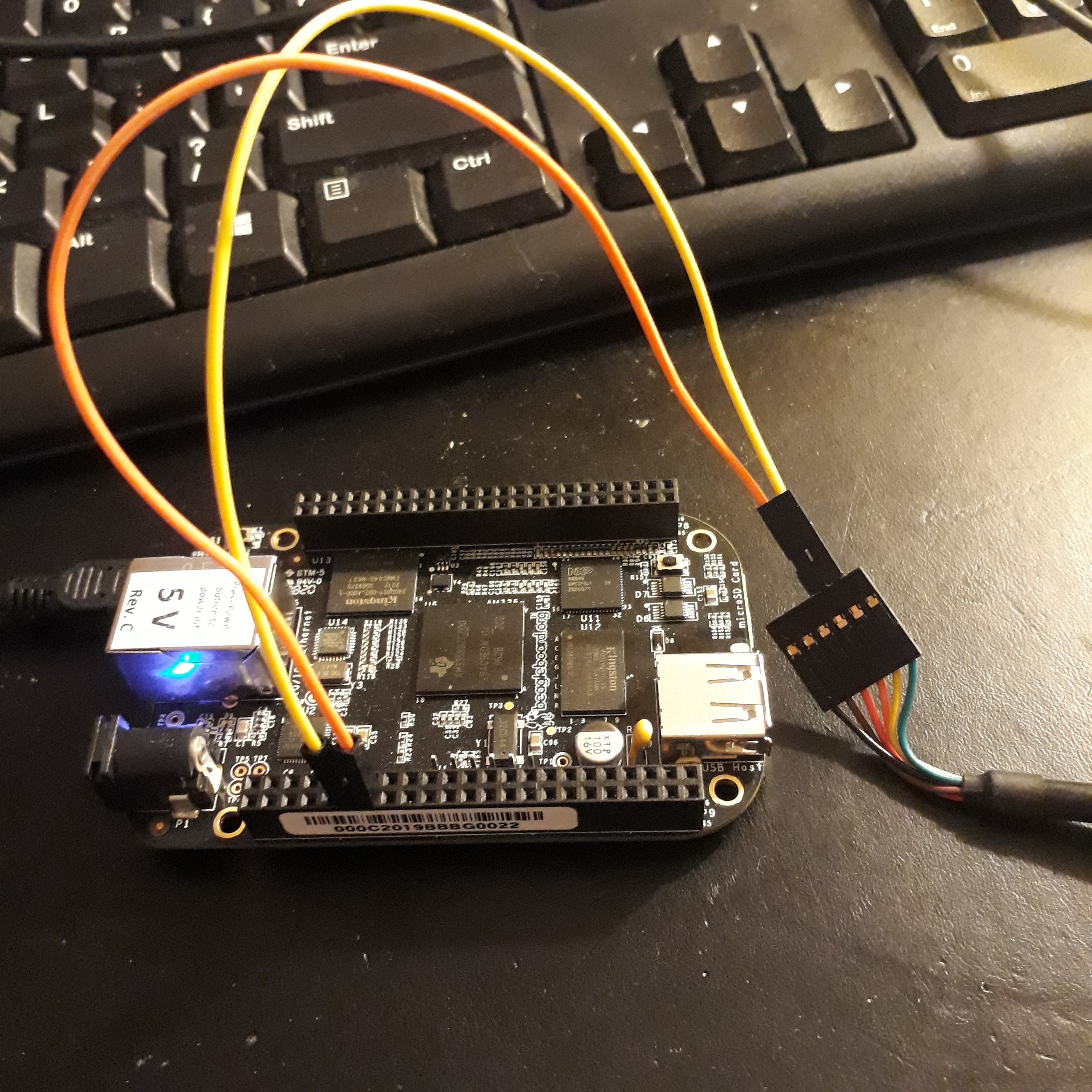
**One limitation in USB devices over VCP is: we cannot see the bootup process:**

**Step 4 (task4): Connecting the FTDI Rx/Tx with Beaglebone:**

P9.11 RXD - Orange TX

P9.13 TXD - Yellow RD

Note: Before connecting, you can check the RX/TX on the Beaglebone working or not by using loopback like in previouls Lab (Lab 5).



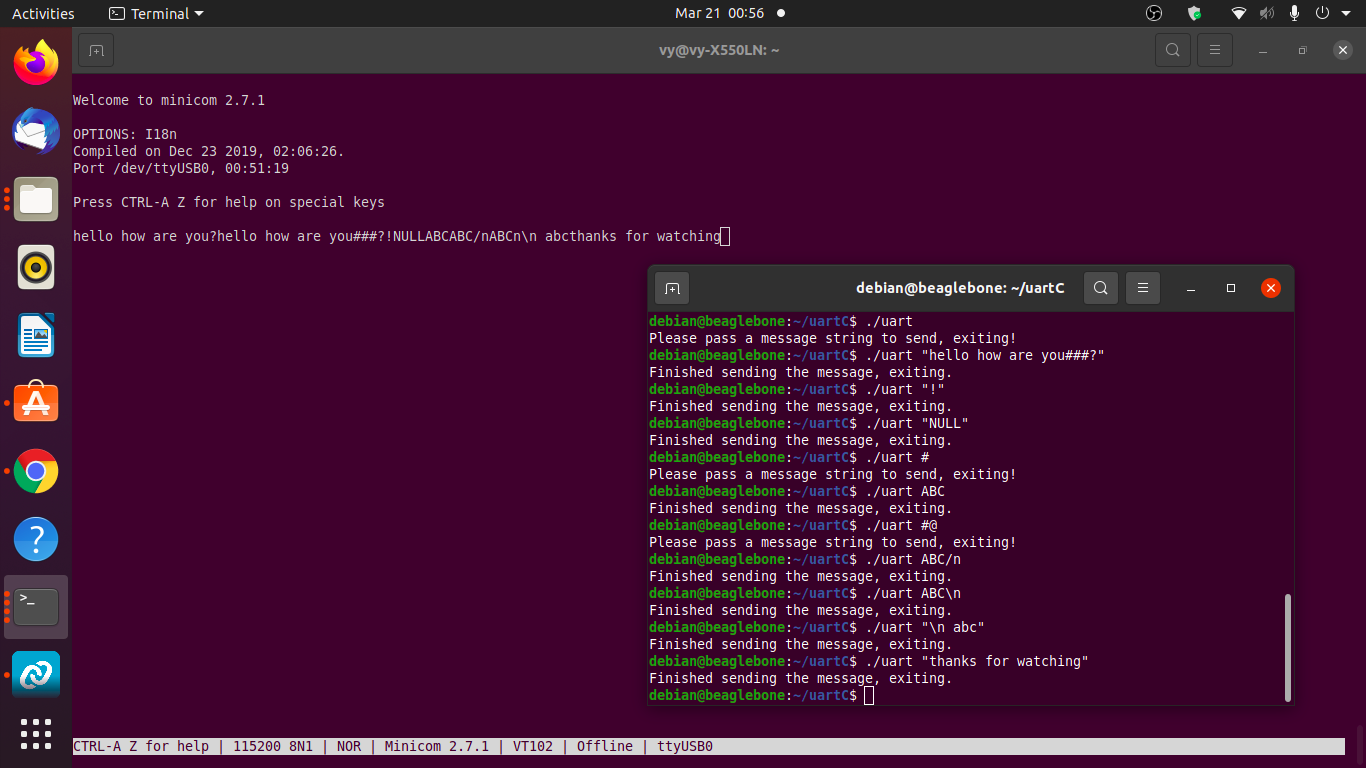
uart.c code can be found on Derek Molloy’s Github repository:

<https://github.com/derekmolloy/exploringBB/blob/version2/chp08/uart/uartC/uart.c>

Transfer the code from your host machine using stfp or create a c file directly on beaglebone.

|  |
| --- |
| /\* Simple send message example for communicating with the \* UART that is connected to a desktop PC. \*/ #include<stdio.h> #include<fcntl.h> #include<unistd.h> #include<termios.h> #include<string.h>  int main(int argc, char \*argv[]){  int file, count;  if(argc!=2){  printf("Please pass a message string to send, exiting!\n");  return -2;  }  if ((file = open("/dev/ttyO4", O\_RDWR | O\_NOCTTY | O\_NDELAY))<0){  perror("UART: Failed to open the device.\n");  return -1;  }  struct termios options;  tcgetattr(file, &options);  options.c\_cflag = B115200 | CS8 | CREAD | CLOCAL;  options.c\_iflag = IGNPAR | ICRNL;  tcflush(file, TCIFLUSH);  tcsetattr(file, TCSANOW, &options);   // send the string plus the null character  if ((count = write(file, argv[1], strlen(argv[1])+1))<0){  perror("UART: Failed to write to the output.\n");  return -1;  }  close(file);  printf("Finished sending the message, exiting.\n");  return 0; } |

**Step 5: Complying and running the execuable uart file, remember to add the message in the command:**

**Maximum data rate is 115200 bps over this USB VCPchannel.**

**The protocal is UART.**

<https://youtu.be/ZYtyOd0xJZg>