

# Software Installation Tutorial for Linux for HaHa v3.0 Board

## Contents

Install GOWIN FPGA Designer .....	2
Install Atmel Toolchain and Programmer .....	9
Install Digilent Waveforms (Only if you have AD2 board) .....	16
References.....	20

**Instructor**: Dr. Swarup Bhunia

**Co-Instructor**: Reiner Dizon-Paradis

# Install GOWIN FPGA Designer

1. Get a free license to use GOWIN FPGA Designer using link [1]. Please note that this will take one or more business days. You will need your computer's Ethernet/Wi-Fi MAC address.
  - a. Use the net-tools library in Terminal: `ifconfig`
2. Navigate to the `~/Downloads`. Then, download the tar file [2]. If you would like the latest version, go to link [3] and create a GOWIN account.

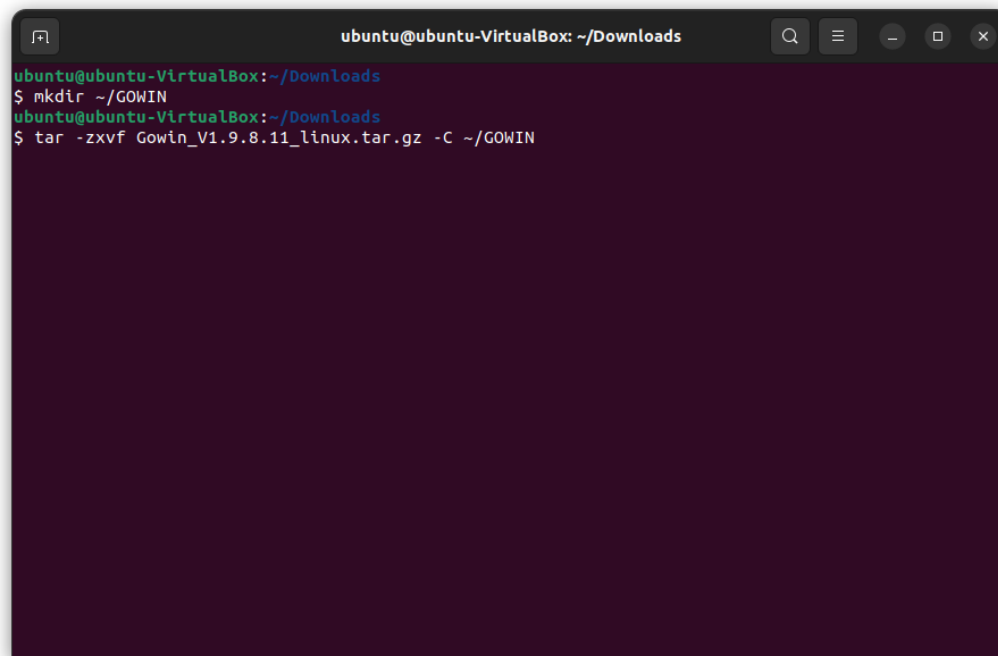
```
cd ~/Downloads/  
wget https://cdn.gowinsemi.com.cn/Gowin_V1.9.8.11_linux.tar.gz
```



```
ubuntu@ubuntu-VirtualBox: ~/Downloads  
ubuntu@ubuntu-VirtualBox:~  
$ cd ~/Downloads/  
ubuntu@ubuntu-VirtualBox:~/Downloads  
$ wget https://cdn.gowinsemi.com.cn/Gowin_V1.9.8.11_linux.tar.gz  
--2023-05-18 16:54:52-- https://cdn.gowinsemi.com.cn/Gowin_V1.9.8.11_linux.tar.gz  
Resolving cdn.gowinsemi.com.cn (cdn.gowinsemi.com.cn)... 23.90.190.178  
Connecting to cdn.gowinsemi.com.cn (cdn.gowinsemi.com.cn)[23.90.190.178]:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 417865308 (399M) [application/x-compressed]  
Saving to: 'Gowin_V1.9.8.11_linux.tar.gz'  
  
Gowin_V1.9.8.11_linux.t 100%[=====] 398.51M 27.2MB/s in 17s  
2023-05-18 16:55:09 (23.9 MB/s) - 'Gowin_V1.9.8.11_linux.tar.gz' saved [417865308/417865308]  
ubuntu@ubuntu-VirtualBox:~/Downloads  
$
```

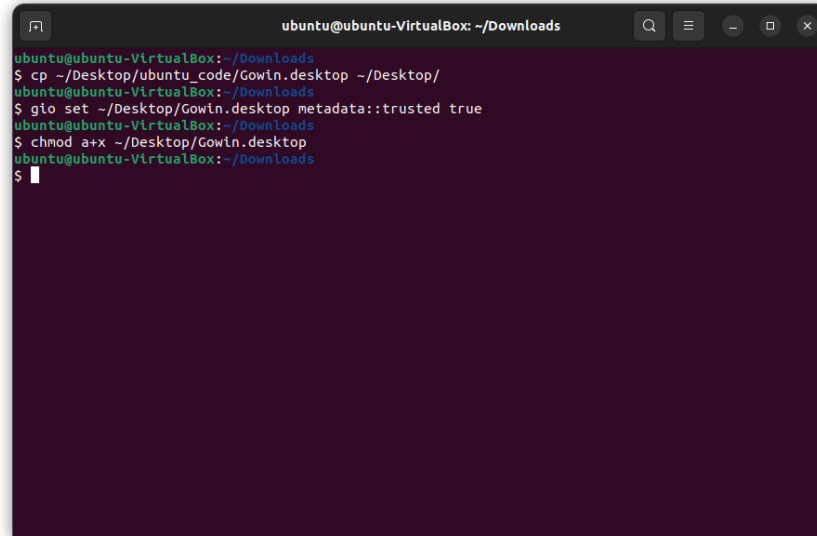
3. Create a directory in your home folder called *GOWIN*. Uncompress the tar file into that folder.

```
mkdir ~/GOWIN  
tar -zxvf Gowin_v1.9.8.11_linux.tar.gz -C ~/GOWIN
```

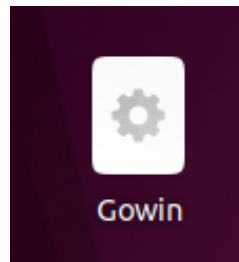


- 4. Copy the *Gowin.desktop* file from the *ubuntu\_code.zip* file to *~/Desktop*. Set the executable as trusted and add the execution permission.

```
cp <path/to/Gowin.desktop> ~/Desktop/  
gio set ~/Desktop/Gowin.desktop metadata::trusted true  
chmod a+x Gowin.desktop
```

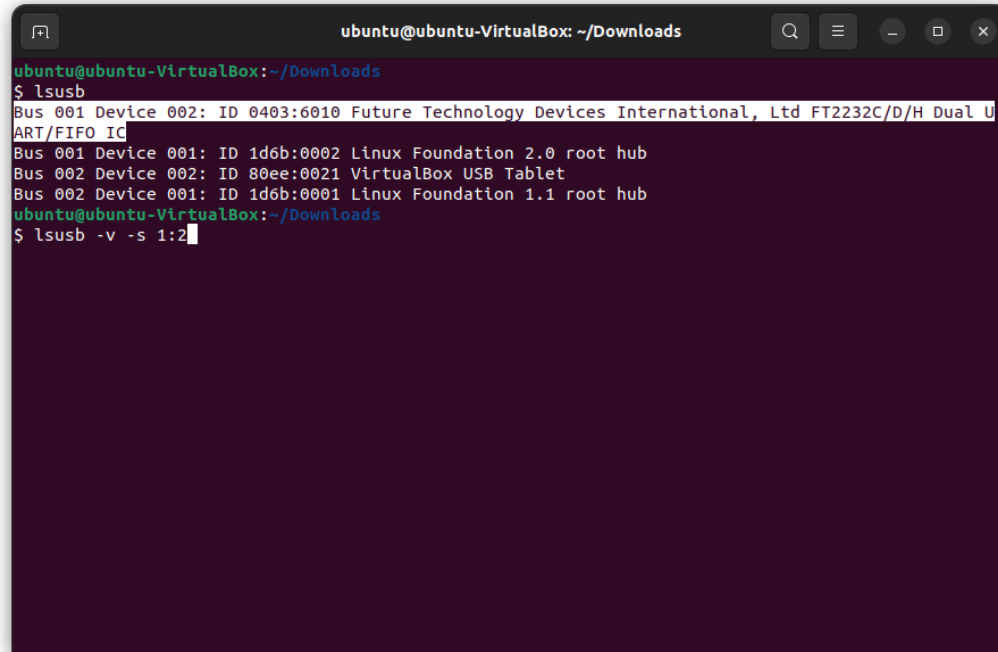


- 5. In your desktop, you will see this icon. You can use this to access the Gowin FPGA Designer software.



6. Plug in J14 of HaHa board to computer. If you are using VirtualBox, make sure to let the USB through the virtual machine. Run `lsusb` command. Look for "Future Technology Devices International, Ltd FT2232C/D/H Dual UART/FIFO IC" Take a note of the Bus and Device numbers for this USB Device.

### lsusb



```
ubuntu@ubuntu-VirtualBox: ~/Downloads
$ lsusb
Bus 001 Device 002: ID 0403:6010 Future Technology Devices International, Ltd FT2232C/D/H Dual UART/FIFO IC
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 002: ID 80ee:0021 VirtualBox USB Tablet
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
ubuntu@ubuntu-VirtualBox:~/Downloads
$ lsusb -v -s 1:2
```

- Run this `lsusb` command with extra flags. Replace the bus and device numbers. Take a note of the `idVendor` and `idProduct`.

`lsusb -v -s <bus no>:<device no>`

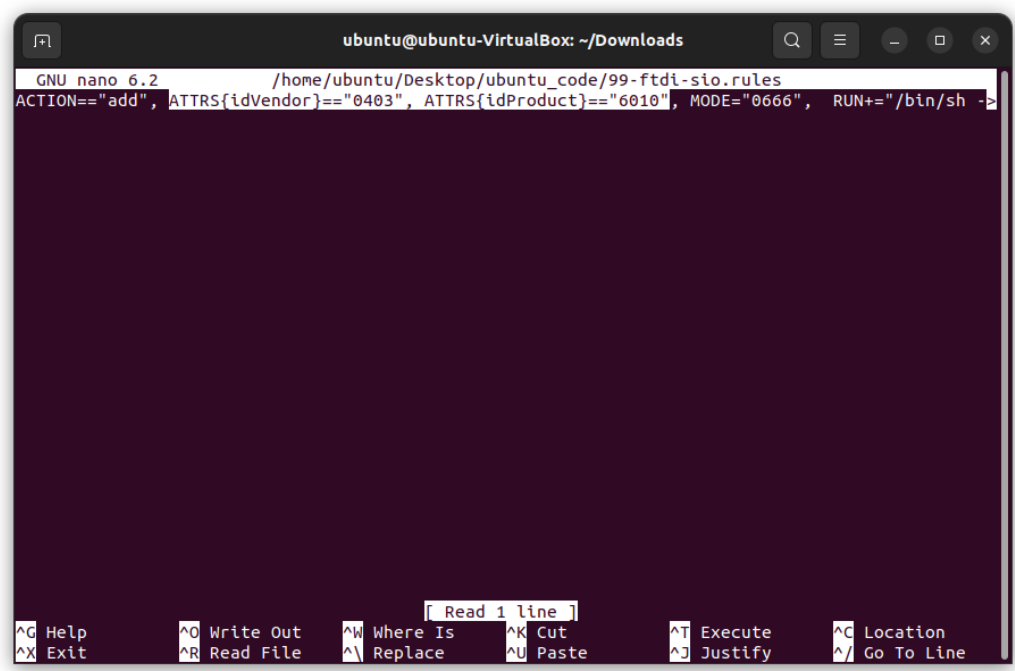
```

ubuntu@ubuntu-VirtualBox: ~/Downloads
$ lsusb -v -s 1:2

Bus 001 Device 002: ID 0403:6010 Future Technology Devices International, Ltd FT2232C/D/H Dual UART/FIFO IC
Couldn't open device, some information will be missing
Device Descriptor:
  bLength                18
  bDescriptorType        1
  bcdUSB                  2.00
  bDeviceClass            0
  bDeviceSubClass        0
  bDeviceProtocol        0
  bMaxPacketSize0       64
  idVendor                0x0403 Future Technology Devices International, Ltd
  idProduct              0x6010 FT2232C/D/H Dual UART/FIFO IC
  bcdDevice               7.00
  iManufacturer          1 FTDI
  iProduct               2 Dual RS232-HS
  iSerial                0
  bNumConfigurations     1
Configuration Descriptor:
  bLength                9
  bDescriptorType        2
  wTotalLength           0x0037
  bNumInterfaces         2
  bConfigurationValue    1
  iConfiguration        0
  bmAttributes           0x80
  
```

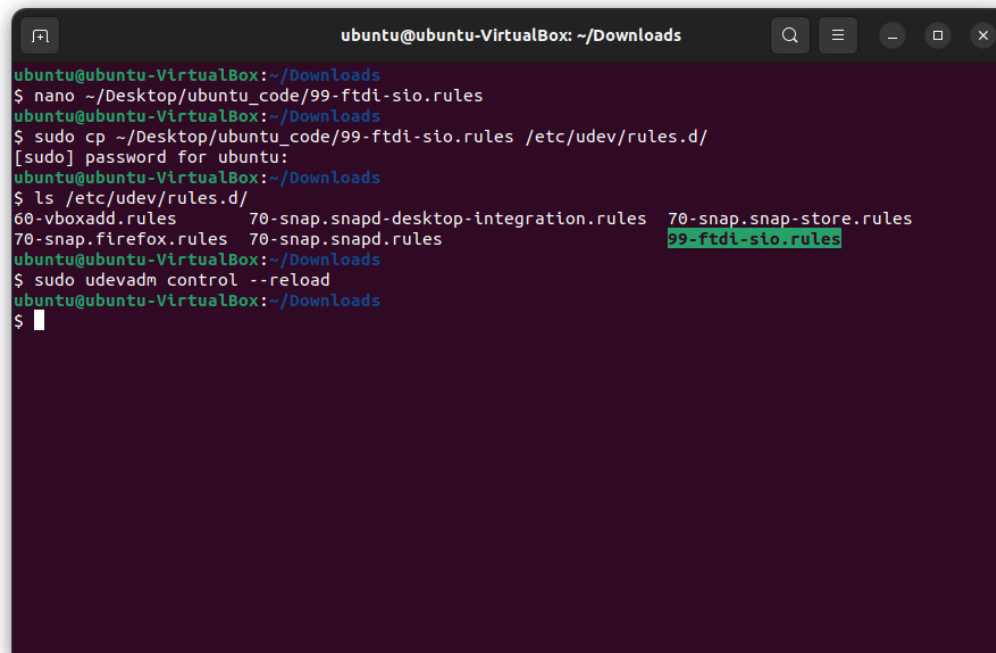
- 8. Look for the 99-ftdi-sio.rules file inside the ubuntu\_code.zip file. Edit the Rules file using Nano to plug in the *idVendor* and *idProduct*. Note: The file may already contain the correct values. In this case, please double check the numbers.

nano <path/to/99-ftdi-sio.rules>



- Copy the modified rules file to the Rules directory. Then, enable the USB rule for the GOWIN FPGA. Reboot your machine.

```
sudo cp <path/to/99-ftdi-sio.rules> /etc/udev/rules.d/  
sudo udevadm control --reload  
sudo reboot
```

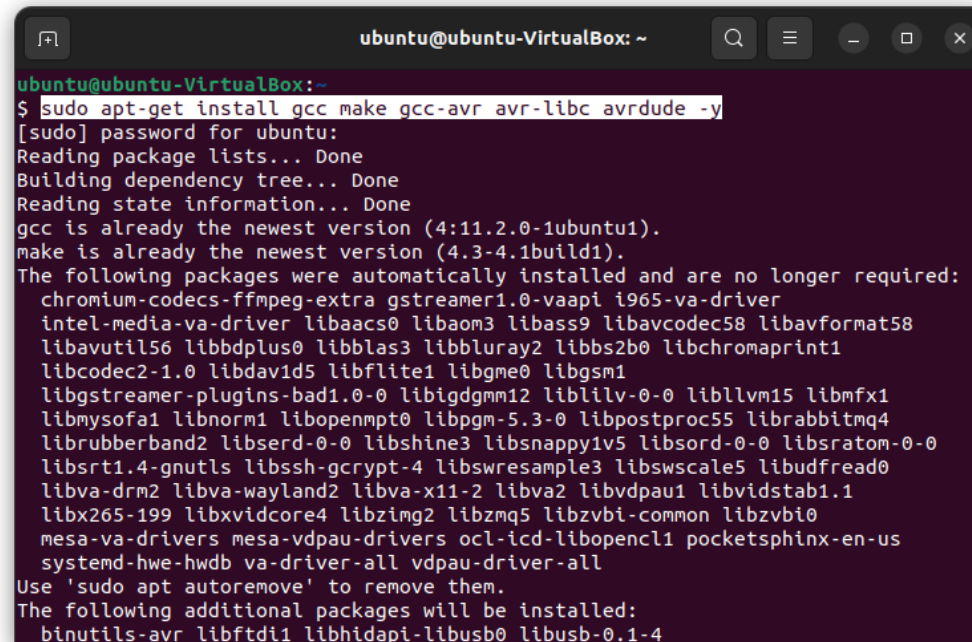




# Install Atmel Toolchain and Programmer

1. Install Atmel Toolchain and Programmer. That's it. The rest of the steps are additional guide.

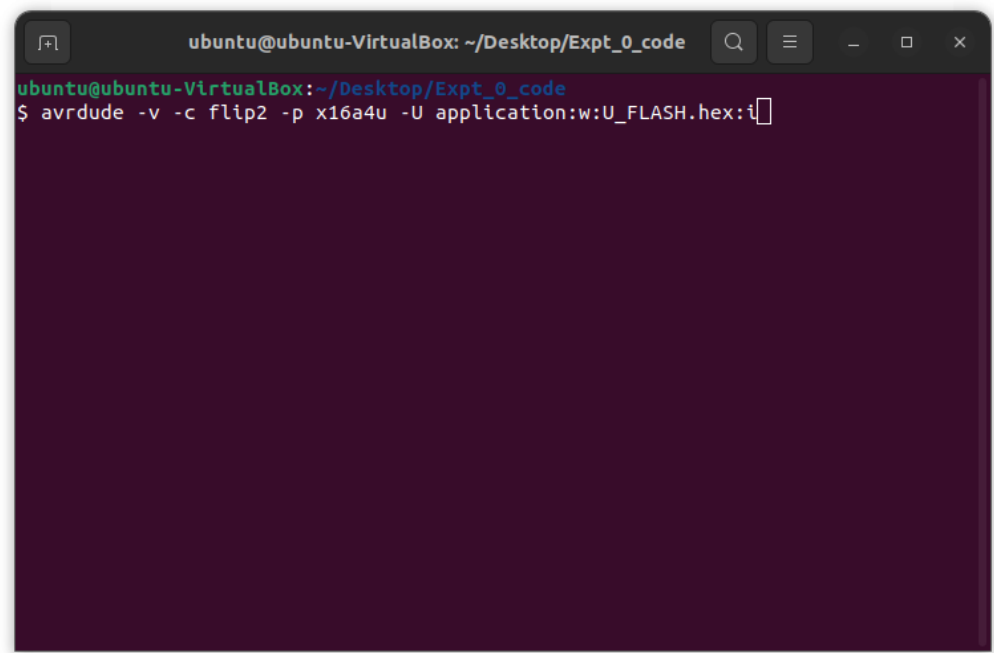
```
sudo apt-get install gcc make gcc-avr avr-libc avrdude -y
```



```
ubuntu@ubuntu-VirtualBox: ~  
ubuntu@ubuntu-VirtualBox:~  
$ sudo apt-get install gcc make gcc-avr avr-libc avrdude -y  
[sudo] password for ubuntu:  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
gcc is already the newest version (4:11.2.0-1ubuntu1).  
make is already the newest version (4.3-4.1build1).  
The following packages were automatically installed and are no longer required:  
chromium-codecs-ffmpeg-extra gstreamer1.0-vaapi i965-va-driver  
intel-media-va-driver libaacs0 libaom3 libass9 libavcodec58 libavformat58  
libavutil56 libbdplus0 libblas3 libbluray2 libbs2b0 libchromaprint1  
libcodec2-1.0 libdav1d5 libflite1 libgme0 libgsm1  
libgstreamer-plugins-bad1.0-0 libigdgmm12 liblilv-0-0 libllvm15 libmfx1  
libmysofa1 libnorm1 libopenmpt0 libpgm-5.3-0 libpostproc55 librabbitmq4  
librubberband2 libserd-0-0 libshine3 libsnappy1v5 libsord-0-0 libstratom-0-0  
libsrt1.4-gnutls libssh-gcrypt-4 libswresample3 libswscale5 libudfread0  
libva-drm2 libva-wayland2 libva-x11-2 libva2 libvdpau1 libvidstab1.1  
libx265-199 libxvidcore4 libzing2 libzmq5 libzvi-common libzvi0  
mesa-va-drivers mesa-va-drivers ocl-icd-libopencl1 pocketchinx-en-us  
systemd-hwe-hwdb va-driver-all vdpau-driver-all  
Use 'sudo apt autoremove' to remove them.  
The following additional packages will be installed:  
binutils-avr libftdi1 libhidapi-libusb0 libusb-0.1-4
```

2. Make sure your HaHa v3.0 board is in programming mode. You can program using the command:

```
avrdude -v -c flip2 -p x16a4u -U application:w:<path/to/hex/file>:i
```



3. The program is successfully installed if it shows the following:

```
ubuntu@ubuntu-VirtualBox: ~/Desktop/Expt_0_code
avrdude: NOTE: "application" memory has been specified, an erase cycle will be performed
      To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "U_FLASH.hex"
avrdude: writing application (1272 bytes):

Writing | ##### | 100% 0.32s

avrdude: 1272 bytes of application written
avrdude: verifying application memory against U_FLASH.hex:
avrdude: load data application data from input file U_FLASH.hex:
avrdude: input file U_FLASH.hex contains 1272 bytes
avrdude: reading on-chip application data:

Reading | ##### | 100% 0.26s

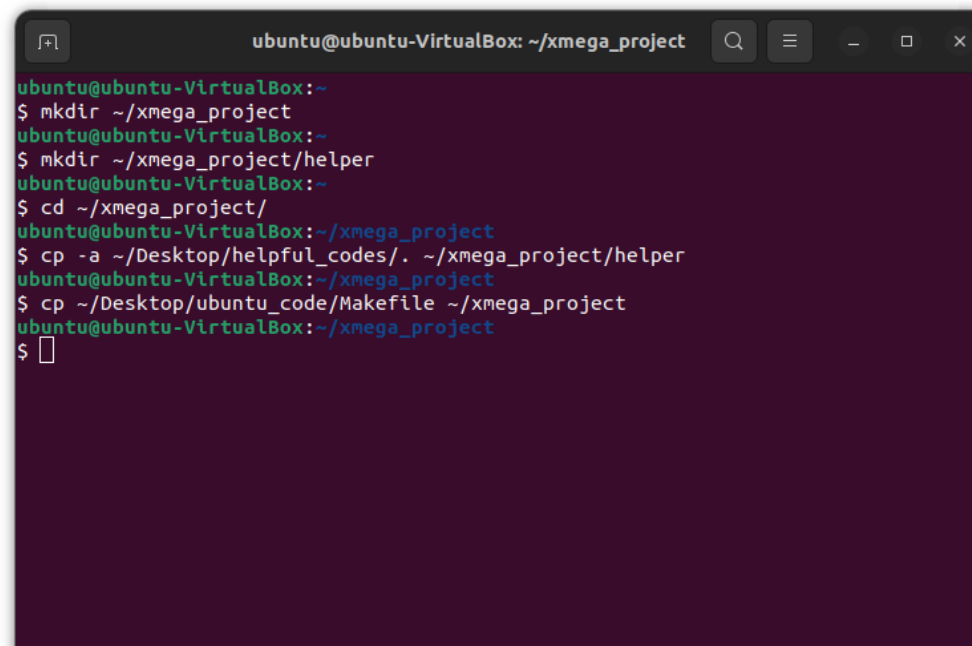
avrdude: verifying ...
avrdude: 1272 bytes of application verified

avrdude done. Thank you.

ubuntu@ubuntu-VirtualBox:~/Desktop/Expt_0_code
$
```

4. You can create a Xmega project folder wherever you like. Then, copy the Xmega helpful code inside the *helper/* directory inside the project folder. Copy the makefile inside the project folder.

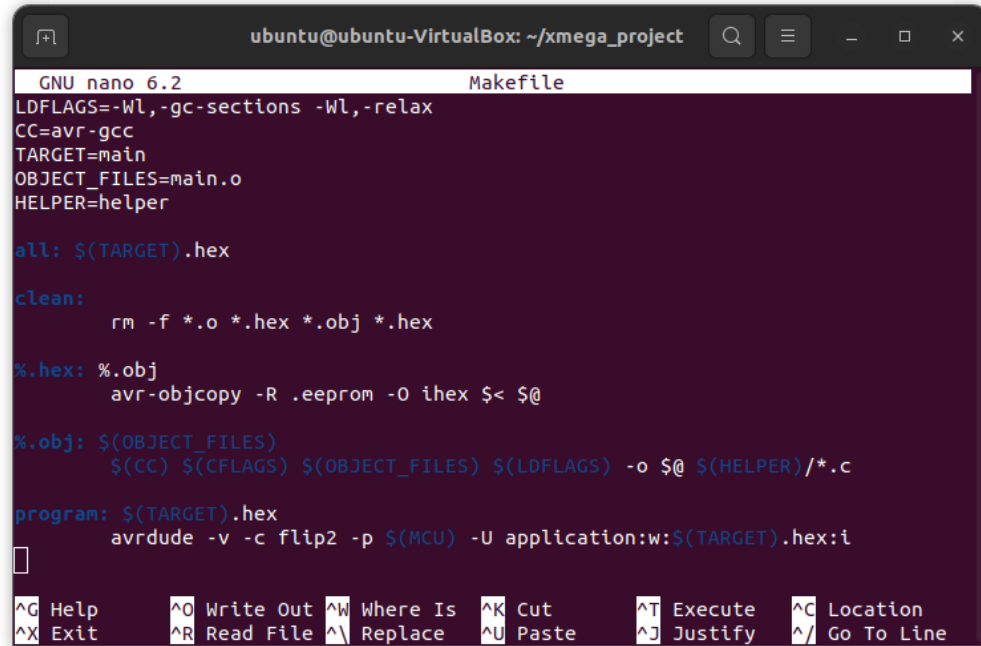
```
mkdir ~/xmega_project
mkdir ~/xmega_project/helper
cd ~/xmega_project/
cp -a <path/to/helpful/codes>/ . ~/xmega_project/helper
cp <path/to/Makefile> ~/xmega_project/
```



```
ubuntu@ubuntu-VirtualBox: ~/xmega_project
ubuntu@ubuntu-VirtualBox:~$ mkdir ~/xmega_project
ubuntu@ubuntu-VirtualBox:~$ mkdir ~/xmega_project/helper
ubuntu@ubuntu-VirtualBox:~$ cd ~/xmega_project/
ubuntu@ubuntu-VirtualBox:~/xmega_project$ cp -a ~/Desktop/helpful_codes/. ~/xmega_project/helper
ubuntu@ubuntu-VirtualBox:~/xmega_project$ cp ~/Desktop/ubuntu_code/Makefile ~/xmega_project
ubuntu@ubuntu-VirtualBox:~/xmega_project$
```

- 5. After this, make your code according to the experiment instructions. Then, make changes to the makefile as needed with new code.

### nano Makefile



```
GNU nano 6.2 Makefile
LD_FLAGS=-Wl,-gc-sections -Wl,-relax
CC=avr-gcc
TARGET=main
OBJECT_FILES=main.o
HELPER=helper

all: $(TARGET).hex

clean:
    rm -f *.o *.hex *.obj *.hex

%.hex: %.obj
    avr-objcopy -R .eeprom -O ihex $< $@

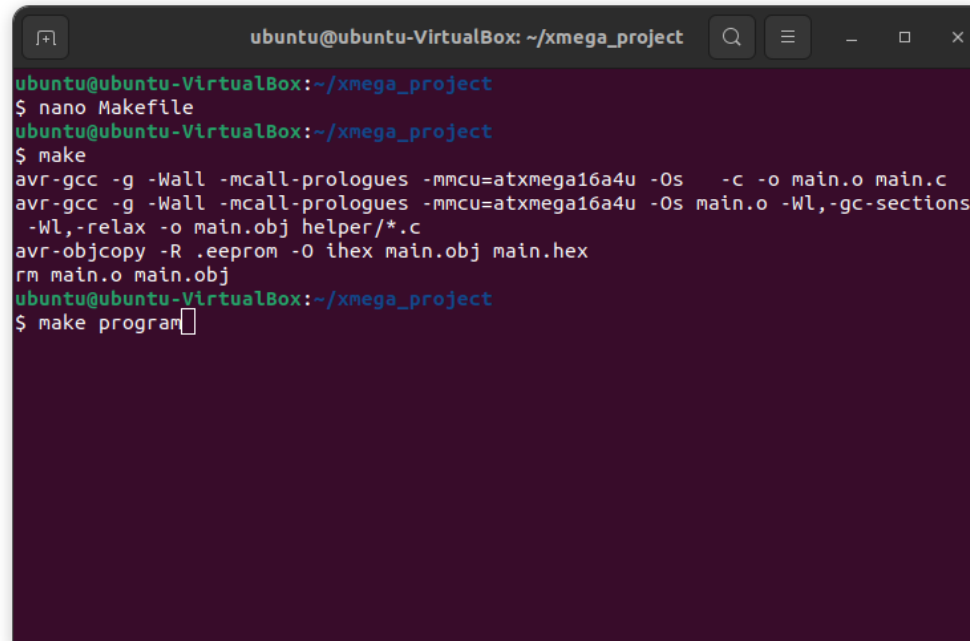
%.obj: $(OBJECT_FILES)
    $(CC) $(CFLAGS) $(OBJECT_FILES) $(LD_FLAGS) -o $@ $(HELPER)/*.c

program: $(TARGET).hex
    avrdude -v -c flip2 -p $(MCU) -U application:w:$(TARGET).hex:i
```

^G Help    ^O Write Out    ^W Where Is    ^K Cut    ^T Execute    ^C Location  
^X Exit    ^R Read File    ^\ Replace    ^U Paste    ^J Justify    ^\_ Go To Line

- You can create the hex file using the `make` command. Then, you can program using `make program` command.

make  
make program



```
ubuntu@ubuntu-VirtualBox: ~/xmega_project
ubuntu@ubuntu-VirtualBox:~/xmega_project
$ nano Makefile
ubuntu@ubuntu-VirtualBox:~/xmega_project
$ make
avr-gcc -g -Wall -mcall-prologues -mmcu=atxmega16a4u -Os -c -o main.o main.c
avr-gcc -g -Wall -mcall-prologues -mmcu=atxmega16a4u -Os main.o -Wl,-gc-sections
-Wl,-relax -o main.obj helper/*.c
avr-objcopy -R .eeprom -O ihex main.obj main.hex
rm main.o main.obj
ubuntu@ubuntu-VirtualBox:~/xmega_project
$ make program
```

7. The program is successfully installed if it shows the following:

make  
make program

```
ubuntu@ubuntu-VirtualBox: ~/xmega_project
avrdude: NOTE: "application" memory has been specified, an erase cycle will be performed
      To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "main.hex"
avrdude: writing application (1776 bytes):

Writing | ##### | 100% 0.37s

avrdude: 1776 bytes of application written
avrdude: verifying application memory against main.hex:
avrdude: load data application data from input file main.hex:
avrdude: input file main.hex contains 1776 bytes
avrdude: reading on-chip application data:

Reading | ##### | 100% 0.44s

avrdude: verifying ...
avrdude: 1776 bytes of application verified

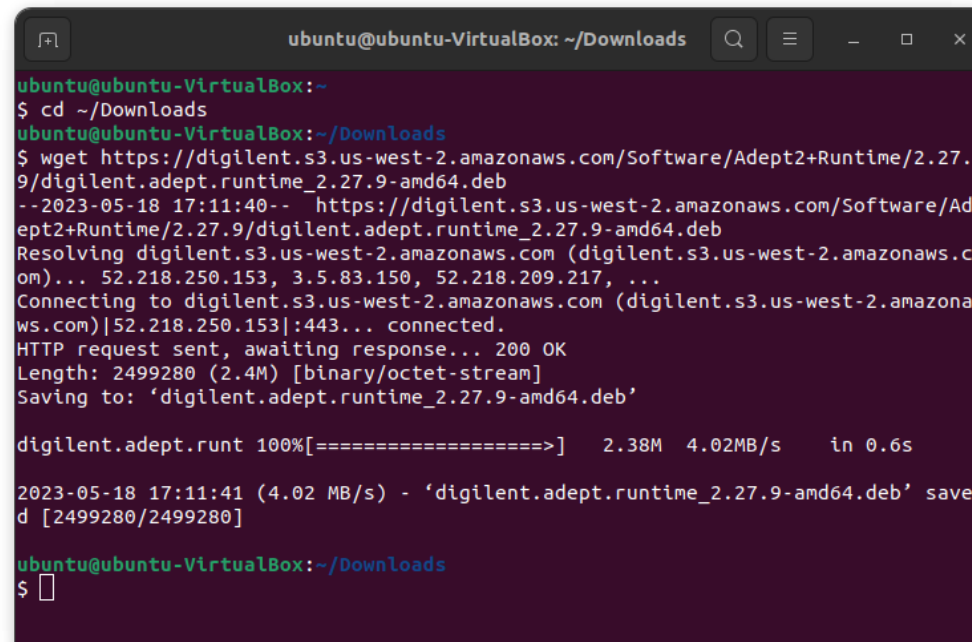
avrdude done. Thank you.

ubuntu@ubuntu-VirtualBox:~/xmega_project
$
```

# Install Digilent Waveforms (Only if you have AD2 board)

1. Navigate to the `~/Downloads`. Then, download the deb file for Adept Runtime [4]. If you would like the latest version, go to link [5].

```
cd ~/Downloads/  
wget https://digilent.s3.us-west-2.amazonaws.com/Software/Adept2+Runtime/2.27.9/digilent.adept.runtime_2.27.9-amd64.deb
```

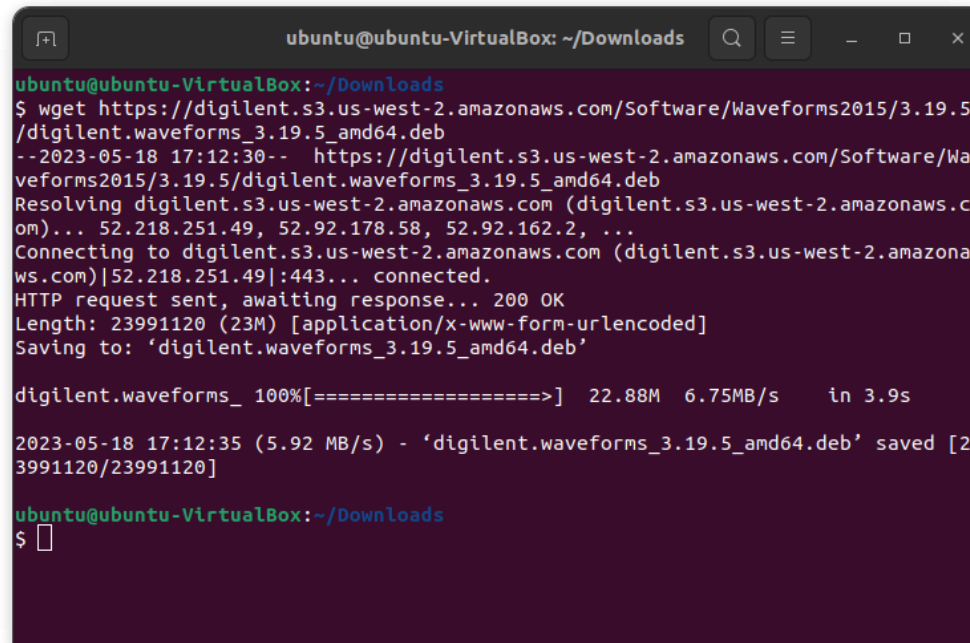


```
ubuntu@ubuntu-VirtualBox: ~/Downloads  
ubuntu@ubuntu-VirtualBox:~  
$ cd ~/Downloads  
ubuntu@ubuntu-VirtualBox:~/Downloads  
$ wget https://digilent.s3.us-west-2.amazonaws.com/Software/Adept2+Runtime/2.27.9/digilent.adept.runtime_2.27.9-amd64.deb  
--2023-05-18 17:11:40-- https://digilent.s3.us-west-2.amazonaws.com/Software/Adept2+Runtime/2.27.9/digilent.adept.runtime_2.27.9-amd64.deb  
Resolving digilent.s3.us-west-2.amazonaws.com (digilent.s3.us-west-2.amazonaws.com)... 52.218.250.153, 3.5.83.150, 52.218.209.217, ...  
Connecting to digilent.s3.us-west-2.amazonaws.com (digilent.s3.us-west-2.amazonaws.com)|52.218.250.153|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 2499280 (2.4M) [binary/octet-stream]  
Saving to: 'digilent.adept.runtime_2.27.9-amd64.deb'  
  
digilent.adept.runt 100%[=====] 2.38M 4.02MB/s in 0.6s  
  
2023-05-18 17:11:41 (4.02 MB/s) - 'digilent.adept.runtime_2.27.9-amd64.deb' saved [2499280/2499280]  
  
ubuntu@ubuntu-VirtualBox:~/Downloads  
$
```



2. Download the deb file for Waveforms [6]. If you would like the latest version, go to link [7].

```
wget https://digilent.s3.us-west-2.amazonaws.com/Software/Waveforms2015/3.19.5/digilent.waveforms_3.19.5_amd64.deb
```



```
ubuntu@ubuntu-VirtualBox: ~/Downloads
ubuntu@ubuntu-VirtualBox:~/Downloads
$ wget https://digilent.s3.us-west-2.amazonaws.com/Software/Waveforms2015/3.19.5/digilent.waveforms_3.19.5_amd64.deb
--2023-05-18 17:12:30-- https://digilent.s3.us-west-2.amazonaws.com/Software/Waveforms2015/3.19.5/digilent.waveforms_3.19.5_amd64.deb
Resolving digilent.s3.us-west-2.amazonaws.com (digilent.s3.us-west-2.amazonaws.com)... 52.218.251.49, 52.92.178.58, 52.92.162.2, ...
Connecting to digilent.s3.us-west-2.amazonaws.com (digilent.s3.us-west-2.amazonaws.com)|52.218.251.49|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 23991120 (23M) [application/x-www-form-urlencoded]
Saving to: 'digilent.waveforms_3.19.5_amd64.deb'

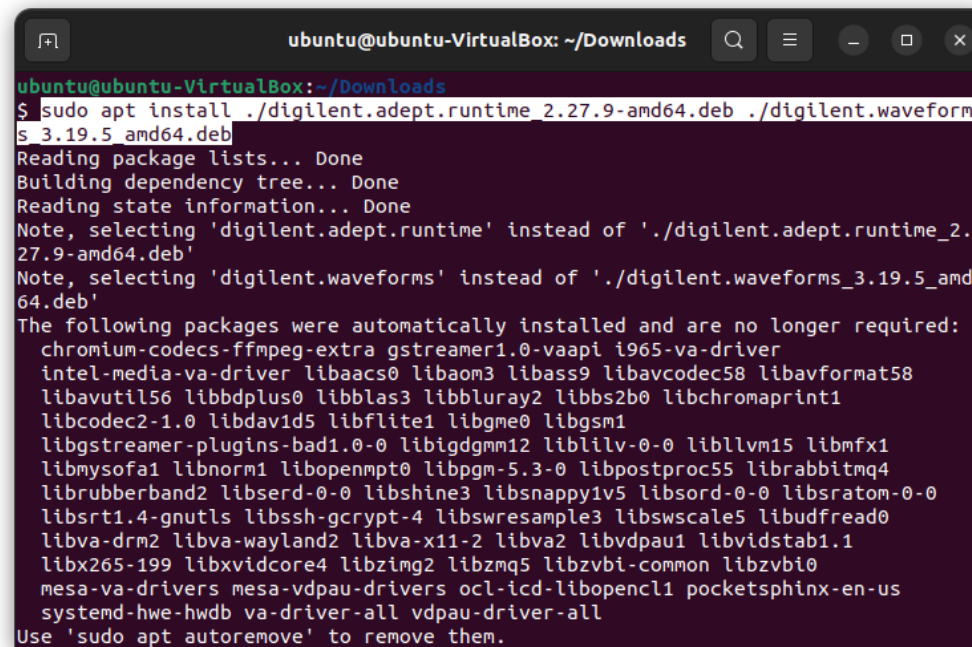
digilent.waveforms_ 100%[=====] 22.88M 6.75MB/s in 3.9s

2023-05-18 17:12:35 (5.92 MB/s) - 'digilent.waveforms_3.19.5_amd64.deb' saved [23991120/23991120]

ubuntu@ubuntu-VirtualBox:~/Downloads
$
```

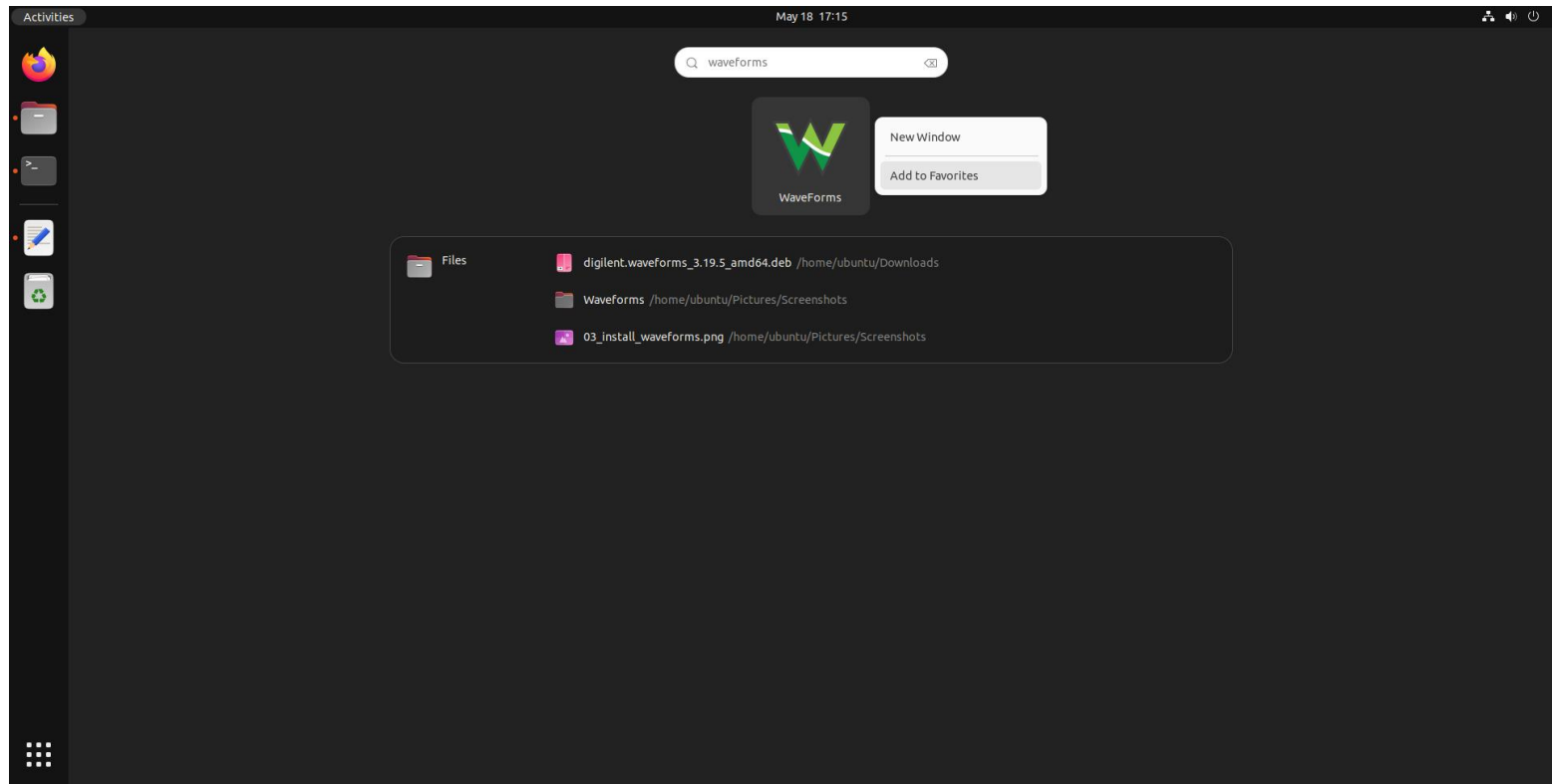
3. Install Waveforms using this command:

```
sudo apt install ./digilent.adept.runtime_2.27.9-  
amd64.deb ./digilent.waveforms_3.19.5_amd64.deb -y
```



```
ubuntu@ubuntu-VirtualBox: ~/Downloads
ubuntu@ubuntu-VirtualBox:~/Downloads
$ sudo apt install ./digilent.adept.runtime_2.27.9-amd64.deb ./digilent.waveform
$ 3.19.5 amd64.deb
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'digilent.adept.runtime' instead of './digilent.adept.runtime_2.
27.9-amd64.deb'
Note, selecting 'digilent.waveforms' instead of './digilent.waveforms_3.19.5_amd
64.deb'
The following packages were automatically installed and are no longer required:
chromium-codecs-ffmpeg-extra gstreamer1.0-vaapi i965-va-driver
intel-media-va-driver libaac3 libaom3 libass9 libavcodec58 libavformat58
libavutil56 libbdplus0 libblas3 libbluray2 libbs2b0 libchromaprint1
libcodec2-1.0 libdavid5 libflite1 libgme0 libgsm1
libgstreamer-plugins-bad1.0-0 libigdgmm12 liblilv-0-0 libllvm15 libmfx1
libmysofa1 libnorm1 libopenmpt0 libpgm-5.3-0 libpostproc55 librabbitmq4
librubberband2 libserd-0-0 libshine3 libsnappy1v5 libsord-0-0 libstratom-0-0
libstr1.4-gnutls libssh-gcrypt-4 libswresample3 libswscale5 libudfread0
libva-drm2 libva-wayland2 libva-x11-2 libva2 libvdpau1 libvidstab1.1
libx265-199 libxvidcore4 libzimg2 libzmq5 libzvi-common libzvi0
mesa-va-drivers mesa-va-drivers ocl-icd-libopencl1 pocketsphinx-en-us
systemd-hwe-hwdb va-driver-all vdpau-driver-all
Use 'sudo apt autoremove' to remove them.
```

4. The installation is complete. You can search for it in your machine.



## References

- [1] <https://www.gowinsemi.com/en/support/license/>
- [2] [https://cdn.gowinsemi.com.cn/Gowin\\_V1.9.8.11\\_linux.tar.gz](https://cdn.gowinsemi.com.cn/Gowin_V1.9.8.11_linux.tar.gz)
- [3] [https://www.gowinsemi.com/en/support/download\\_eda/](https://www.gowinsemi.com/en/support/download_eda/)
- [4] [https://digilent.s3.us-west-2.amazonaws.com/Software/Adept2+Runtime/2.27.9/digilent.adept.runtime\\_2.27.9-amd64.deb](https://digilent.s3.us-west-2.amazonaws.com/Software/Adept2+Runtime/2.27.9/digilent.adept.runtime_2.27.9-amd64.deb)
- [5] <https://lp.digilent.com/complete-adept-runtime-download>
- [6] [https://digilent.s3.us-west-2.amazonaws.com/Software/Waveforms2015/3.19.5/digilent.waveforms\\_3.19.5\\_amd64.deb](https://digilent.s3.us-west-2.amazonaws.com/Software/Waveforms2015/3.19.5/digilent.waveforms_3.19.5_amd64.deb)
- [7] <https://lp.digilent.com/complete-waveforms-download>