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

Multi (5) QEMU Ultibo Bare Metal
JPEG2000 with different Compression Ratios
25:1, 35:1, 45:1, 50:1 and 100:1
with
Remote Shell & Webstatus
08/02/21

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

A greater detailed on the above steps is found in QEMU-JPEG2000.odt.

There is very little visual difference in the decompresed images.

```
-rw-r--r-- 1 devel devel 7848 Aug  1 15:43 25.j2k
-rw-r--r-- 1 devel devel 5631 Aug  1 15:39 35.j2k
-rw-r--r-- 1 devel devel 4352 Aug  1 16:48 45.j2k
-rw-r--r-- 1 devel devel 3936 Aug  1 12:36 50.j2k
-rw-r--r-- 1 devel devel 1972 Aug  1 12:00 100.j2k
```

The size of different j2k files is reduced significatanly from 7848 to 1972.
This would improve the data transfer of the images.
The original image

```
-rw-r--r--  1 devel devel   196730 Jul 28 12:47 MyBitmap.bmp
```

Note:
https://ultibo.org/forum/viewtopic.php?f=13&t=1303&p=11632#p11632

By Ultibo Wed Jul 21, 2021 9:01 pm

I suspect the version of QEMU that you have on the RPI3B+ is later than the one on the RPi4, try doing qemu-system-arm -version on each one.

We recently discovered that the Ultibo SD card driver was not compatible with the latest versions of QEMU, a fix for this is included in the release from today (Ultibo core 2.1.079) so if you update your RTL to the latest either using the RTL Builder or by rerunning the ultiboinstaller script then it should work now.
https://en.m.wikipedia.org/wiki/QEMU. On the pi400-1 I ran ./ultiboinstaller.sh on pi400-1.

QEMU is a hosted virtual machine monitor: it emulates the machine's processor through dynamic binary translation and provides a set of different hardware and device models for the machine, enabling it to run a variety of guest operating systems. It also can be used with Kernel-based Virtual Machine (KVM) to run virtual machines at near-native speed (by taking

advantage of hardware extensions such as Intel VT-x). QEMU can also do emulation for user-level processes, allowing applications compiled for one architecture to run on another.[3]

Note : Additional software is needed to run QEMU "sudo apt-get install qemu-system-arm". The following programs are added.

/usr/bin/qemu-img /usr/bin/qemu-nbd /usr/bin/qemu-system-aarch64
/usr/bin/qemu-io /usr/bin/qemu-pr-helper /usr/bin/qemu-system-arm

The command line for starting **Lazarus IDE (Ultibo Edition) "~/ultibo/core/lazarus.sh"**

Requirements: 5 QEMU Ultibo systems running DWT_LIFT_QEMU.lpi with different testfile. The testfile determines the Compression Ratio 1st 4 bytes , weather to Encode / Decode bytes 0004-0007 , the origin of image bytes 0007-000f, and the size of image .bytes 0010-0017
QEMU testfile where 19 hex is 25.

```
00000000  19 00 00 00  01 00 00 00  00 00 00 00  00 00 00 00  #...............
00000010  00 01 00 00  00 01 00 00
```

QEMU1 testfile where 23 hex is 35.

```
00000000  23 00 00 00  01 00 00 00  00 00 00 00  00 00 00 00  #...............
00000010  00 01 00 00  00 01 00 00
```

QEMU2 testfile where 2D hex is 45

```
00000000  2D 00 00 00  01 00 00 00  00 00 00 00  00 00 00 00  -...............
00000010  00 01 00 00  00 01 00 00
```

QEMU3 testfile where 32 hex is 50

```
00000000  32 00 00 00  01 00 00 00  00 00 00 00  00 00 00 00  2...............
00000010  00 01 00 00  00 01 00 00
```

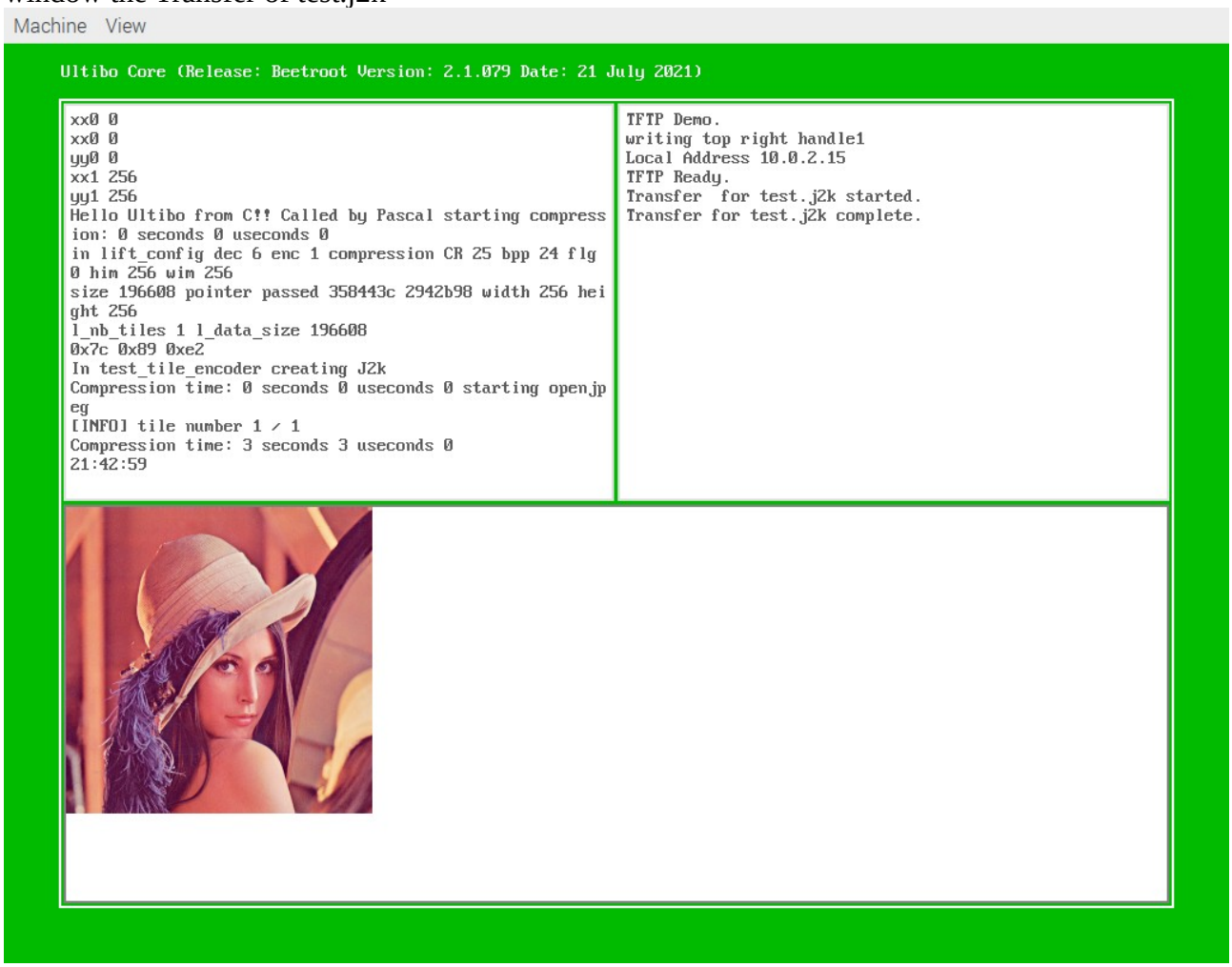
QEMU4 testfile where 64 hex is 100

```
00000000  64 00 00 00  01 00 00 00  00 00 00 00  00 00 00 00  d...............
00000010  00 01 00 00  00 01 00 00
```

startqemu.sh 25:1 QEMU

```
#!/bin/bash
qemu-system-arm -machine versatilepb -cpu cortex-a8 -kernel kernel.bin \
-net
user,hostfwd=tcp::5080-:80,hostfwd=tcp::5023-:23,hostfwd=udp::5069-:69,hostfwd=tcp::6050-:505
0 -net nic \
 -drive file=disk.img,if=sd,format=raw
```

See commpression CR 25 8 lines from top below in top left window. An in the top right hand window the Transfer of test.j2k



devel@pi400-1:~/Ultibo_Projects/jpeg2000/QEMU $ tftp pi400-1 5069
tftp> binary
tftp> trace
Packet tracing on.
tftp> get test.j2k 25.j2k
sent RRQ <file=test.j2k, mode=octet>
received DATA <block=1, 512 bytes>
sent ACK <block=1>
received DATA <block=2, 512 bytes>
sent ACK <block=2>
received DATA <block=3, 512 bytes>
sent ACK <block=3>
received DATA <block=4, 512 bytes>
sent ACK <block=4>
received DATA <block=5, 512 bytes>
sent ACK <block=5>
received DATA <block=6, 512 bytes>
sent ACK <block=6>
received DATA <block=7, 512 bytes>

sent ACK <block=7>
received DATA <block=8, 512 bytes>
sent ACK <block=8>
received DATA <block=9, 512 bytes>
sent ACK <block=9>
received DATA <block=10, 512 bytes>
sent ACK <block=10>
received DATA <block=11, 512 bytes>
sent ACK <block=11>
received DATA <block=12, 512 bytes>
sent ACK <block=12>
received DATA <block=13, 512 bytes>
sent ACK <block=13>
received DATA <block=14, 512 bytes>
sent ACK <block=14>
received DATA <block=15, 512 bytes>
sent ACK <block=15>
received DATA <block=16, 168 bytes>
Received 7848 bytes in 0.2 seconds
tftp>

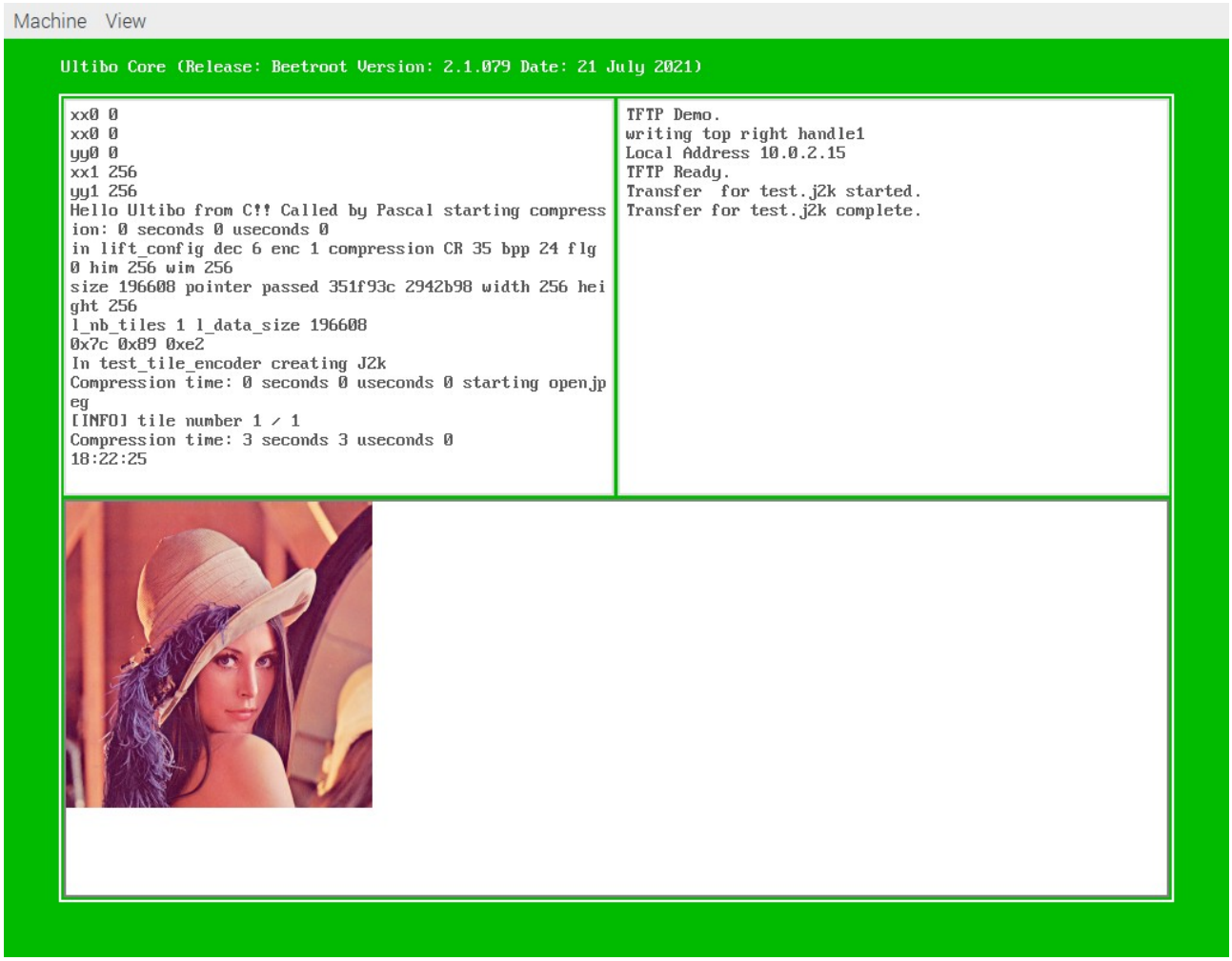
devel@pi400-1:~/Ultibo_Projects/jpeg2000/QEMU $ ~/t_ultibo/build/bin/opj_decompress -i 25.j2k -o 25.bmp

[INFO] Start to read j2k main header (0).
[INFO] Main header has been correctly decoded.
[INFO] No decoded area parameters, set the decoded area to the whole image
[INFO] Header of tile 1 / 1 has been read.
[INFO] Generated Outfile 25.bmp
decode time: 21 ms

startqemu1.sh 35:1 QEMU1

#!/bin/bash
qemu-system-arm -machine versatilepb -cpu cortex-a8 -kernel kernel.bin \
-net
user,hostfwd=tcp::7080-:80,hostfwd=tcp::7023-:23,hostfwd=udp::7069-:69,hostfwd=tcp::7050-:5050 -net nic \
 -drive file=disk.img,if=sd,format=raw

See commpression CR 35 8 lines from top below in top left window. An in the top right hand window the Transfer of test.j2k



devel@pi400-1:~/Ultibo_Projects/jpeg2000/QEMU $ tftp pi400-1 7069
tftp> binary
tftp> trace
Packet tracing on.
tftp> get test.j2k 35.j2k
sent RRQ <file=test.j2k, mode=octet>
received DATA <block=1, 512 bytes>
sent ACK <block=1>
received DATA <block=2, 512 bytes>
sent ACK <block=2>
received DATA <block=3, 512 bytes>
sent ACK <block=3>
received DATA <block=4, 512 bytes>
sent ACK <block=4>
received DATA <block=5, 512 bytes>
sent ACK <block=5>
received DATA <block=6, 512 bytes>
sent ACK <block=6>
received DATA <block=7, 512 bytes>
sent ACK <block=7>devel@pi400-1:~/Ultibo_Projects/jpeg2000/QEMU $ tftp pi400-1 7069
tftp> binary
tftp> trace

Packet tracing on.
tftp> get test.j2k 35.j2k
sent RRQ <file=test.j2k, mode=octet>
received DATA <block=1, 512 bytes>
sent ACK <block=1>
received DATA <block=2, 512 bytes>
sent ACK <block=2>
received DATA <block=3, 512 bytes>
sent ACK <block=3>
received DATA <block=4, 512 bytes>
sent ACK <block=4>
received DATA <block=5, 512 bytes>
sent ACK <block=5>
received DATA <block=6, 512 bytes>
sent ACK <block=6>
received DATA <block=7, 512 bytes>
sent ACK <block=7>
received DATA <block=8, 512 bytes>
sent ACK <block=8>
received DATA <block=9, 512 bytes>
sent ACK <block=9>
received DATA <block=10, 512 bytes>
sent ACK <block=10>
received DATA <block=11, 511 bytes>
Received 5631 bytes in 0.8 seconds
tftp>
received DATA <block=8, 512 bytes>
sent ACK <block=8>
received DATA <block=9, 512 bytes>
sent ACK <block=9>
received DATA <block=10, 512 bytes>
sent ACK <block=10>
received DATA <block=11, 511 bytes>
Received 5631 bytes in 0.8 seconds
tftp>

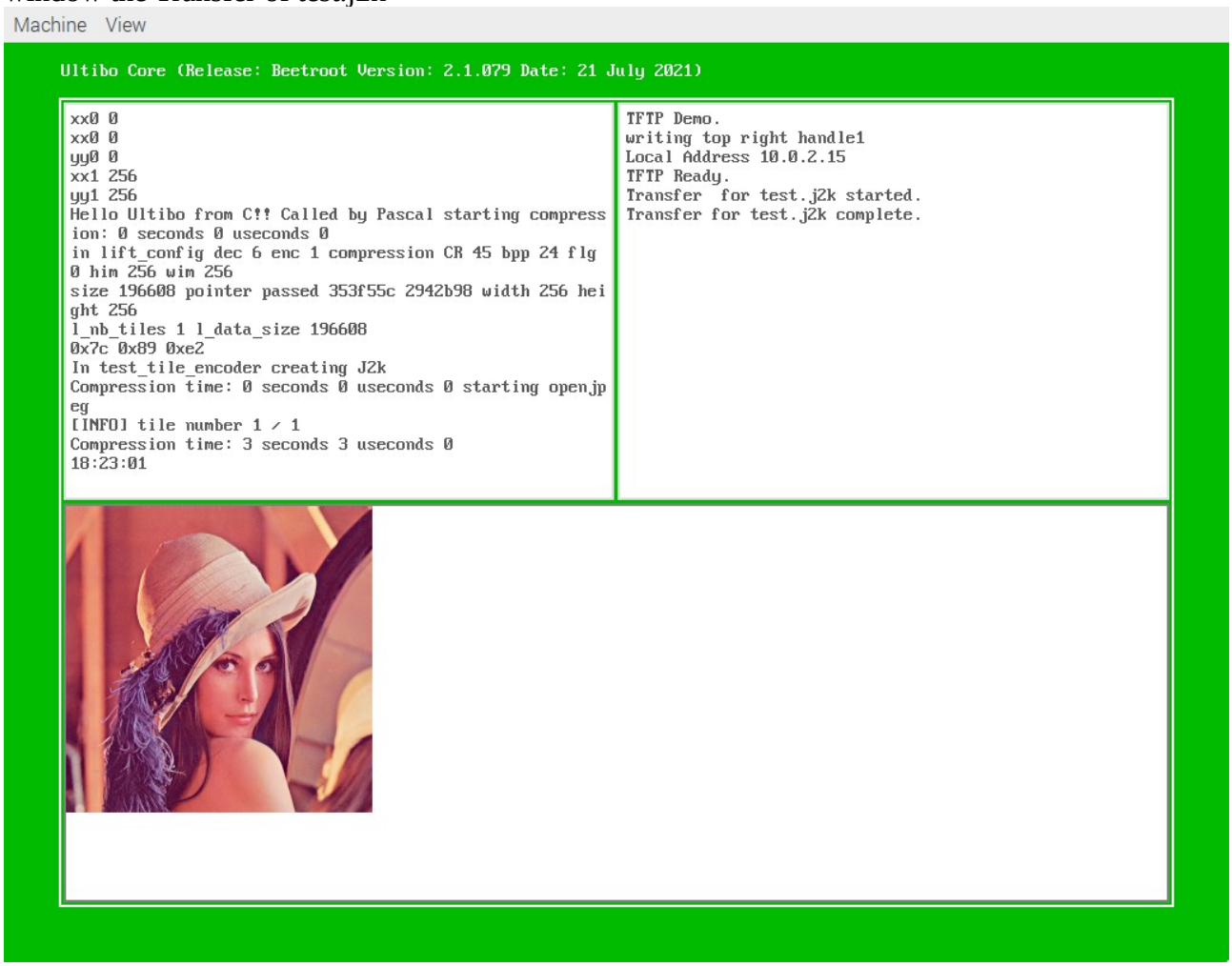
devel@pi400-1:~/Ultibo_Projects/jpeg2000/QEMU $ ~/t_ultibo/build/bin/opj_decompress -i 35.j2k -o 35.bmp

[INFO] Start to read j2k main header (0).
[INFO] Main header has been correctly decoded.
[INFO] No decoded area parameters, set the decoded area to the whole image
[INFO] Header of tile 1 / 1 has been read.
[INFO] Generated Outfile 35.bmp
decode time: 19 ms

startqemu2.sh 45:1 QEMU2

```
#!/bin/bash
qemu-system-arm -machine versatilepb -cpu cortex-a8 -kernel kernel.bin \
-net
user,hostfwd=tcp::8080-:80,hostfwd=tcp::8023-:23,hostfwd=udp::8069-:69,hostfwd=tcp::8050-:505
0 -net nic \
 -drive file=disk.img,if=sd,format=raw
```

See commpression CR 45 8 lines from top below in top left window. An in the top right hand window the Transfer of test.j2k



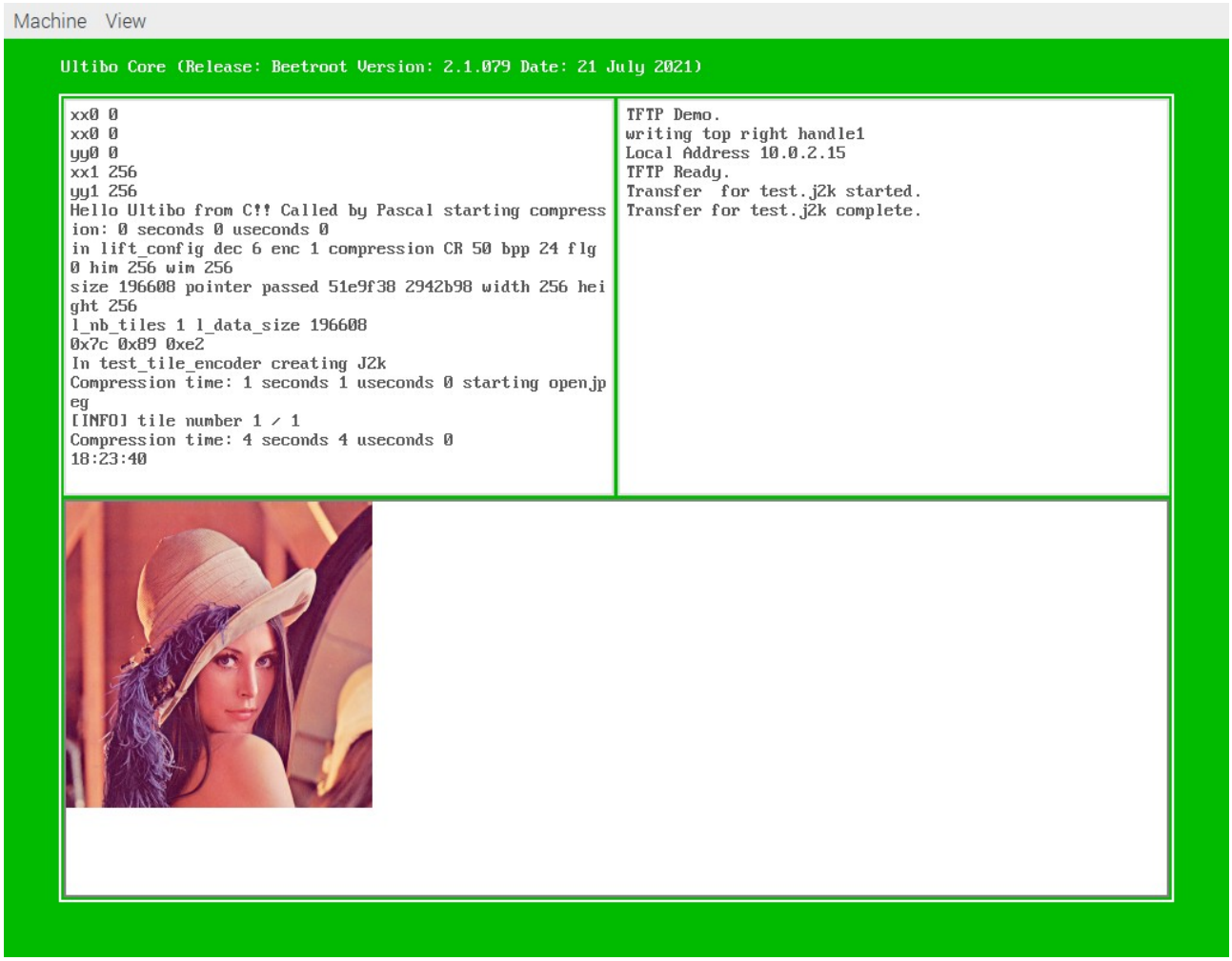

```
devel@pi400-1:~/Ultibo_Projects/jpeg2000/QEMU $ tftp  pi400-1 8069
tftp> binary
tftp> trace
Packet tracing on.
tftp> get test.j2k 45.j2k
sent RRQ <file=test.j2k, mode=octet>
received DATA <block=1, 512 bytes>
sent ACK <block=1>
received DATA <block=2, 512 bytes>
sent ACK <block=2>
received DATA <block=3, 512 bytes>
```

sent ACK <block=3>
received DATA <block=4, 512 bytes>
sent ACK <block=4>
received DATA <block=5, 512 bytes>
sent ACK <block=5>
received DATA <block=6, 512 bytes>
sent ACK <block=6>
received DATA <block=7, 512 bytes>
sent ACK <block=7>
received DATA <block=8, 512 bytes>
sent ACK <block=8>
received DATA <block=9, 256 bytes>
Received 4352 bytes in 0.2 seconds
tftp>

startqemu3.sh 50:1 QEMU3

```
#!/bin/bash
qemu-system-arm -machine versatilepb -cpu cortex-a8 -kernel kernel.bin \
-net
user,hostfwd=tcp::9080-:80,hostfwd=tcp::9023-:23,hostfwd=udp::9069-:69,hostfwd=tcp::9050-:505
0 -net nic \
 -drive file=disk.img,if=sd,format=raw
```

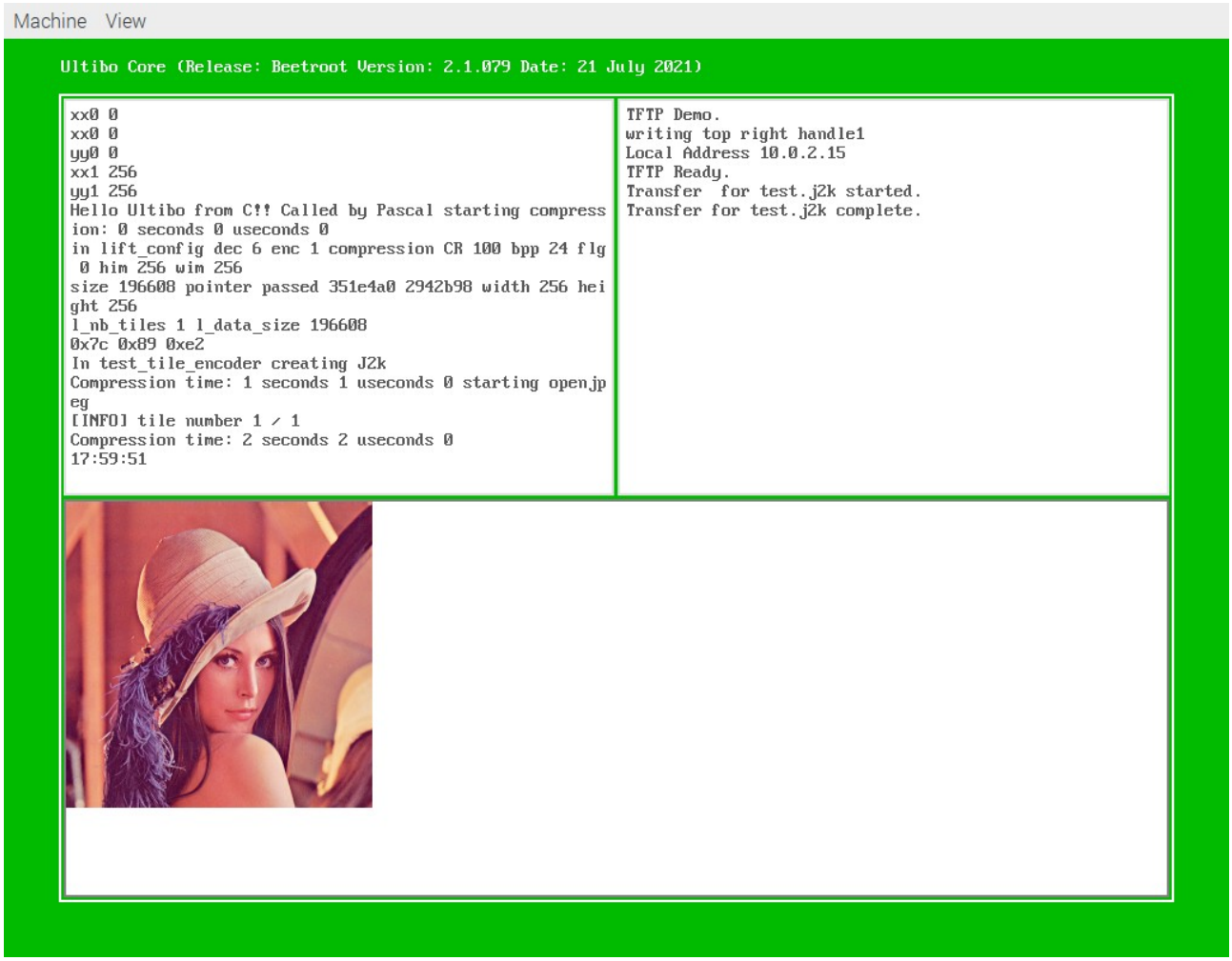
See commpression CR 50 8 lines from top below in top left window. An in the top right hand window the Transfer of test.j2k



startqemu4.sh 100:1 QEMU4

```
#!/bin/bash
qemu-system-arm -machine versatilepb -cpu cortex-a8 -kernel kernel.bin \
-net
user,hostfwd=tcp::11080-:80,hostfwd=tcp::11023-:23,hostfwd=udp::11069-:69,hostfwd=tcp::11050
-:5050 -net nic \
 -drive file=disk.img,if=sd,format=raw
```

See commpression CR 100 8 lines from top below in top left window. An in the top right hand window the Transfer of test.j2k



devel@pi400-1:~/Ultibo_Projects/jpeg2000/QEMU $ tftp pi400-1 11069
tftp> binary
tftp> trace
Packet tracing on.
tftp> get test.j2k 100.j2k
sent RRQ <file=test.j2k, mode=octet>
received DATA <block=1, 512 bytes>
sent ACK <block=1>
received DATA <block=2, 512 bytes>
sent ACK <block=2>
received DATA <block=3, 512 bytes>
sent ACK <block=3>
received DATA <block=4, 436 bytes>

devel@pi400-1:~/Ultibo_Projects/jpeg2000/QEMU $ ~/t_ultibo/build/bin/opj_decompress -i 100.j2k -o 100.bmp

[INFO] Start to read j2k main header (0).
[INFO] Main header has been correctly decoded.
[INFO] No decoded area parameters, set the decoded area to the whole image
[INFO] Header of tile 1 / 1 has been read.
[INFO] Generated Outfile 100.bmp
decode time: 50 ms