**********Draft**********

Testing Upgrades to icestorm, yosys, zipcpu and nextpnr
zipcpu, autofpga & verilator
icozip catzip-dev branch
mypi3-20
10/13/21

**********Draft**********

Chg'ed to 48MHz instead of 40Mhz. Now the values of Q & F are what is provided by icepll. The first build got a Bus Err when loading jpeg. The 2nd build was okay.

```
icetime -d hx8k -c 48 catzip.asc
// Reading input .asc file..
// Reading 8k chipdb file..
// Creating timing netlist..
// Timing estimate: 15.91 ns (62.86 MHz)
// Checking 20.83 ns (48.00 MHz) clock constraint: PASSED.
```

This is what gets inserted into toplevel.v

```
`ifdef        VERILATOR
        assign       s_clk = i_clk;
`else
        wire clk_48mhz, pll_locked;
        SB_PLL40_CORE #(
                .FEEDBACK_PATH("SIMPLE"),
                .DELAY_ADJUSTMENT_MODE_FEEDBACK("FIXED")
,
                .DELAY_ADJUSTMENT_MODE_RELATIVE("FIXED"),
                .PLLOUT_SELECT("GENCLK"),
                .FDA_FEEDBACK(4'b1111),
                .FDA_RELATIVE(4'b1111),
                .DIVR(4'b0010),              // DIVR =  2
                .DIVF(4'b0010110),           // DIVQ =  22
                .DIVQ(3'b100),         // DIVF =  4
                .FILTER_RANGE(3'b011)       // FILTER_RANGE = 3
        ) plli (
                .REFERENCECLK     (i_clk        ),
                .PLLOUTCORE     (clk_48mhz     ),
```

```
            .LOCK           (pll_locked  ),
            .BYPASS         (1'b0        ),
            .RESETB         (1'b1        )
        );
        //SB_GB global_buffer(clk_48mhz, s_clk);
        assign       s_clk = clk_48mhz;
`endif

icepll -i 100 -o 48

F_PLLIN:   100.000 MHz (given)
F_PLLOUT:   48.000 MHz (requested)
F_PLLOUT:   47.917 MHz (achieved)

FEEDBACK: SIMPLE
F_PFD:   33.333 MHz
F_VCO:  766.667 MHz

DIVR:  2 (4'b0010)
DIVF: 22 (7'b0010110)
DIVQ:  4 (3'b100)

FILTER_RANGE: 3 (3'b011)

endmodule // end of toplevel.v module definition
```

Build steps:

git clone git@github.com:develone/icozip.git -b catzip-dev catzip-dev

cd catzip-dev/

devel@mypi3-20:~/testbuilds/catzip-dev $ mousepad rtl/catzip/cpu/zipcpu.v
line 737

Note: These next 4 steps might be needed to be repeated one or more times.
This creates a catzip.bin. If the runjpeg does not end
in ./arm-netpport window with the following.
. w=256 h=256
. 1401000 2
. flag 2 0x204fe08
. 1401004 1
. 140100c 200fe08

cd ~/testbuilds/catzip/rtl/catzip/

rm -f catzip.asc catzip.json catzip.bin

make bin

. ./myenv-a.sh

Testing upgrade to zipcpu that fixes LOCK instruction generation in GCC
rtl/catzip/cpu/zipcpu.v line 737

.OPT_LOCK(OPT_LOCK),

FPGA Tools:

autofpga
commit 5b57538081d60a04cd32aeba2436816062cb0141 (HEAD -> master, upstream/master, upstream/dev)
Author: ZipCPU <dgisselq@ieee.org>
Date: Fri Aug 14 07:55:55 2020 -0400

    Fixes MacOS build err--std::string vs char * in gbl_msg.err()

zipcpu

devel@mypi3-20:~/zipcpu $ git log
commit 548a0efbe5729279f2ae9779a2af4229e8241e52 (HEAD -> master, upstream/master, upstream/dev)
Author: ZipCPU <dgisselq@ieee.org>
Date: Wed Oct 6 11:54:56 2021 -0400

    Fixed LOCK instruction generation in GCC

devel@mypi3-20:~/autofpga $ yosys -V
Yosys 0.9+4081 (git sha1 83a21814, gcc 8.3.0-6+rpi1 -fPIC -Os)

devel@mypi3-20:~/autofpga $ nextpnr-ice40 -V
nextpnr-ice40 -- Next Generation Place and Route (Version 67bd349e)

Verilator 4.100 2020-09-07 rev v4.100-10-g39eea781

Simulation:

devel@mypi3-20:~/testbuilds/catzip-dev/sim/verilated $ clear; ./arm-main_tb

Listening on port 8363

Listening on port 8364
> T

Running CPU self-test

SIM Instructions Is this a simulator?
CIS Instructions Supported
Break test #1 Pass
Break test #2 Pass
Break test #3 Pass
Early Branch test Pass
Trap test/AND Pass
Trap test/CLR Pass
Overflow test Pass
Carry test Pass
Loop test Pass
Shift test Pass
Pipeline test Pass
Mem-Pipeline test Pass
Conditional Execution test Pass
No-waiting pipeline test Pass
Conditional Branching test Pass
Ill Instruction test, NULL PC Pass
Ill Instruction test, two Pass
Comparison test, == Pass
Comparison test, != Pass
CC Register test Pass
Multi-Arg test Pass
Multiply test Pass

devel@mypi3-20:~/testbuilds/catzip-dev/sim/verilated $ clear; ./arm-main_tb

Listening on port 8363
Listening on port 8364
> T
Accepted CMD connection
< A04000001Wf
> A04000001K00000000
< [CLOSED]
Hello, World

devel@mypi3-20:~/testbuilds/catzip-dev/sw/host $ ./arm-zipload -v ../board/cputest

Halting the CPU
Memory regions:
Block RAM: 01400000 - 01402000
SDRAM : 02000000 - 03000000
Loading: ../board/cputest
Section 0: 02000000 - 020039f0
Writing to MEM: 02000000-020039f0

Clearing the CPUs registers
Setting PC to 02000000
The CPU should be fully loaded, you may now
start it (from reset/reboot) with:
> wbregs cpu 0x0f

CPU Status is: 0000060f

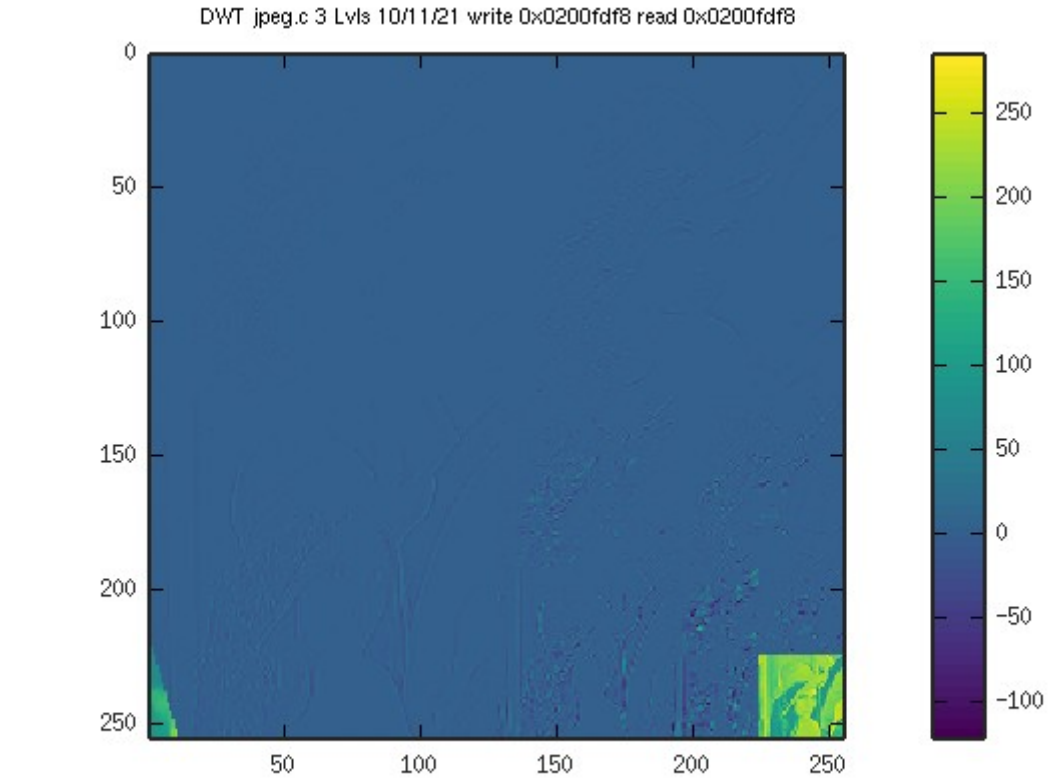
devel@mypi3-20:~/testbuilds/catzip-dev/sw/host $ ./arm-wbregs cpu 0x0f
04000000 (CPU)-> 0000000f

devel@mypi3-20:~/testbuilds/catzip-dev/sw/host $ ./arm-zipload -v ../board/hello
Halting the CPU
Memory regions:
        Block RAM: 01400000 - 01402000
        SDRAM   : 02000000 - 03000000
Loading: ../board/hello
Section 0: 02000000 - 020047c8
Writing to MEM: 02000000-020047c8
Clearing the CPUs registers
Setting PC to 02000000
The CPU should be fully loaded, you may now
start it (from reset/reboot) with:
> wbregs cpu 0x0f

CPU Status is: 0000060f

devel@mypi3-20:~/testbuilds/catzip-dev/sw/host $ ./arm-wbregs cpu 0x0f

jpeg simulation



Testing Verilog bins:

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/rtl/uart $ sudo config_cat helloworld.bin

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/rtl/uart $ sudo config_cat rtl/basic/dimmer.bin

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/rtl/uart $ sudo config_cat rtl/basic/blinky.bin

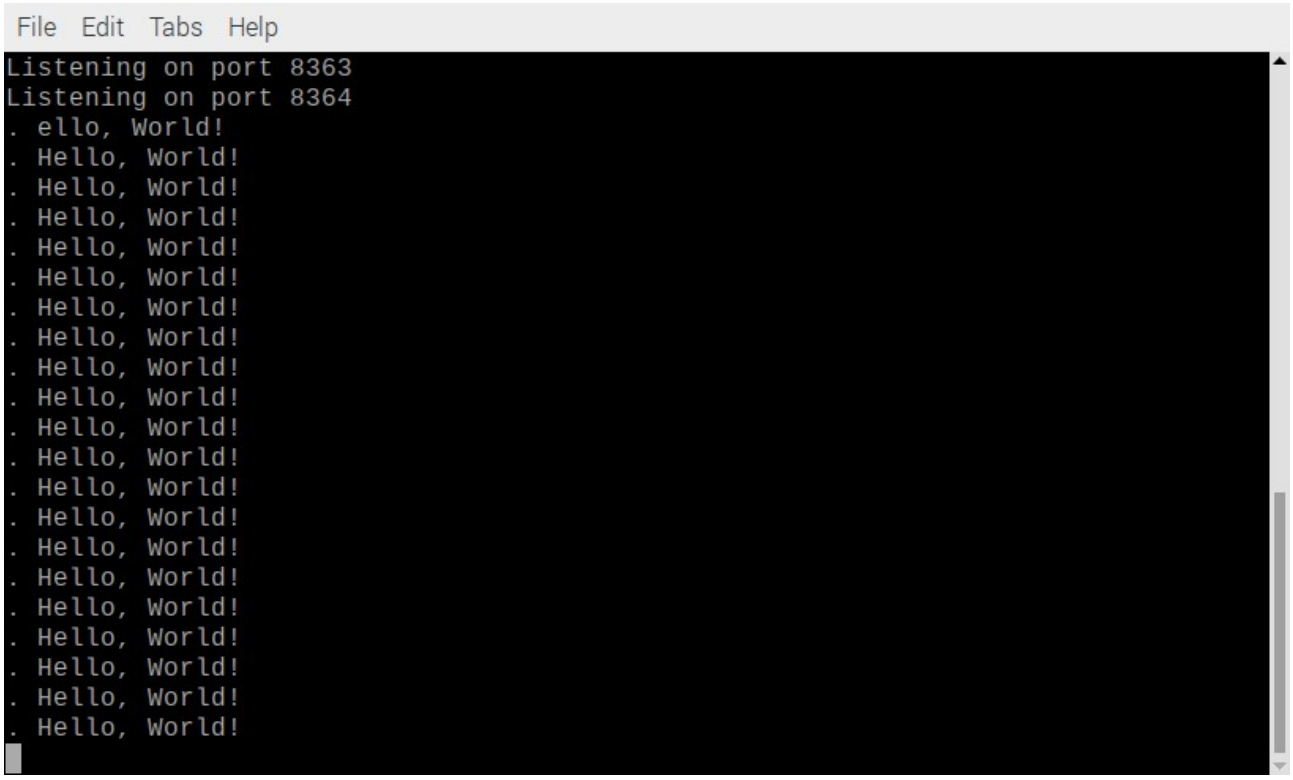
devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/rtl/uart $ sudo config_cat rtl/switch_leds/switch_leds.bin

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/rtl/uart $ sudo config_cat rtl/leddigits/leddigits.bin

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ sudo ./arm-netpport

Listening on port 8363
Listening on port 8364

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/rtl/pptest $ sudo config_cat hellopp.bin



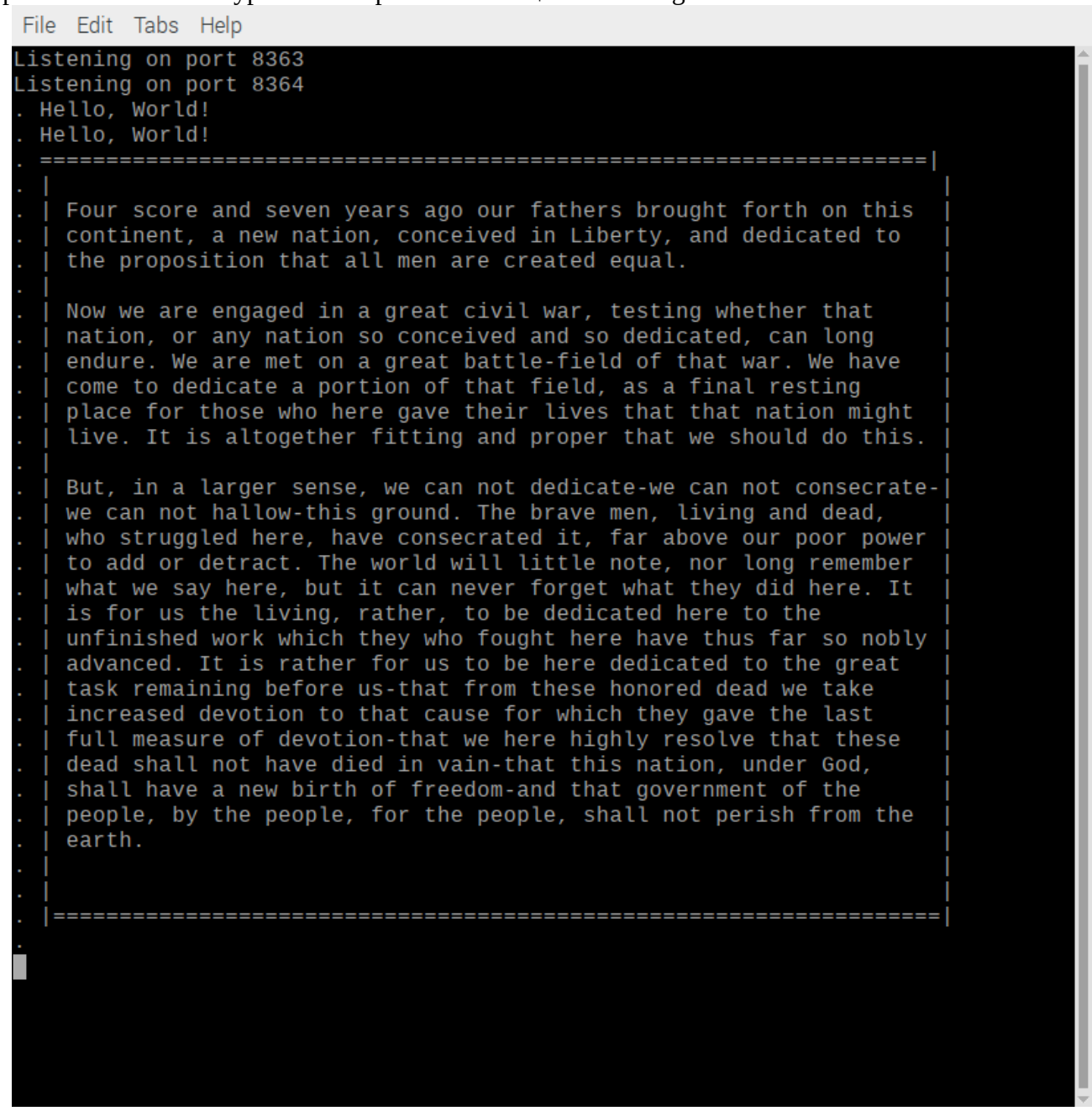
devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/rtl/pptest $ sudo config_cat speechpp.bin

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ sudo config_cat ../../rtl/catzip/catzip.bin

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ clear; sudo ./arm-netpport

Listening on port 8363
Listening on port 8364

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ ./arm-wbregs version



01000014 (VERSION) : [.!..] 20211012

Testing sdram Write & Read:

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ ./arm-wrsdram r.bin
The size of the buffer is 0x010000 or 65536 words

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ ./arm-rdsdram sdram.bin

Write-COMPLETE
The size of the buffer is 0x010000 or 65536 words
devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ diff r.bin sdram.bin

Command port is now connected
< A04000001Wf
> A04000001K00000000
Command port disconnect
. w=2-5 h=2-5
. 1401000 2
. flag 2 0x204fdf8
. 1401004 1
. 140100c 200fdf8

. ptrs.alt 0x2054498 0x200fdf8
. 1:p/?1

. 0x200fdf8 0xe2
. 0x200fdfc 0xe2
. 0x200fe00 0xe1
. 0x200fe04 0xe1
.
. 0x202fdf8 0xcf
. 0x202fdfc 0xcf
. 0x202fe00 0xcf
. 0x202fe04 0xcf
.
. 0x204fde8 0x96
. 0x204fdec 0xa1
. 0x204fdf0 0xab
. 0x204fdf4 0xb4
. 2-5 0x200fdf8 0x2054498 0x 1401004
. 2-5

The runjpeg,sh sets several values in blkram used to wait on user.

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ ./runjpeg.sh
01401000 (   )-> 00000002
01401004 (   )-> 00000001
01401008 (   )-> 00000001
0140100c (   )-> 00000000
01401010 (   )-> 00000001
01401018 (   )-> 00000001
Halting the CPU
Memory regions:

```
        Block RAM: 01400000 - 01402000
        SDRAM      : 02000000 - 03000000
Loading: ../board/jpeg
Section 0: 02000000 - 0204fe7c
Writing to MEM: 02000000-0204fe7c

Clearing the CPUs registers
Setting PC to 02000000
The CPU should be fully loaded, you may now
start it (from reset/reboot) with:
> wbregs cpu 0x0f

CPU Status is: 0000060f
04000000 (     CPU)-> 0000000f
```

This step writes the file r.bin to the sdram. The values can be g.bin or b.bin.

```
devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ ./arm-wrsdram r.bin
The size of the buffer is 0x010000 or 65536 words

READ-COMPLETE
```

This steps lets the user where the input and output will be written and read.

```
devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ ./arm-wbregs 0x0140100c
0140100c (        ) : [....] 0200fdf8
```

This step clears the wait loop.

```
devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ ./arm-wbregs 0x01401008 0x0
01401008 (        )-> 00000000
```

This step clears the wait1 loop.

```
devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ ./arm-wbregs 0x01401010 0x0
01401010 (        )-> 00000000
```

This step reads the sdram and stores the data in dwt.bin used by octave to show the lifting step results.

```
devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ rm -f dwt.bin; ./arm-rdsdram dwt.bin

Write-COMPLETE
The size of the buffer is 0x010000 or 65536 words

devel@mypi3-19:~/testbuilds/mypi3-20/catzip-dev/sw/host $ octave
libEGL warning: DRI3: failed to query the version
libEGL warning: DRI2: failed to authenticate
GNU Octave, version 4.4.1
Copyright (C) 2018 John W. Eaton and others.
This is free software; see the source code for copying conditions.
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE.  For details, type 'warranty'.
```

Octave was configured for "arm-unknown-linux-gnueabihf".

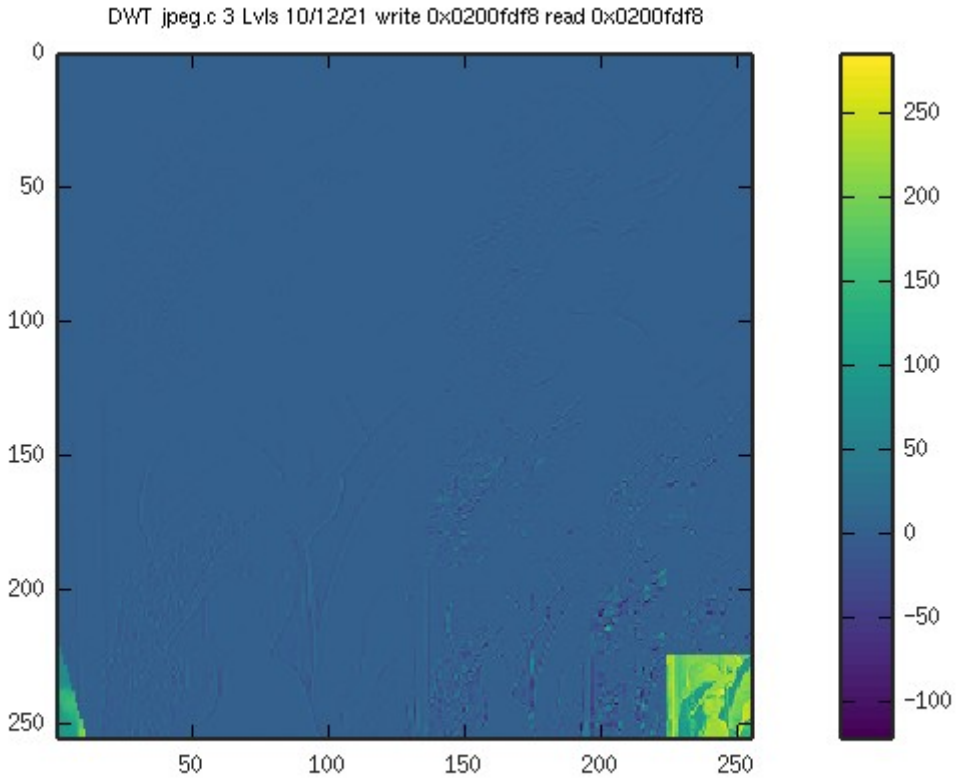
Additional information about Octave is available at https://www.octave.org.

Please contribute if you find this software useful.
For more information, visit https://www.octave.org/get-involved.html

Read https://www.octave.org/bugs.html to learn how to submit bug reports.
For information about changes from previous versions, type 'news'.

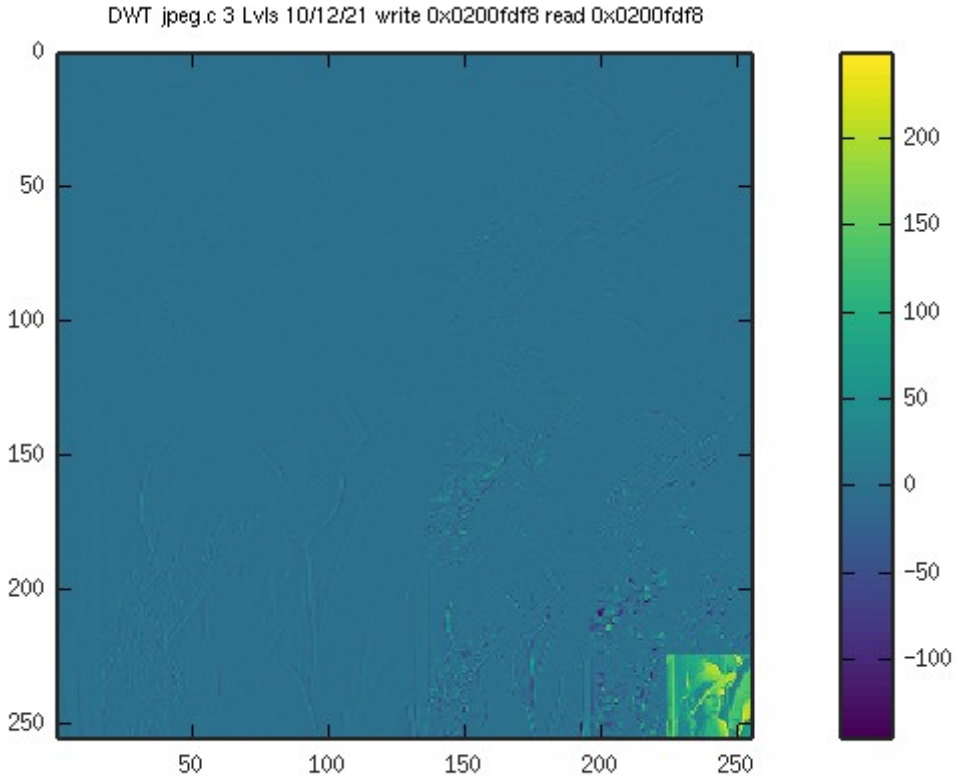
octave:1>rgb

dwt-r



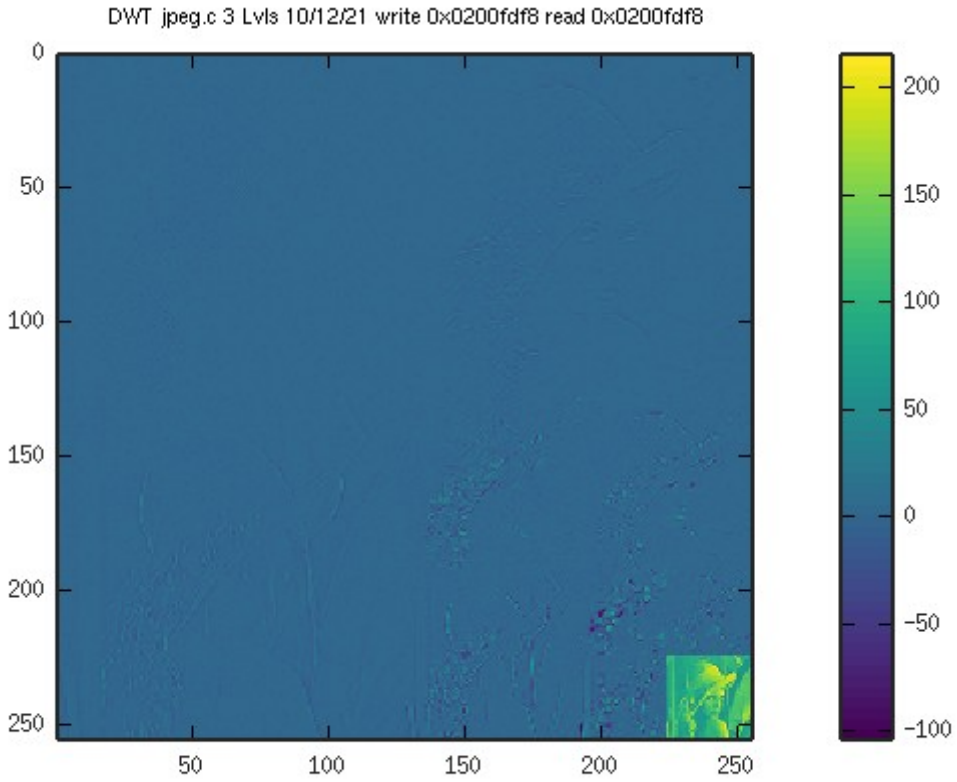
y2= 13.1087

dwt-g



y2=-199.205

dwt-b


DWT jpeg.c 3 Lvls 10/12/21 write 0x0200fdf8 read 0x0200fdf8
0
50
100
150
200
250
50
100
150
200
250
200
150
100
50
0
-50
-100

y2=-146.934