**********DRAFT**********

SDRAM

09/28/18

**********DRAFT**********

Trying to add sdram support to the catzip. Steps

Create sdramdev.v Verilog file and the testbench main_sdramdev_tb.vcd.

Steps to obtain the Values Change Dump VCD. This creates the file sdramdev.v.

In the directory /home/pi/testbuilds/learning_hdl/MyHDL/sdram

"rm -f *.vcd"

"python main_sdramdev.py"

"gtkwave main_sdramdev_tb.vcd"

Create sdramdev.txt this is currently a draft version for use with /home/pi/testbuilds/catzip_simulated/catzip/auto-data for the catzip.

/home/pi/testbuilds/catzip_simulated/catzip/make autodata

Copying auto-data/toplevel.v to rtl/catzip/toplevel.v

Copying auto-data/main.v to rtl/catzip/main.v

Copying auto-data/regdefs.h to sw/host/regdefs.h

Copying auto-data/regdefs.cpp to sw/host/regdefs.cpp

Copying auto-data/board.h to sw/board/board.h

Copying auto-data/board.ld to sw/board/board.ld

Copying auto-data/rtl.make.inc to rtl/catzip/auto.mk

Copying auto-data/main_tb.cpp to sim/verilated/main_tb.cpp

Copying auto-data/testb.h to sim/verilated/testb.h

Steps to obtain the Values Change Dump VCD. This creates the file sdramdev.v.

Add the sdramsim.cpp & sdramsim.h to /testbuilds/catzip_simulated/catzip/sim/verilated.

The script restores "config_catzip_simulation.sh"

Starting with a working catzip

/home/pi/testbuilds/catzip_simulation/catzip/sim/verilated/

cd testbuilds/catzip_simulation/catzip/sim/verilated/

In 1 shell

./arm-main_tb

The script used to execute the arm-wbregs command is found at Appendix A: Script that executes the arm-wbregs commands sim_hw_test.sh

The output of the arm-main_tb is found at Appendix B: CATZIP testing the verilator simulation.

Listening on port 8363

Listening on port 8364

> T

In 2nd shell

cd testbuilds/catzip_simulation/catzip/sw/host/

./ico_sim_hw_test.sh

The output of the ico_sim_hw_test.sh is found at Appendix C: CATZIP testing the verilator simulation running wbregs command from the script sim_hw_test.sh .

cp testbuilds/learning_hdl/MyHDL/sdram/sdramdev.v ~/testbuilds/icozip/rtl/icozip/

cp ~/testbuilds/learning_hdl/MyHDL/sdram/sdramdev.txt ~/testbuilds/icozip/auto-data/

cp ~/testbuilds/learning_hdl/MyHDL/sdram/Makefile_autodata ~/testbuilds/icozip/auto-data/Makefile

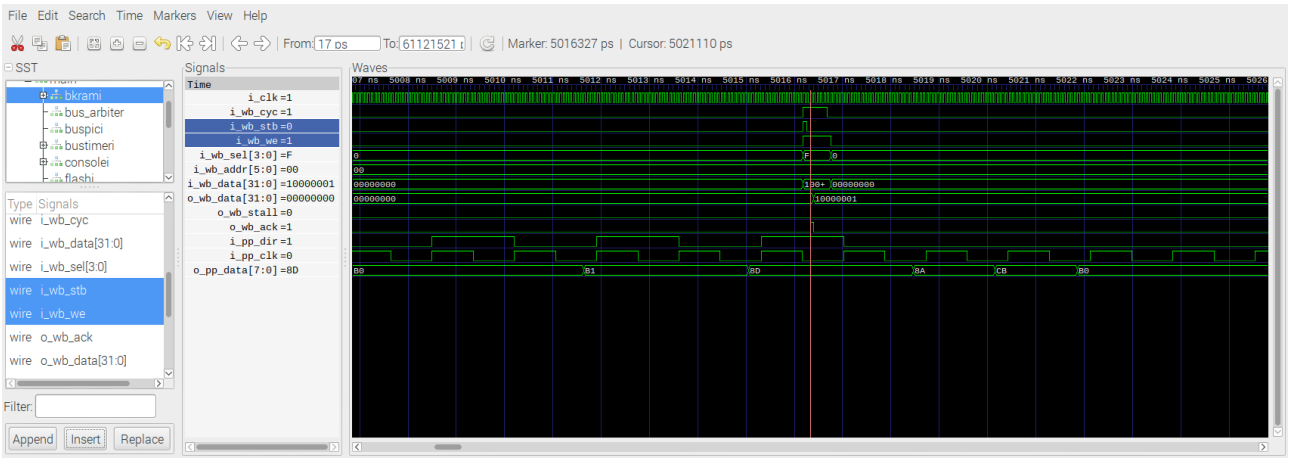
Note This provides a verilator sim of both sram & blkram

. ~/testbuilds/catzip/myenv.sh

cd ~/testbuilds/icozip/autodata

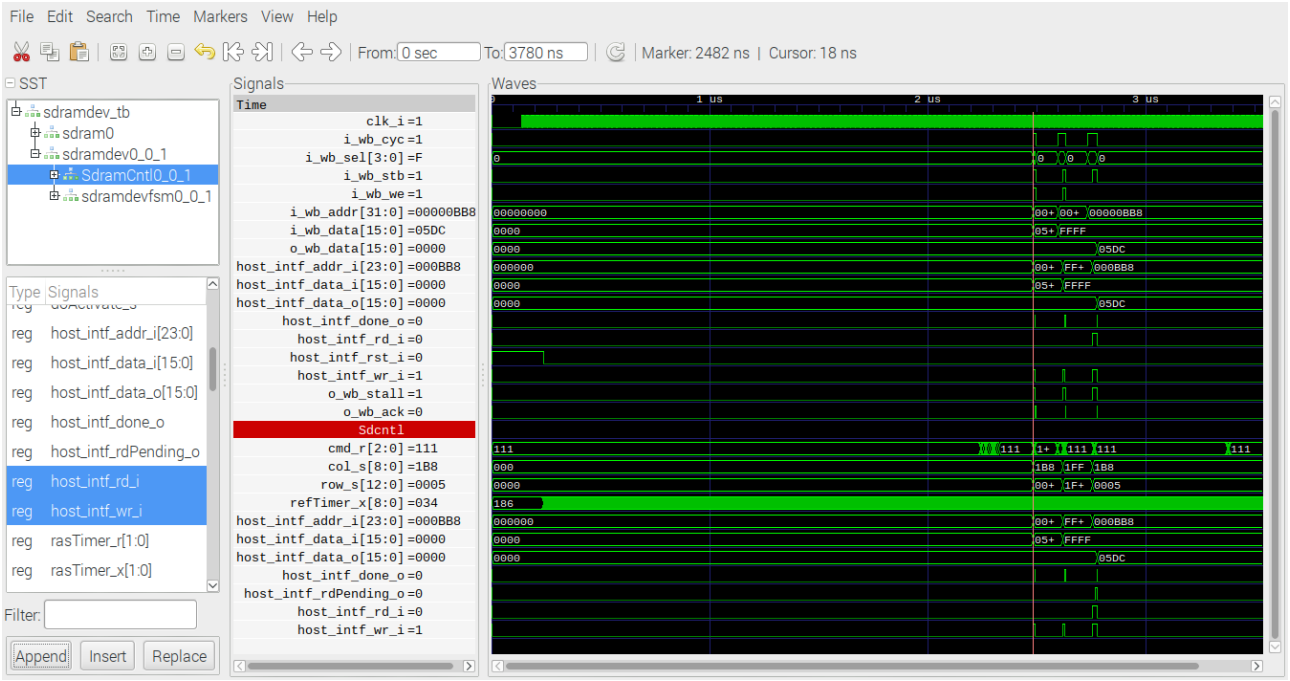
Modified sramdev.txt → sdramdev.txt

Modified Makefile in icozip/autodata → Makefile_autodata



The Catboard has HX8K FPGA with a SDRAM that provides 0x000000 to 0xFFFFFF 16 bit locations.

The SDCONTROLLER takes care of refreshing the SDRAM and interface between the FPGA and the SDRAM. The data 0x05DC 1500 is being wrote to address 0x000BB8 3000, the data 0xFFFF 65535 is being wrote to address 0xFFFFFF 16777215. Then the data is read from address 0x000BB8 3000.



This test writes two SDRAM address and reads the value wrote to first location.
The data 0x05DC 1500 is being wrote to address 0x00000BB8 3000, 0xFFFF 65535 is being wrote to address 0x00FFFFFF 16777215. This is followed by reading the value at address 0x00000BB8 3000 which was 0x05DC 1500.

```
BANK 0 STATE : [CHANGE] Uninitialized -> Initialized @  2138
 BANK 1 STATE : [CHANGE] Uninitialized -> Initialized @  2138
 BANK 2 STATE : [CHANGE] Uninitialized -> Initialized @  2138
 BANK 3 STATE : [CHANGE] Uninitialized -> Initialized @  2138
-------------------------
 Mode   | CAS   |   Burst
--------|-------|---------
 Burst  | 3     |   1
-------------------------
 SDRAM : Bank 0 has active row 0005
 BANK 0 STATE : WRITING @  2514
 DATA : [WRITE] Addr: 440  Data: 1500
```

The data 0x05DC 1500 is being wrote to address 0x000BB8 3000 Row 0005 col Addr: 440 0x1B8 is where the 1500 0x05DC is written.

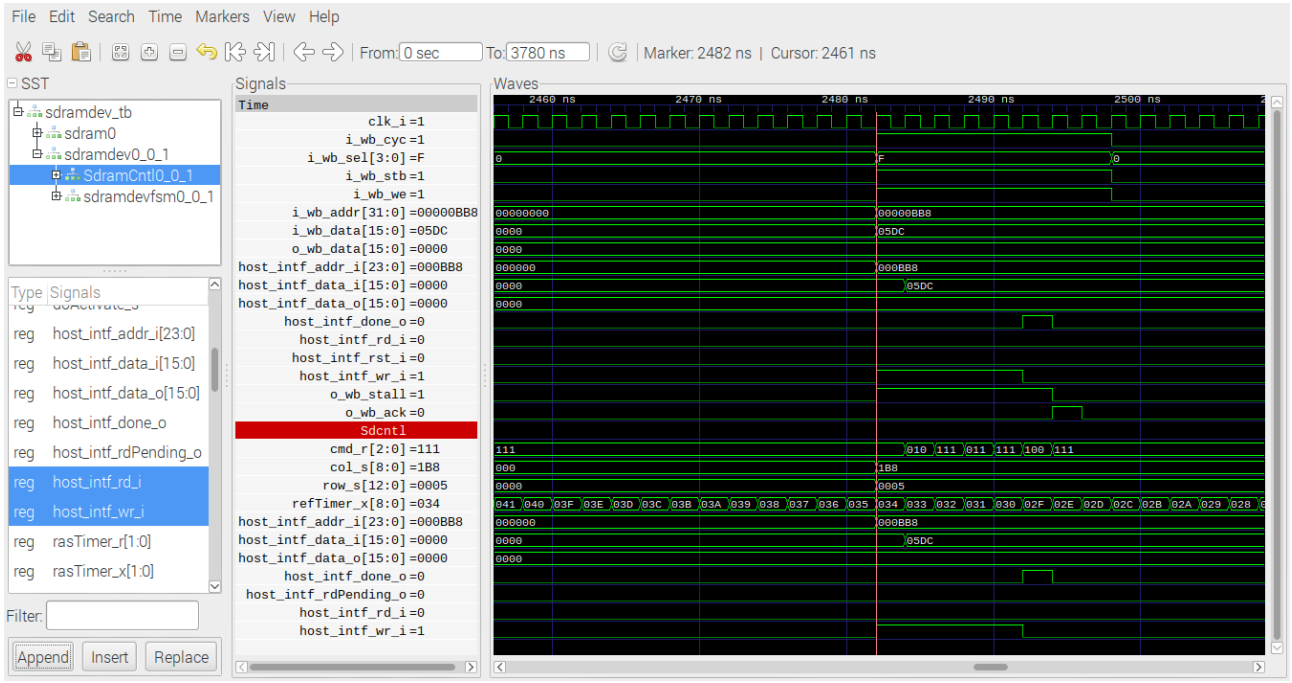
```
SDRAM : Bank 3 has active row 1fff
 BANK 3 STATE : WRITING @  2652
 DATA : [WRITE] Addr: 511  Data: 65535
 BANK 3 STATE : WRITING TO 01ff:511
```

The data 0xFFFF 65535 is being wrote to address 0xFFFFFF 16777215 (this is the last location in the sdram) Row 1FFF col Addr: 511 0x1FF is where the 65535 0xFFFF is written.

```
SDRAM : Bank 0 has active row 0005
 BANK 0 STATE : READING @  2790
 SDRAM : [READ] Commnad registered
 BANK 0 STATE : READING FROM 01b8:440
 STATE : [READ] Data Ready @  2794  value :  1500
```

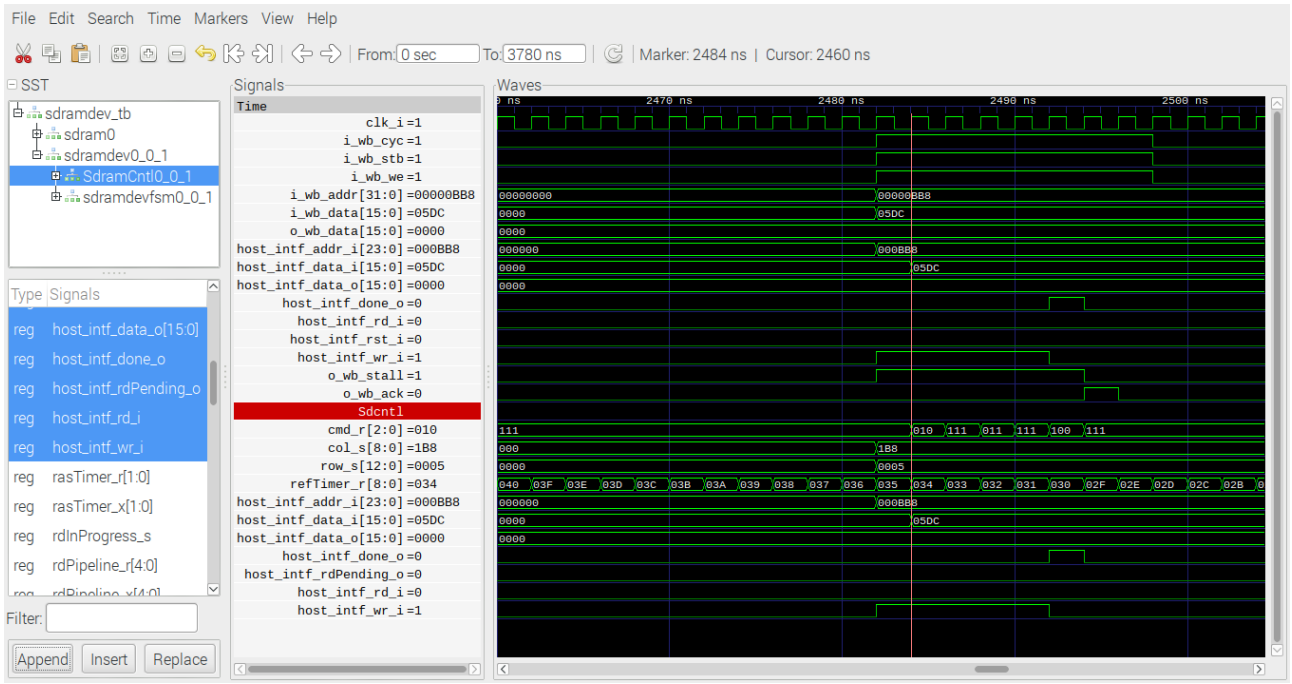
BANK 0 STATE : WRITING TO 01b8:440
4400440: 1500
SDRAM : Bank 3 has active row 1fff
BANK 3 STATE : WRITING @ 2652
DATA : [WRITE] Addr: 511 Data: 65535
BANK 3 STATE : WRITING TO 01ff:511
4400440: 1500
5110511: 65535
SDRAM : Bank 0 has active row 0005
BANK 0 STATE : READING @ 2790
SDRAM : [READ] Commnad registered
BANK 0 STATE : READING FROM 01b8:440
STATE : [READ] Data Ready @ 2794 value : 1500

In the image below the data 0x05DC 1500 is being wrote to address 0x00000BB8 3000, 0xFFFF 65535 is being wrote to address 0x00FFFFFF 16777215. This is followed by reading the value at address 0x00000BB8 3000 which was 0x05DC 1500.
BANK 3 STATE : WRITING @ 2652

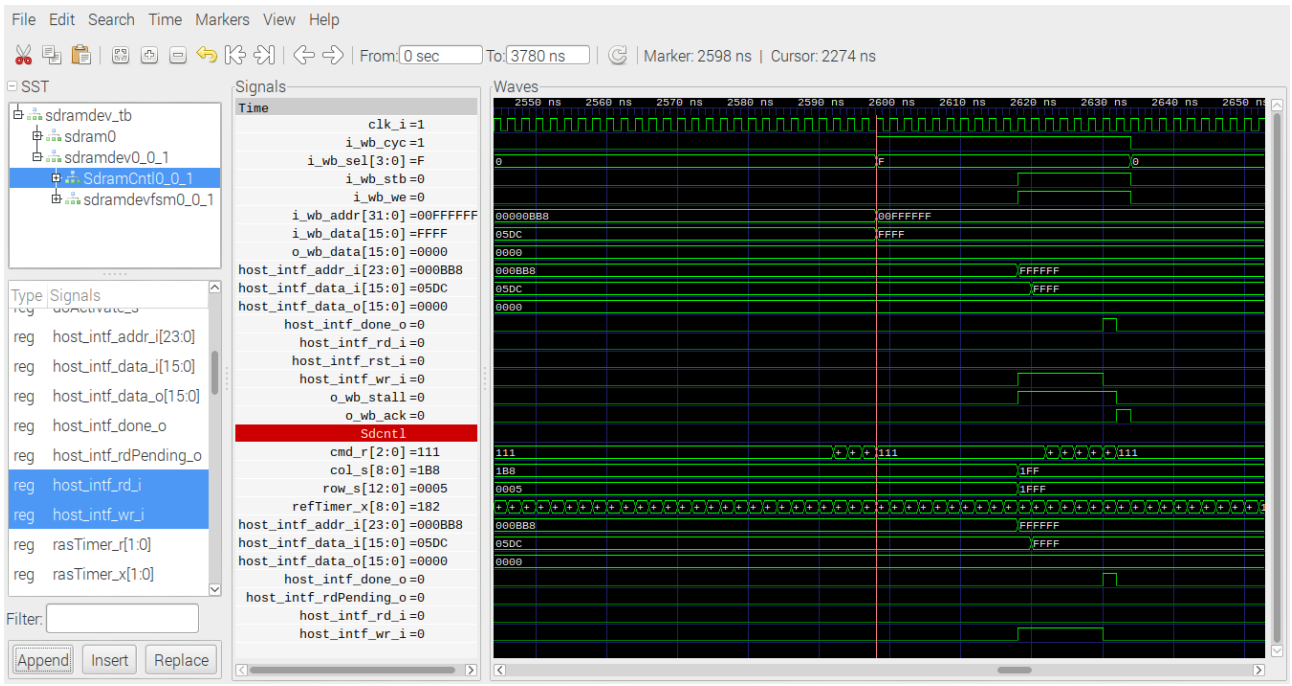


DATA : [WRITE] Addr: 511 Data: 65535
BANK 3 STATE : WRITING TO 01ff:511

SDRAM : Bank 0 has active row 0005
BANK 0 STATE : WRITING @ 2514
DATA : [WRITE] Addr: 440 Data: 1500
BANK 0 STATE : WRITING TO 01b8:440
In the image below the data 0x05DC 1500 is being wrote to address 0x000BB8 3000

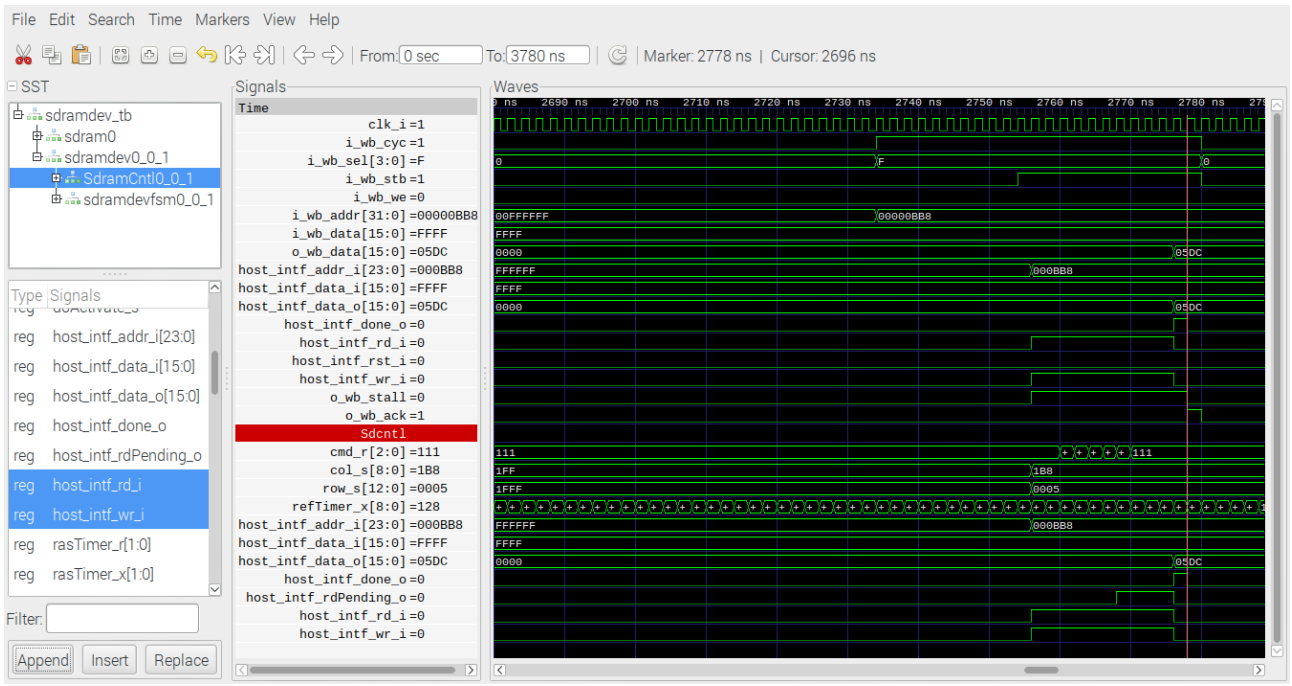


SDRAM : Bank 3 has active row 1fff
 BANK 3 STATE : WRITING @ 2652
 DATA : [WRITE] Addr: 511 Data: 65535
 BANK 3 STATE : WRITING TO 01ff:511
In the image below the data 0xFFFF 65535 is being wrote to address 0xFFFFFF 16777215.



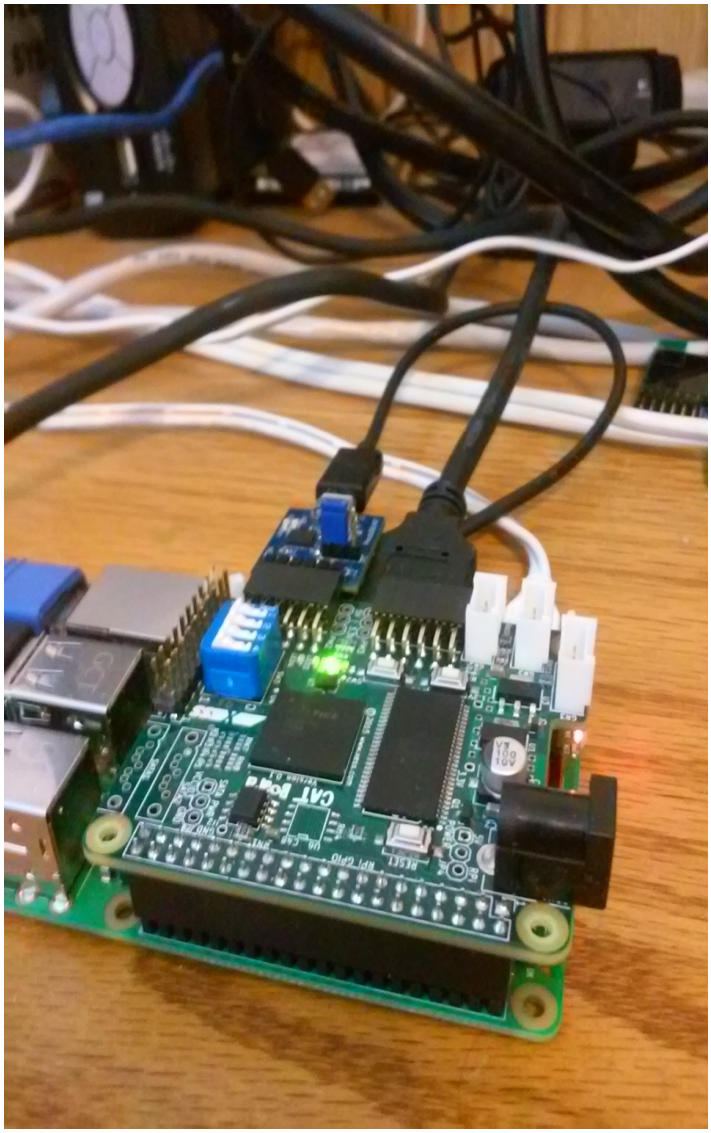
BANK 0 STATE : READING @ 2790
 SDRAM : [READ] Commnad registered
 BANK 0 STATE : READING FROM 01b8:440
 STATE : [READ] Data Ready @ 2794 value : 1500

In the image below reading the value at address 0x00000BB8 3000 which was 0x05DC 1500.



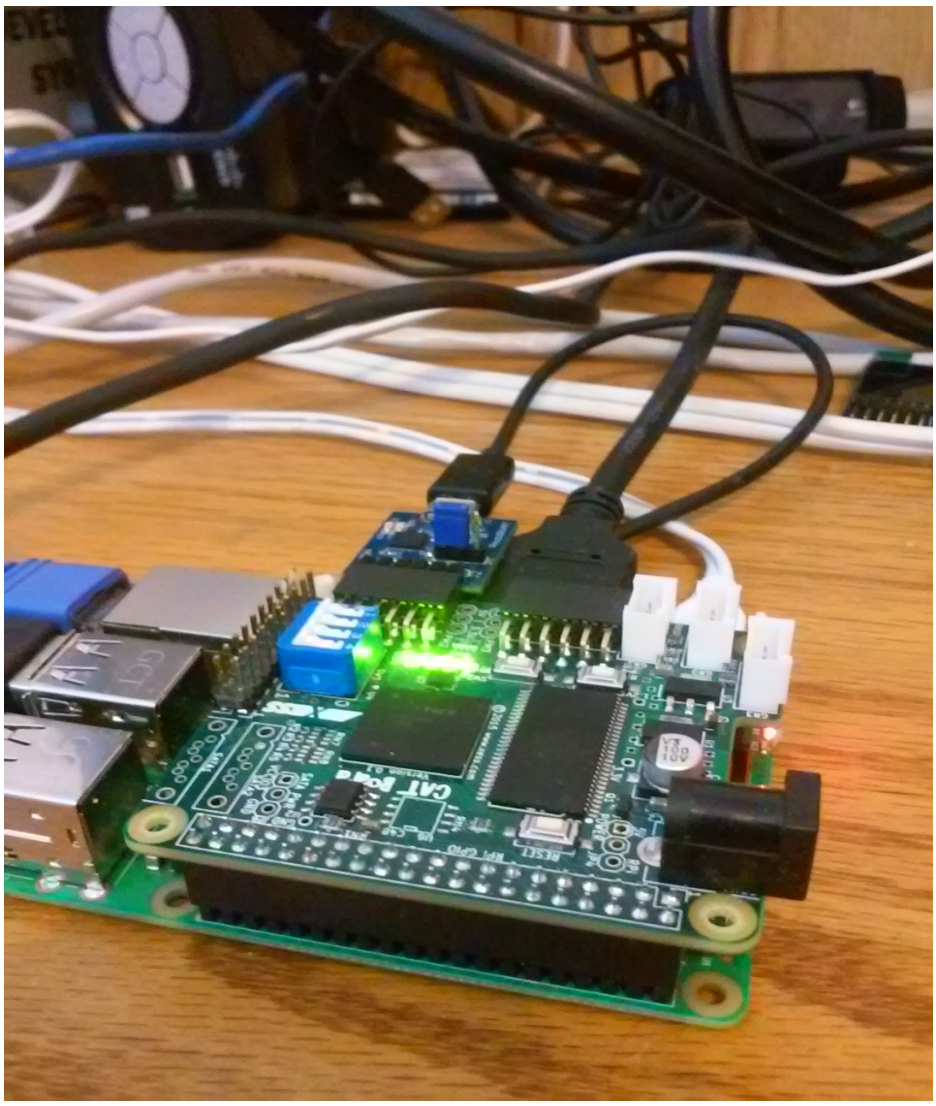
The memory_test takes approximately 3.5 sec or 175e6 clocks to complete at 50MHz. The memory_test writes to 16777215 locations and reads the data written.

Catboard running top.bin after the program has been load starting to write to memory during the memory_test



Catboard running top.bin after the program has been load starting to read from memory during the memory_test

Catboard running top.bin after the program has been load and the end memory memory_test. Note all 4 leds are on.



Now the sdram_test is memdev

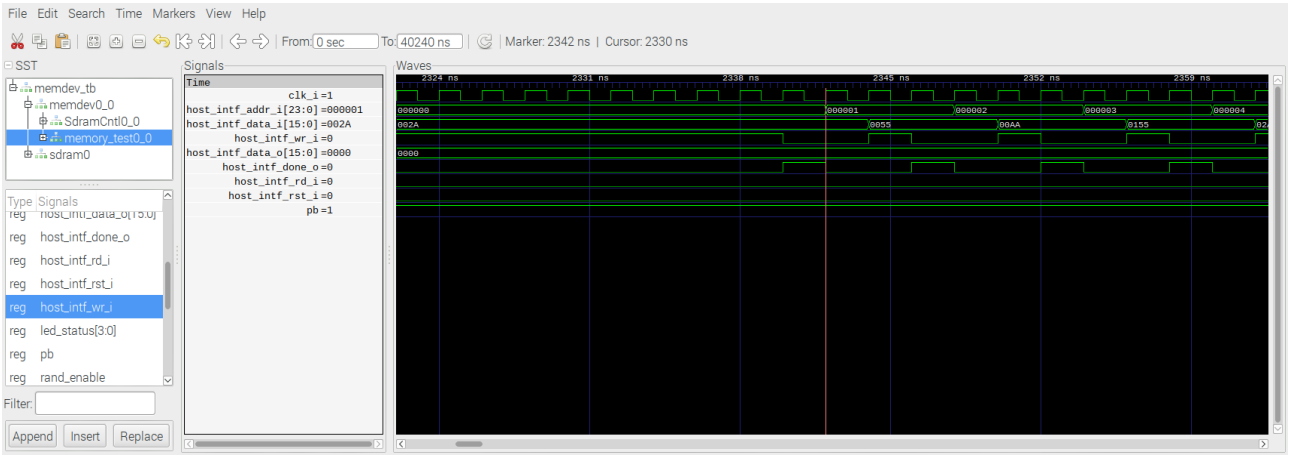
```
module top (
    clk100MHz,
    sdram_clk,
    sdram_return_clk,
    led_status,
    pb,
    memdev0_SdramCntl0_sd_intf_cke,
    memdev0_SdramCntl0_sd_intf_we,
    memdev0_SdramCntl0_sd_intf_addr,
    memdev0_SdramCntl0_sd_intf_dqml,
    memdev0_SdramCntl0_sd_intf_cas,
    memdev0_SdramCntl0_sd_intf_dqmh,
    memdev0_SdramCntl0_sd_intf_ras,
    memdev0_SdramCntl0_sd_intf_bs,
    memdev0_SdramCntl0_sd_intf_cs,
    memdev0_SdramCntl0_sd_intf_dq
);
=== top ===

   Number of wires:                441
```

```
   Number of wire bits:            968
   Number of public wires:          84
   Number of public wire bits:     537
   Number of memories:               0
   Number of memory bits:            0
   Number of processes:              0
   Number of cells:                719
     $_TBUF_                       16
     SB_CARRY                      68
     SB_DFF                         4
     SB_DFFE                        4
     SB_DFFER                      98
     SB_DFFES                       1
     SB_DFFESR                     45
     SB_DFFESS                      4
     SB_DFFR                       52
     SB_DFFS                        8
     SB_DFFSR                       3
     SB_DFFSS                       4
     SB_LUT4                      412
```

Host interface signals
wire [23:0] memdev0_SdramCntl0_host_intf_addr_i;
wire [15:0] memdev0_SdramCntl0_host_intf_data_i;
wire [15:0] memdev0_SdramCntl0_host_intf_data_i;
wire [15:0] memdev0_SdramCntl0_host_intf_data_o;
wire memdev0_SdramCntl0_host_intf_done_o;
wire memdev0_SdramCntl0_host_intf_wr_i;
wire memdev0_memory_test0_host_intf_rst_i;

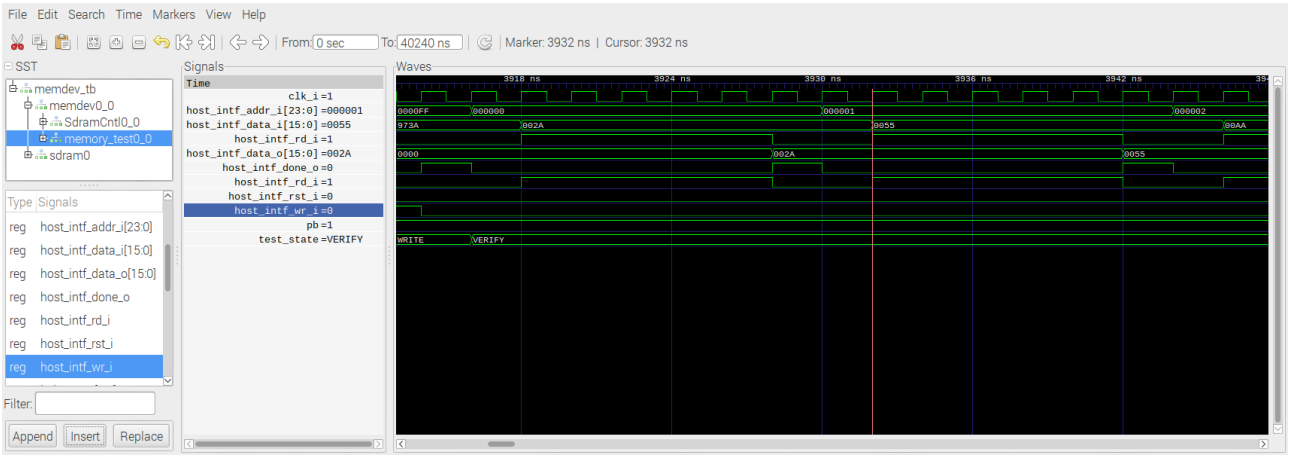
Write function
during a write
4 clks in between address chg
4 clks in between data chg
80 nsec

The 24 bit memdev0_SdramCntl0_host_intf_addr_i is set on the raising edge of clk. One clk later the 16 bit
memdev0_SdramCntl0_host_intf_data_i and memdev0_SdramCntl0_host_intf_wr_i goes hi. A clk later the memdev0_SdramCntl0_host_intf_done_o goes hi and the memdev0_SdramCntl0_host_intf_wr_i goes lo.



Read function

The 24 bit memdev0_SdramCntl0_host_intf_addr_i is set on the raising edge of clk. One clk later the 16 bit memdev0_SdramCntl0_host_intf_rd_i goes hi. Five clk later the memdev0_SdramCntl0_host_intf_done_o goes hi, memdev0_SdramCntl0_host_intf_data_o is valid and the memdev0_SdramCntl0_host_intf_rd_i goes lo.



During the first part of the memory_test Write takes place.

python memdev.py

make
sudo config_cat top.bin

2^24 16777216 0x000000 0xFFFFFF 16Bits
@PREFIX=sdram
@DEVID=SDRAM
@$LGMEMSZ=24
@LGMEMSZ.FORMAT=%d
@$NADDR=(1<<(@$THIS.LGMEMSZ-2))
@$NBYTES=(1<<(@$THIS.LGMEMSZ))
@NBYTES.FORMAT=0x%08x
@ACCESS=@$(DEVID)_ACCESS

```
@SLAVE.TYPE=MEMORY
@SLAVE.BUS=wb
@LD.PERM=wx
#!/bin/bash
```

Appendix A: Script that executes the arm-wbregs commands sim_hw_test.sh

```
#!/bin/bash


echo "The date built"
./arm-wbregs version
sleep 2
./arm-wbregs 0x700 0x10000001
sleep 2
./arm-wbregs 0x704 0x10000002
sleep 2
./arm-wbregs 0x708 0x10000003
sleep 2
./arm-wbregs 0x70c 0x10000004
sleep 2
./arm-wbregs 0x700
sleep 2
./arm-wbregs 0x704
sleep 2
./arm-wbregs 0x708
sleep 2
./arm-wbregs 0x70c
sleep 2
echo "Turning on the 4th led "
./arm-wbregs gpio 0x00010001
sleep 2
echo "Turning on the 1st led "
./arm-wbregs gpio 0x00020002
sleep 2
echo "Turning on the 2nd led "
./arm-wbregs gpio 0x00040004
sleep 5
echo "Turning off the leds "
./arm-wbregs gpio 0x00070000
```

Appendix B: CATZIP testing the verilator simulation.

```
./arm-main_tb
Listening on port 8363
Listening on port 8364
Accepted CMD connection
POLL = 1
RCVD: 6 bytes
< A611R
> A00000611R20180812
POLL = 1
RCVD: 0 bytes
```

< [CLOSED]
Accepted CMD connection
POLL = 1
RCVD: 14 bytes
< A701W10000001
> A00000701
> K00000000
POLL = 1
RCVD: 0 bytes
< [CLOSED]
Accepted CMD connection
POLL = 1
RCVD: 14 bytes
< A705W10000002
> A00000705
> K00000000
POLL = 1
RCVD: 0 bytes
< [CLOSED]
Accepted CMD connection
POLL = 1
RCVD: 14 bytes
< A709W10000003
> A00000709
> K00000000
POLL = 1
RCVD: 0 bytes
< [CLOSED]
Accepted CMD connection
POLL = 1
RCVD: 14 bytes
< A70dW10000004
> A0000070d
> K00000000
POLL = 1
RCVD: 0 bytes
< [CLOSED]
Accepted CMD connection
POLL = 1
RCVD: 6 bytes
< A701R
> A00000701R10000001
POLL = 1
RCVD: 0 bytes
< [CLOSED]
Accepted CMD connection
POLL = 1
RCVD: 6 bytes
< A705R
> A00000705R10000002
POLL = 1
RCVD: 0 bytes

< [CLOSED]
Accepted CMD connection
POLL = 1
RCVD: 6 bytes
< A709R
> A00000709R10000003
POLL = 1
RCVD: 0 bytes
< [CLOSED]
Accepted CMD connection
POLL = 1
RCVD: 6 bytes
< A70dR
> A0000070dR10000004
POLL = 1
RCVD: 0 bytes
< [CLOSED]
Accepted CMD connection
POLL = 1
RCVD: 11 bytes
< A609W10001
> A00000609K00000000
POLL = 1
RCVD: 0 bytes
< [CLOSED]
Accepted CMD connection
POLL = 1
RCVD: 11 bytes
< A609W20002
> A00000609K00000000
POLL = 1
RCVD: 0 bytes
< [CLOSED]
Accepted CMD connection
POLL = 1
RCVD: 11 bytes
< A609W40004
> A00000609K00000000
POLL = 1
RCVD: 0 bytes
< [CLOSED]
Accepted CMD connection
POLL = 1
RCVD: 11 bytes
< A609W70000
> A00000609K00000000
POLL = 1
RCVD: 0 bytes
< [CLOSED]
> Z
CMD: Only sent 0 bytes of 3
 ^C

Appendix C: CATZIP testing the verilator simulation running wbregs command from the script sim_hw_test.sh .

```
The date built
00000610 ( VERSION) : [....] 20180812
00000700 (     RAM)-> 10000001
00000704 (        )-> 10000002
00000708 (        )-> 10000003
0000070c (        )-> 10000004
00000700 (     RAM) : [....] 10000001
00000704 (        ) : [....] 10000002
00000708 (        ) : [....] 10000003
0000070c (        ) : [....] 10000004
Turning on the 4th led
00000608 (    GPIO)-> 00010001
Turning on the 1st led
00000608 (    GPIO)-> 00020002
Turning on the 2nd led
00000608 (    GPIO)-> 00040004
Turning off the leds
00000608 (    GPIO)-> 00070000
```