

*****Draft*****
pibuild
reference to github notes
07/10/21
*****Draft*****

This micro sd will run on Rpi3 a Rpi4 4Gb or Rpi4 8Gb PI400 will provide better results.

The standard software provides a lot of software VLC. Libreoffice, Editors, Image viewers, GIMP. Videos can be download and play using VLC.

This version provides the software to learn about FPGAs and Bare Metal by adding 481 packages and several compiled C & Python Tools (autofpga, icespice, nextpnr, yosys, zipcpu, and verilator). In addition the dependencies (libgtk2.0-dev libcairo2-dev libpango1.0-dev libgdk-pixbuf2.0-dev libatk1.0-dev libghc-x11-dev) for Lazarus IDE (Ultibo-Edition). With https://github.com/develone/Ultibo_Projects you can compile Bare Metal applications for Rpi, Rpi2, Rpi3, and RPi4.

After connecting a keyboard/mouse, usb pwr, supply and HDMI TV. The system will boot to Login screen. These are a few screenshots of a Raspberry Pi that I recently built. This version has additional software installed to support Bare Metal development. This also has the software to do Synthesis of Verilog Code using Yosys. yosys -V

Yosys 0.9+4052 (git sha1 0ccc7229, gcc 8.3.0-6+rpi1 -fPIC -Os). The Synthesis output needs to be Placed & Routed using nextpnr. nextpnr-ice40 -V

nextpnr-ice40 -- Next Generation Place and Route (Version 4419c36d). The output of nextpnr creates a catzip.bin bitstream which can be used to program the FPGA. The catzip.bin needs to be tested if the clock frequency is not too high using icespice. icetime -d hx8k -c 40 catzip.asc

// Reading input .asc file..

// Reading 8k chipdb file..

// Creating timing netlist..

// Timing estimate: 18.37 ns (54.44 MHz)

// Checking 25.00 ns (40.00 MHz) clock constraint: PASSED.

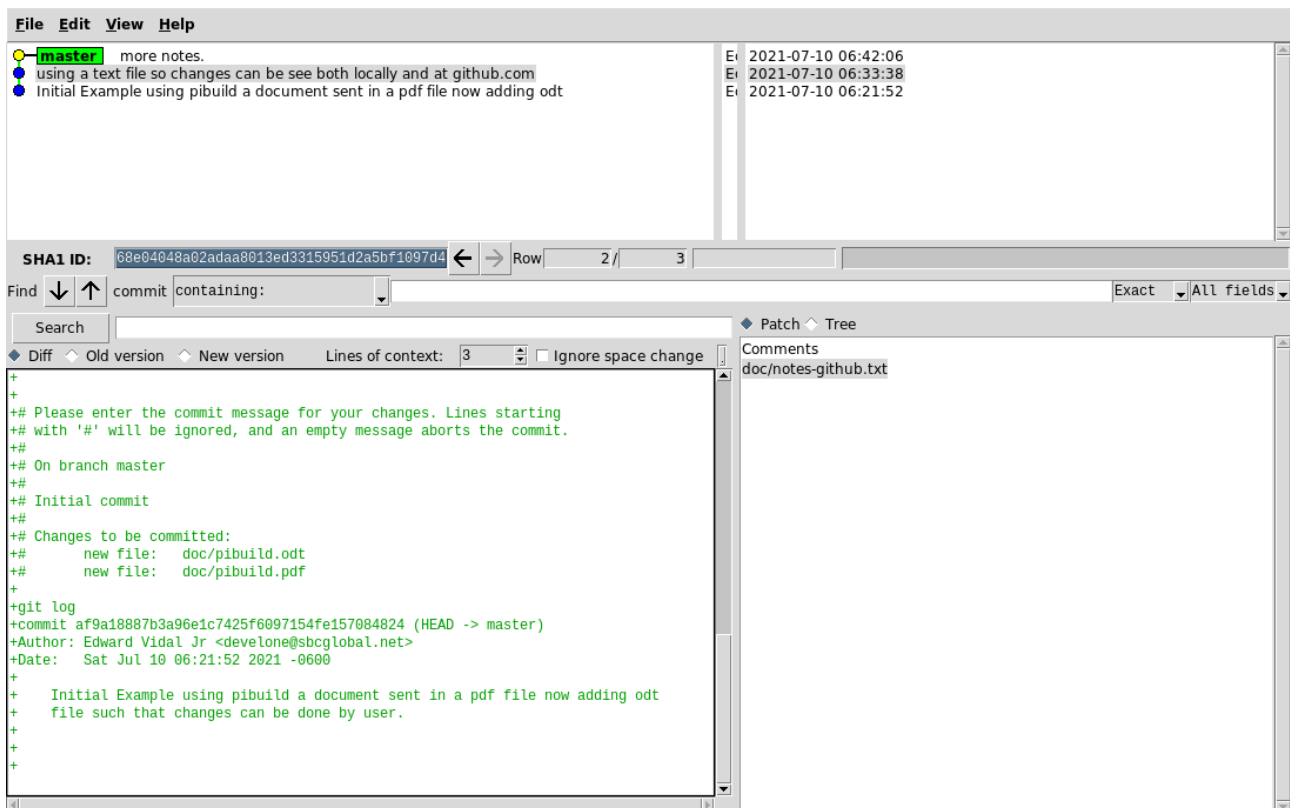
These are software packages needed to work with FPGAs autofpga, icespice, nextpnr, yosys, zipcpu, and verilator.



On the Rpi you can gitk & to see the changes made to local repo.



After a more changes.



After entering the correct passwd user devel will be logged in to DeskTop.



Users can use ssh to make remote connection to the newly created Raspberry Pi.

```
File Edit Tabs Help
devel@mypi3-20:~ $ ssh -Y tatiana
devel@tatiana's password:
Linux tatiana 5.10.17-v7+ #1421 SMP Thu May 27 13:59:01 BST 2021 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Jul  8 21:23:01 2021 from 192.168.1.245
devel@tatiana:~ $ uname -a
Linux tatiana 5.10.17-v7+ #1421 SMP Thu May 27 13:59:01 BST 2021 armv7l GNU/Linux
devel@tatiana:~ $
```

This does require enabling using Preferences/Raspberry Pi Configuration.

System	Display	Interfaces	Performance	Localisation
Camera:	<input checked="" type="radio"/> Enable	<input type="radio"/> Disable		
SSH:	<input checked="" type="radio"/> Enable	<input type="radio"/> Disable		
VNC:	<input type="radio"/> Enable	<input checked="" type="radio"/> Disable		
SPI:	<input checked="" type="radio"/> Enable	<input type="radio"/> Disable		
I2C:	<input checked="" type="radio"/> Enable	<input type="radio"/> Disable		
Serial Port:	<input checked="" type="radio"/> Enable	<input type="radio"/> Disable		
Serial Console:	<input checked="" type="radio"/> Enable	<input type="radio"/> Disable		
1-Wire:	<input type="radio"/> Enable	<input checked="" type="radio"/> Disable		
Remote GPIO:	<input type="radio"/> Enable	<input checked="" type="radio"/> Disable		

Cancel OK

This is also where the Hostname and if a prompt is required to Log in is performed.

I also have a simulation of the software in the FPGA.

To start the simulator with no trace ./arm-main.tb is run ~/pi400/catzip/sim/verilated if the -d option is used the VCD file trace.vcd is generated.

Shell 1

```
./arm-main_tb
Listening on port 8363
Listening on port 8364
> T
> Z
Accepted CMD connection
< A01000015R
> A01000015R20210707
< [CLOSED]
```

Shell 2

```
./arm-wbregs version
01000014 ( VERSION) : [!..] 20210707
```

The VCD file was created by using the simulator with -d option and the command `./arm-wbregs` version

