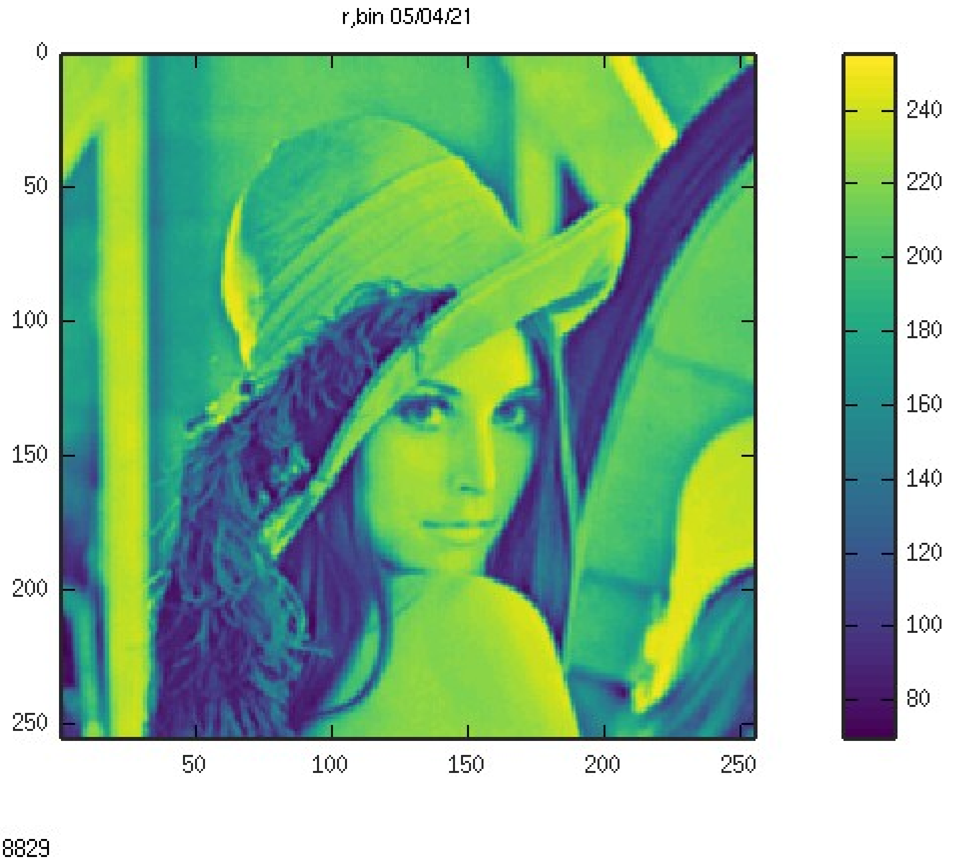
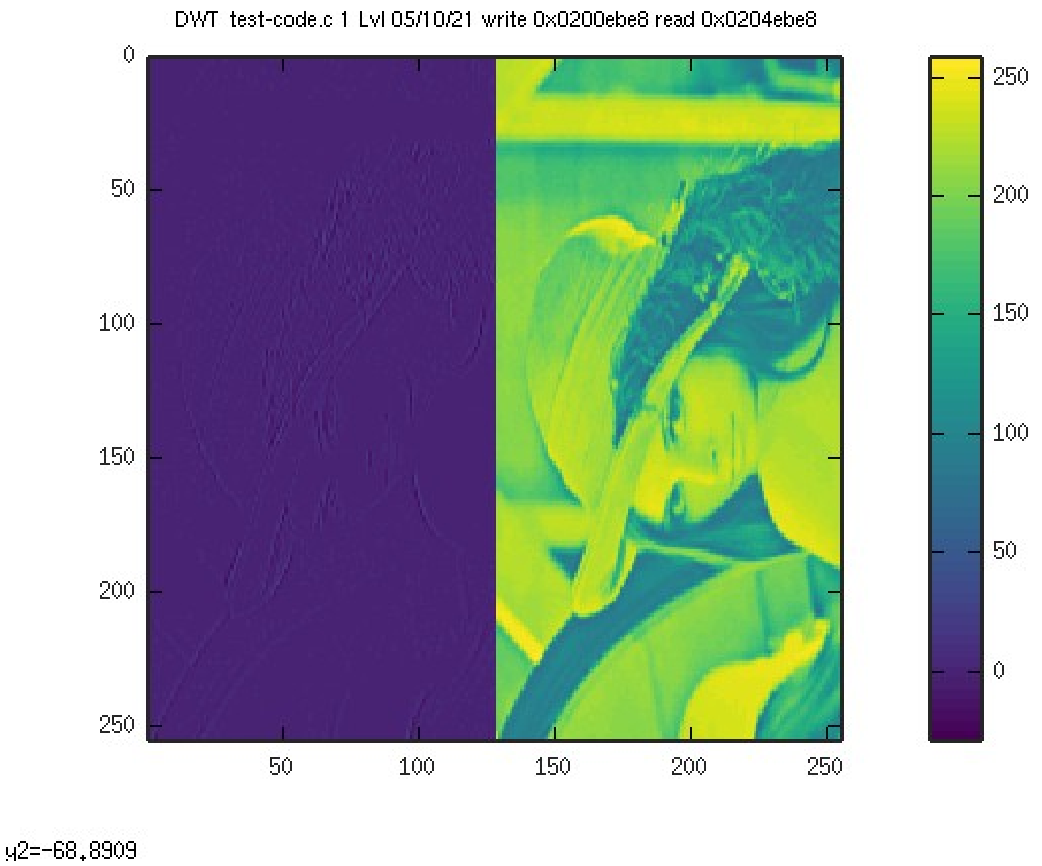
**************************Draft**************************
Lifting step using Verilator simulation
with test-code.c instead of jpeg.c
This is needed since the printf is not working correctly in simulation
These steps are not working on the iCE 40 HX8K FPGA with Rpi3B+.

05/10/21
**************************Draft**************************

Input image





Create the jpep zipcpu C program. The command to compile and create the dissambly **“cp test-code.c jpeg.c; make;zip-objdump -d jpeg > jpeg-disasm.txt”** in the **“~/testbuilds/icozip-catzip-br/icozip/sw/board”** folder.

The file **“jpeg-disasm.txt”** is used to find where the struct ptrs is in memory. Towards the end of file **“0200ec14 <ptrs>:”** is found.

Starting the simulator. With the following command ./arm-main_tb in the ~/testbuilds/icozip-catzip-br/icozip/sim/verilated folder.

Set 3 values in bkram with the following command.
“../host/arm-wbregs 0x01401000 2; ../host/arm-wbregs 0x01401008 2; ../host/arm-wbregs 0x01401004 1” in the **“~/testbuilds/icozip-catzip-br/icozip/sw/board”** folder.

```
01401000 (        )-> 00000002
01401008 (        )-> 00000002
01401004 (        )-> 00000001
```

The value at location **“0x01401008”** is used to prevent the program from performing the lifting step before the data in **“r.bin”** is written to sdram with the command **“./arm-wrsdram r.bin”** in **“~/testbuilds/icozip-catzip-br/icozip/sw/host”** folder.

Load the program jpeg into the simulator with the following command.
“./arm-zipload -v ../board/jpeg;./arm-wbregs cpu 0x0f;./test-code.sh” in **“~/testbuilds/icozip-catzip-br/icozip/sw/host”** folder.   The script “test-code.sh” is used in conjuntion with **“0200ec14 <ptrs>:”** is used to display values from the struct ptrs.

```
#!/bin/bash
```

```
echo "inpbuf"
./arm-wbregs 0x0200ec14
        COUNTER=10
          ADDRESS=0x0200ec14
        until [  $COUNTER -lt 1 ]; do
           echo COUNTER $COUNTER
           let COUNTER-=1
              let ADDRESS+=4
              ./arm-wbregs $ADDRESS
        done
```

The first 2 values are the w & h of the image. The 3rd value is the

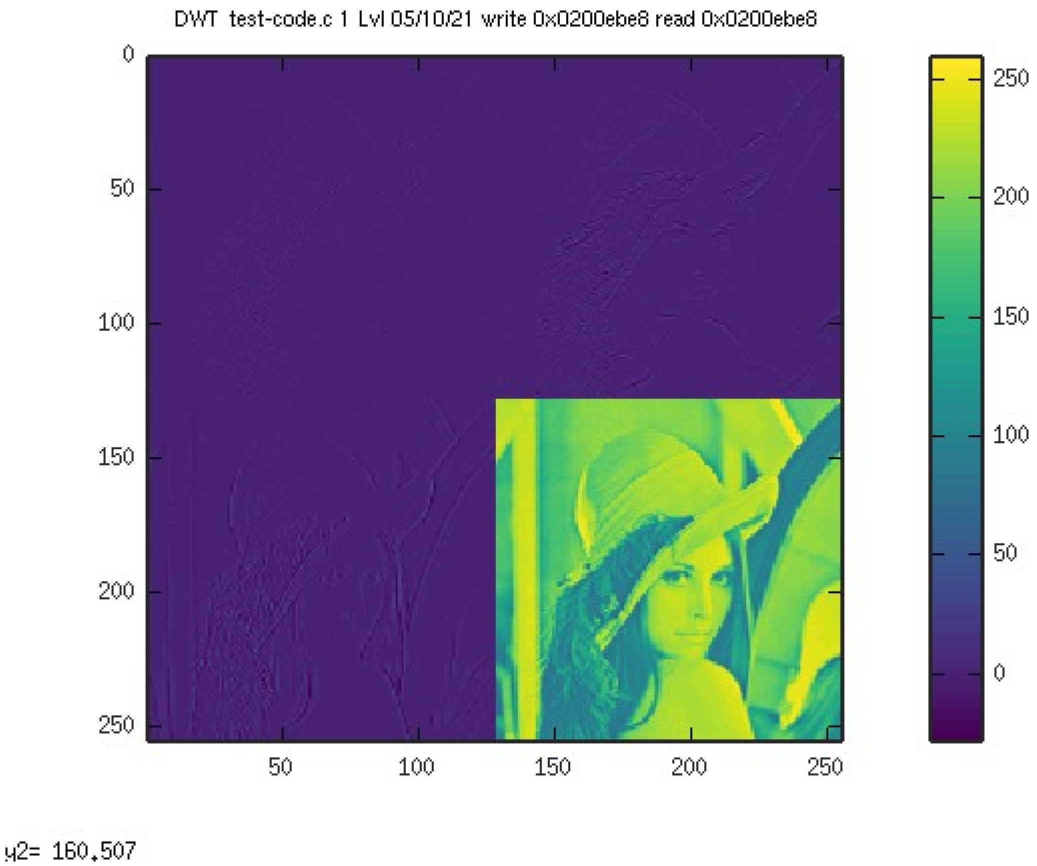
```
while(ptrs.status==2) {
                ptrs.status = ptrs.ptr_blkram_status[0];
          }
```

to prevent the program from performing the lifting step before the data in **“r.bin”** is written to sdram with the command **“./arm-wrsdram r.bin”** in **“~/testbuilds/icozip-catzip-br/icozip/sw/host”** folder. The 4th value is the result of the ptrs.buf_red = (int *)malloc(sizeof(int)* ptrs.w*ptrs.h*2);. While the 5th value is where the lifting step results are found. These pointers are used in **“wrsdram.cpp”** and **“rdsdram.cpp”.**

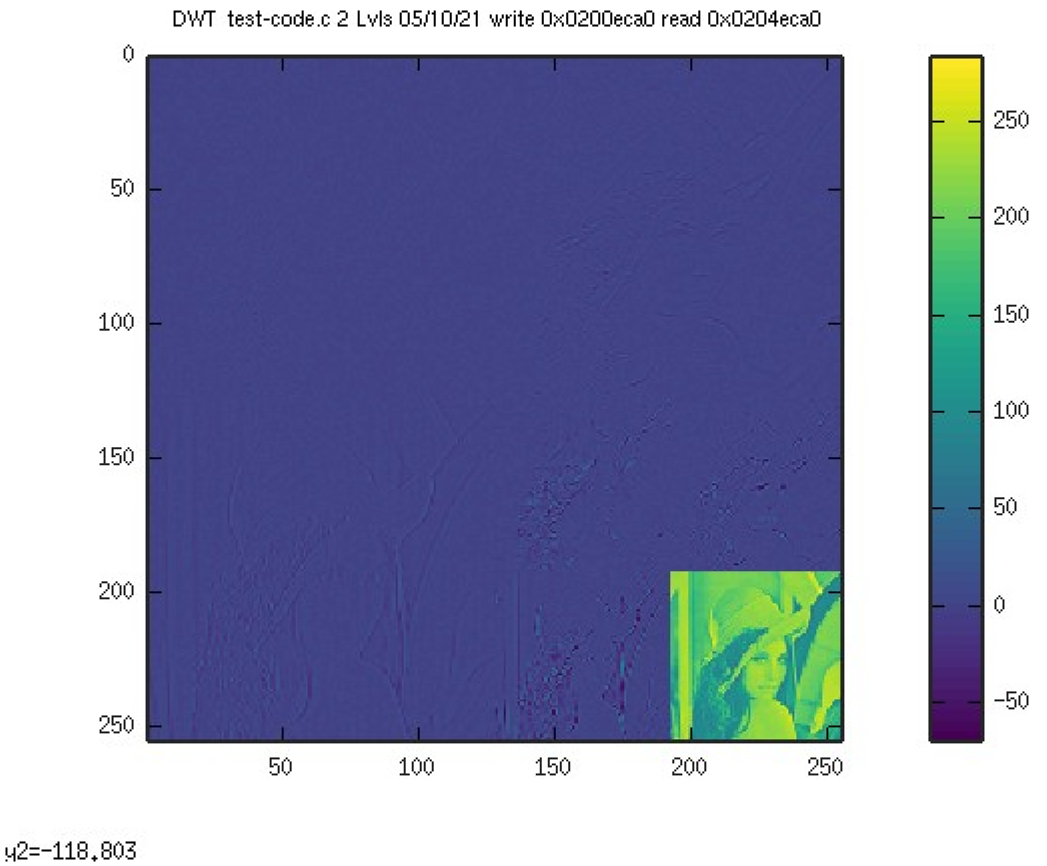
```
inpbuf
0200ec14 (        ) : [....] 00000100
COUNTER 10
0200ec18 (        ) : [....] 00000100
COUNTER 9
0200ec1c (        ) : [....] 00000002
COUNTER 8
0200ec20 (        ) : [....] 0200eca0
COUNTER 7
0200ec24 (        ) : [....] 0204eca0
COUNTER 6
0200ec28 (        ) : [....] 00000001
COUNTER 5
0200ec2c (        ) : [.@..] 01401000
COUNTER 4
0200ec30 (        ) : [.@..] 01401004
COUNTER 3
0200ec34 (        ) : [.@..] 01401008
COUNTER 2
0200ec38 (        ) : [....] 00000002
COUNTER 1
0200ec3c (        ) : [....] 00000000
```

Once the data has been loaded in sdram the signal to start the lifting step is provided with command **“../host/arm-wbregs 0x01401008 1”** in the folder **“~/testbuilds/icozip-catzip-br/icozip/sw/board”.**

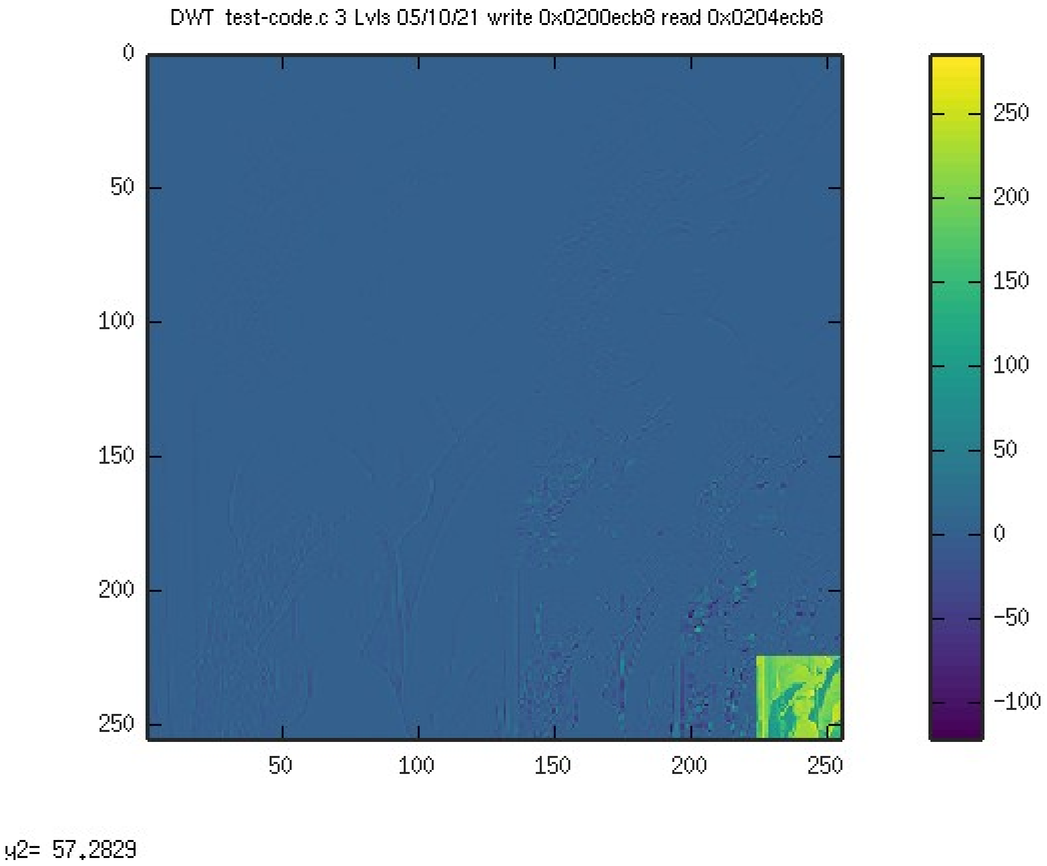
The first level of an image 256x256 is the 128x128.



The 2 level of an image 256x256 is the 64x64.



The 3 levels of an image 256x256 is the 32x32.


DWT test-code.c 3 Lvls 05/10/21 write 0x0200ecb8 read 0x0204ecb8
y2= 57.2829