Installing Linux for the FPGA board (DE1-SoC Board with ARMv7)

Resources used:

FPGA Manuals & Instructions (DE1-SoC rev.F): https://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&CategoryNo=205&No=836&PartNo=4#contents

Linux image: http://download.terasic.com/downloads/cd-rom/de1-soc/linux BSP/

(de1soc_lxde_1604.zip | | de1soc_ubuntu_1604.zip)

Procedure/Documentation:

The FPGA

- O. Device used: Altera DE1-SoC FPGA with ARMv7, University Program
- 1. Set the MSEL [4..0] to "00000" (to revert back to lab-mode: "10010")
- > NOTE: functionality of MSEL at 5 is yet to be discovered.
- 2. Plug in the power source, VGA, keyboard, mouse (optional), ethernet cable (optional)
- 3. Plug in the SD card with Linux to the the SD card slot
- 4. Press the power button to turn on the FPGA
- 5. As things should work, the FPGA will boot Linux and display it on the screen

The Operating System (Linux, with LXDE or Ubuntu)

1. Download the LXDE/Ubuntu image:

```
$ wget http://download.terasic.com/downloads/cd-rom/de1-
soc/linux_BSP/de1soc_ubuntu_1604.zip
```

or

```
$ wget http://download.terasic.com/downloads/cd-rom/de1-
soc/linux_BSP/de1soc_lxde_1604.zip
```

- > NOTE: The Ubuntu flavor is Lubuntu, with LXQt desktop instead of GNOME
- 2. Use an imager to install the image to SD Card (Here I use Raspberry Pi Imager AUR)
- 3. (Optional & Situational) Partition the disk, as the default storage space is not the entire SD card storage space.
- 4. (Optional) Set swap space

Results

Ubuntu trial

- The system booted up and worked as expected
- The OS featured: Lubuntu 16.04 LTS armv7l, an Ubuntu flavor that supports ARMv7 machines
- **Kernel**: 4.5.0, an old kernel version (compared to <u>Linux 6.0.3 on x86 64</u> or <u>Linux 5.15.61 on aarch64</u> (Raspberry Pi 400 ARMv8) at the time of writing this report).
- Desktop Environment: LXDE, with Openbox as the window manager
- RAM: 800MB
- Resolution: Support up to 1280x1024 VGA (see the User Manual, page 34)
- ISSUES:
 - The FPGA easily heats up, as it is not exactly made for PC usage
 - Limited RAM storage, so loading apps takes a lot of time
 - Limited app availability for the FPGA. This is weird considering the large pool of compatible apps for other ARM based computer.

Figures

Figure 1: Using fetch script (neofetch) to display system information



Figure 2: The setup. Here I'm browsing Google on Firefox, on the FPGA

(p/s: This was during a failed trial that I installed an old image of Ubuntu, so it came with GNOME 2 desktop)

Future experiments

- Use the FPGA as a personal computer:
 - + Watch a YouTube video
 - + Send an email
- Program the FGPA directly from the FPGA (maybe?)