Testing Upgrades to icestorm, yosys, zipcpu and nextpnr
autofpga & verilator
mypi3-20 pi400-2 pi400-1
mypi3-16 mypi3-18 ws011
10/11/21

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

```
-           .OPT_LOCK(OPT_LOCK),
+           .OPT_LOCK(1'b1),
```

icepll -i 100 -o 40 -f 40pll.txt -m

cat 40pll.txt
```
/**
 * PLL configuration
 *
 * This Verilog module was generated automatically
 * using the icepll tool from the IceStorm project.
 * Use at your own risk.
 *
 * Given input frequency:        100.000 MHz
 * Requested output frequency:   40.000 MHz
 * Achieved output frequency:    40.000 MHz
 */

module pll(
        input  clock_in,
        output clock_out,
        output locked
        );

SB_PLL40_CORE #(
            .FEEDBACK_PATH("SIMPLE"),
            .DIVR(4'b0100),         // DIVR =  4
            .DIVF(7'b0011111),   // DIVF = 31
            .DIVQ(3'b100),          // DIVQ =  4
            .FILTER_RANGE(3'b010)   // FILTER_RANGE = 2
        ) uut (
            .LOCK(locked),
            .RESETB(1'b1),
            .BYPASS(1'b0),
            .REFERENCECLK(clock_in),
            .PLLOUTCORE(clock_out)
            );

endmodule
```

4th build on pi400-1 finally ran correctly
2nd build on ws011 finally ran correctly
3rd build on mypi3-20 finally ran correctly
2nd build on pi400-2 finally ran correctly
1st build on mypi3-16 ran correctly
6th build on mypi3-18 finally ran correctly

sw/host/rgb.m

```
-title "DWT  jpeg.c 3 Lvls 10/06/21 write 0x0200fe08 read 0x0200fe08"
+title "DWT  jpeg.c 3 Lvls 10/11/21 write 0x0200fdf8 read 0x0200fdf8"
```

sw/host/wrsdram.cpp

```
-#define        DUMPMEM        0x0200fe08
+#define        DUMPMEM        0x0200fdf8
```

sw/host/rdsdram.cpp

```
-#define        DUMPMEM        0x0200fe08
+#define        DUMPMEM        0x0200fdf8
```

auto-data/clockpll40.txt

```
-        .DIVQ(3'b100),        // DIVF =  4
+        .DIVQ(7'b0011111),        // DIVQ =  31
+        .DIVF(3'b100),        // DIVF =  4
```

rtl/catzip/cpu/zipcpu.v

```
-        .OPT_LOCK(OPT_LOCK),
+        .OPT_LOCK(1'b1),
```

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

```
-#define        DUMPMEM        0x0200fe08
+#define        DUMPMEM        0x0200fdf8
```

ws011

./pc-wbregs -n mypi3-19 version
01000014 (VERSION) : [.!..] 20211011

mypi3-16

./arm-wbregs version
01000014 (VERSION) : [.!..] 20211011

mypi3-20

./arm-wbregs version
01000014 (VERSION) : [.!..] 20211011

pi400-1

./arm-wbregs version
01000014 (VERSION) : [.!..] 20211011

pi400-2

/arm-wbregs version
01000014 (VERSION) : [.!..] 20211011

mypi3-18

./arm-wbregs version
01000014 (VERSION) : [.!..] 20211011

devel@pi400-1:~/testbuilds $ git clone git@github.com:develone/icozip.git -b catzip catzip

devel@pi400-1:~/testbuilds $ cd catzip/

evel@pi400-1:cp auto-data/clockpll40.txt.wkg auto-data/clockpll40.txt

/home/devel/zipcpu/sw/install/cross-tools/bin:/home/devel/.local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/local/games:/usr/games

devel@pi400-1:~/testbuilds/catzip $ make

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

Tested verilog & C++
commit 5df54b35ca8724fe0cf3149f1543cfa2dd66acd7 (HEAD -> catzip)
Author: Edward Vidal Jr <develone@sbcglobal.net>
Date: Wed Sep 29 06:23:55 2021 -0600

    missing from last commit

yosys -V
Yosys 0.9+4081 (git sha1 83a21814, gcc 8.3.0-6+rpi1 -fPIC -Os)

nextpnr-ice40 -V
nextpnr-ice40 -- Next Generation Place and Route (Version 67bd349e)

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

zipcpu

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

autofpga
devel@mypi3-20:~/autofpga $ git log
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()

icestorm pi400-1

commit 83b8ef947f77723f602b706eac16281e37de278c (HEAD -> master, upstream/master)
Merge: 9880f6e 97b6e27
Author: Miodrag Milanović <mmicko@gmail.com>
Date: Mon Sep 6 11:14:46 2021 +0200

    Merge pull request #282 from jkiv/iceprog-ignorig-fix

    [iceprog] Fixed typo in error message: "Ignorig"

```
devel@mypi3-19:~/testbuilds/mypi3-20/catzip/sw/host $ sudo ./arm-netpport
Listening on port 8363
Listening on port 8364
.  |===================================================================|
.  |                                                                   |
.  | Four score and seven years ago our fathers brought forth on this  |
.  | continent, a new nation, conceived in Liberty, and dedicated to   |
.  | the proposition that all men are created equal.                   |
.  |                                                                   |
.  | Now we are engaged in a great civil war, testing whether that     |
.  | nation, or any nation so conceived and so dedicated, can long     |
.  | endure. We are met on a great battle-field of that war. We have   |
.  | come to dedicate a portion of that field, as a final resting      |
.  | place for those who here gave their lives that that nation might  |
.  | live. It is altogether fitting and proper that we should do this. |
.  |                                                                   |
.  | But, in a larger sense, we can not dedicate-we can not consecrate-|
.  | we can not hallow-this ground. The brave men, living and dead,    |
.  | who struggled here, have consecrated it, far above our poor power |
.  | to add or detract. The world will little note, nor long remember  |
.  | what we say here, but it can never forget what they did here. It  |
.  | is for us the living, rather, to be dedicated here to the         |
.  | unfinished work which they who fought here have thus far so nobly  |
.  | advanced. It is rather for us to be here dedicated to the great    |
.  | task remaining before us-that from these honored dead we take      |
.  | increased devotion to that cause for which they gave the last      |
.  | full measure of devotion-that we here highly resolve that these    |
.  | dead shall not have died in vain-that this nation, under God,      |
.  | shall have a new birth of freedom-and that government of the       |
.  | people, by the people, for the people, shall not perish from the  |
.  | earth.                                                            |
.  |                                                                   |
.  |                                                                   |
.  |===================================================================|
.
.   ello, World!
. Hello, World!
. Hello, World!
^C
devel@mypi3-19:~/testbuilds/mypi3-20/catzip/sw/host $ 
```

cputest

```
> A04000005K00000000
< A04000001W41f
> A04000001K00000000
< A04000005W
> A04000005K00000000
< A04000001Wc00
> A04000001K00000000
< A04000001W40f
> A04000001K00000000
< A04000005W2000000
> A04000005
> K00000000
< A04000001R
> A04000001R0000060f
Command port disconnect
Command port is now connected
< A04000001Wf
> A04000001K00000000
Command port disconnect
#
# Running CPU self-test
# --------------------------------
# SIM Instructions              Pass
# 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
# Multiply HI-word test         Pass
# Divide test                   Pass
#
# --------------------------------
# All tests passed.  Halting CPU.
#
```

hello

```
File   Edit   Tabs   Help
> A04000005K00000000
< A04000001W417
> A04000001K00000000
< A04000005W
> A04000005K00000000
< A04000001W418
> A04000001K00000000
< A04000005W
> A04000005K00000000
< A04000001W419
> A04000001K00000000
< A04000005W
> A04000005K00000000
< A04000001W41a
> A04000001K00000000
< A04000005W
> A04000005K00000000
< A04000001W41b
> A04000001K00000000
< A04000005W
> A04000005K00000000
< A04000001W41c
> A04000001K00000000
< A04000005W
> A04000005K00000000
< A04000001W41d
> A04000001K00000000
< A04000005W
> A04000005K00000000
< A04000001W41e
> A04000001K00000000
< A04000005W
> A04000005K00000000
< A04000001W41f
> A04000001K00000000
< A04000005W
> A04000005K00000000
< A04000001Wc00
> A04000001K00000000
< A04000001W40f
> A04000001K00000000
< A04000005W2000000
> A04000005
> K00000000
< A04000001R
> A04000001R0000060f
Command port disconnect
Command port is now connected
< A04000001Wf
> A04000001K00000000
Command port disconnect
# Hello, World!
```

now using b.bin which was extracted from blu.pgm with wrpgm.m
This does not need to use split.
./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 - 0204fe6c
Writing to MEM: 02000000-0204fe6c
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

# w=256  h=256
# 1401000 2
# flag 2 0x204fde8
# 1401004 1
# 140100c 200fde8

./arm-wrsdram b.bin  or ./arm-wrsdram g.bin or ./arm-wrsdram r.bin
The size of the buffer is 0x010000 or 65536 words

READ-COMPLETE

./arm-wbregs 0x01401008 0
01401008 (        )-> 00000000

#  ptrs.alt 0x20544c8 0x200fde8
# 131072

./arm-wbregs 0x01401010 0
01401010 (        )-> 00000000

./arm-wbregs 0x01401008 0
01401008 (        )-> 00000000

./arm-wbregs 0x01401010 0
01401010 (        )-> 00000000

# 0x200fde8 0x7e
# 0x200fdec 0x7d
# 0x200fdf0 0x7b
# 0x200fdf4 0x76
#
# 0x202fde8 0x6a
```
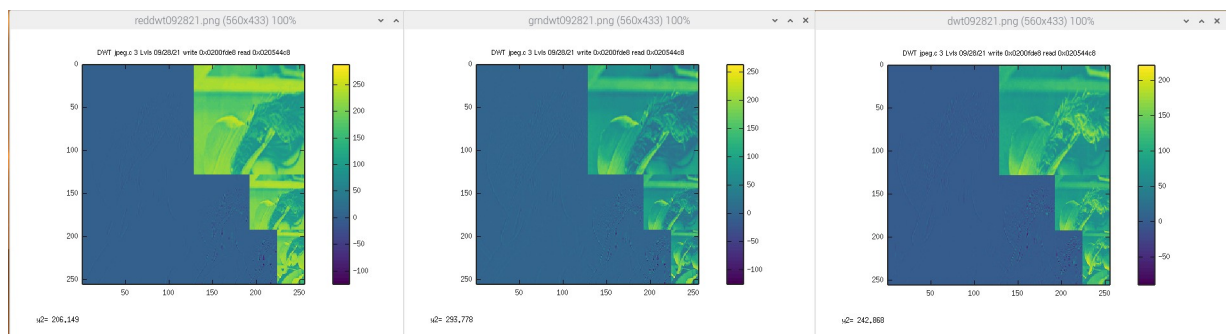
0x202fdec 0x6a
0x202fdf0 0x68
0x202fdf4 0x64
#
0x204fdd8 0x48
0x204fddc 0x4a
0x204fde0 0x4e
0x204fde4 0x50
256 0x200fde8 0x20544c8 0x 1401004
256

rm -f dwt.bin; ./arm-rdsdram dwt.bin

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

red grn & blu ptrs.alt

red grn & blu ptrs.inpbuf