Setting up baremetal environment on Raspberry Pi 3

Resources

Software

- GNU ARM toolchain(provided in training material)
- minicom

Hardware

- Linux machine
- Raspberry Pi 3
- 3.3V USB to UART module
- Micro SD card, its adapter and card reader

Steps

Steps to compile bootloader and prepare SD card

1. Download and extract the training material tar ball "raspberry_pi3_baremetal.tar.bz2" given at this location
2. Go through the main README of "raspberrypi" repo for basic understanding
3. Try to compile bootloader07 at location "raspberrypi/boards/pi3/aarch32" using make command, this should create "kernel7.img"
    a. Copy of this bootloader is provided in "firmware" directory, in case one want to move ahead without compilation procedure
4. Flash the SD card as mentioned in main README
    a. All required firmware files to prepare SD card are available in "firmware" directory
    b. Format memory card as FAT32
    c. Place "bootcode.bin", "start.elf" and "kernel7.img"(prepared in step 3)

Steps to run bootloader and download application code in Raspberry Pi 3

1. Connect 3.3V USB to UART module on 40-pin connector of Raspberry Pi 3 as per connections mentioned below

Raspberri Pi 3 connector	USB to UART Module
Pin 10(Rx)	Tx
Pin 8(Tx)	Rx
Pin 6(GND)	GND

2. Connect the USB to UART module with Linux host machine
3. Open minicom on Linux host machine with following settings
    a. 115200 8N1, No flow control
4. Now insert the SD card(prepared using above steps) and power on the Raspberry Pi 3
5. You should see some commands on minicom ending with "IHEX"
6. At this point our bootloader is up an running on RPi 3 board and ready to accept the new firmware file
7. Try to send the precompiled hex file "firmware/uart_echo.hex" using minicom
    a. In minicom terminal go to command screen by pressing "Ctrl A" and "Z" key

    b. You should see (Send files...S) command there, press "S"
    c. Select "ascii" option
    d. Navigate through the hex file "firmware/uart_echo.hex"
8. After completing the transfer press "g" to jump to run downloaded application code
9. After pressing "g", if firmware is downloaded correctly then you should receive echo back of all commands send on USB to UART module

References for more details

- https://www.raspberrypi.org/forums/viewforum.php?f=72
- https://archive.fosdem.org/2017/schedule/event/programming_rpi3/attachments/slides/1475/export/events/attachments/programming_rpi3/slides/1475/bare_metal_rpi3.pdf
- https://en.wikibooks.org/wiki/Bare-metal_Raspberry_Pi_Programming
- https://github.com/dwelch67/raspberrypi
- http://www.valvers.com/open-software/raspberry-pi/step01-bare-metal-programming-in-cpt1/
- http://www.valvers.com/open-software/raspberry-pi/step02-bare-metal-programming-in-c-pt2/