

Testing step for EPROM with baseline firmware

Hualin Mei

Introduction

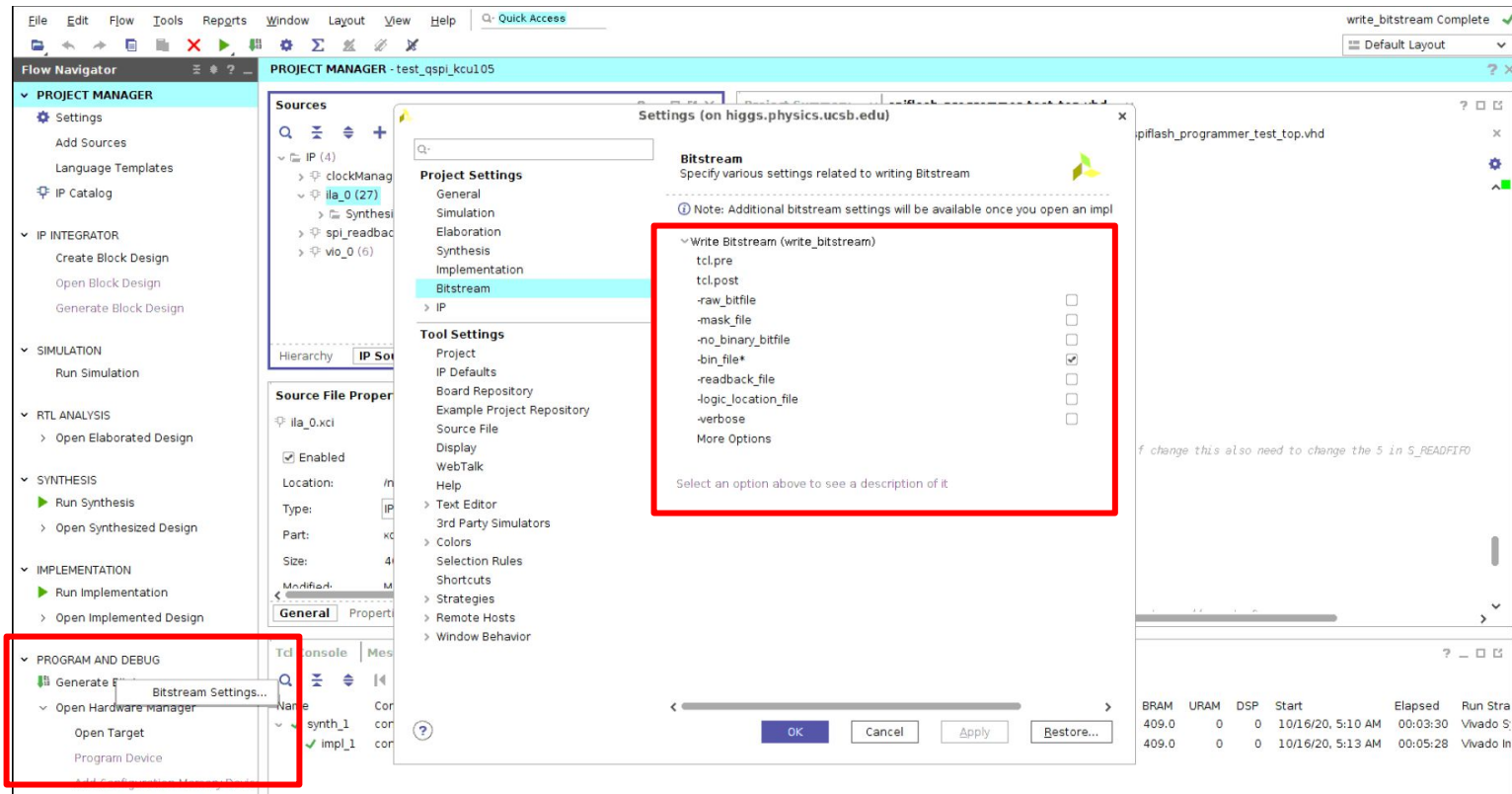
There are two ways verifying interaction between FPGA and EPROM

1. Use Vivado to load firmware (.bin) to EPROM, then configure FPGA from EPROM
2. Use dedicated firmware to evaluate read/erase/write operation on EPROM

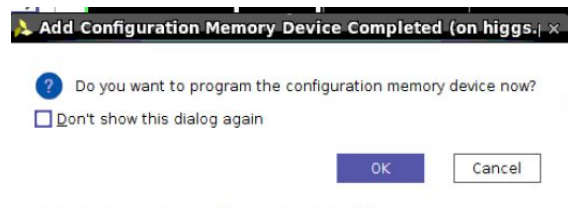
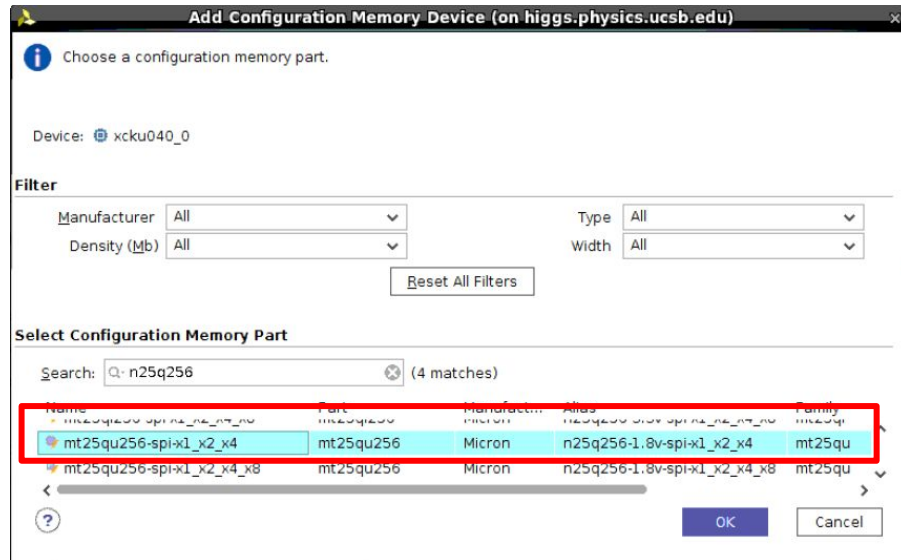
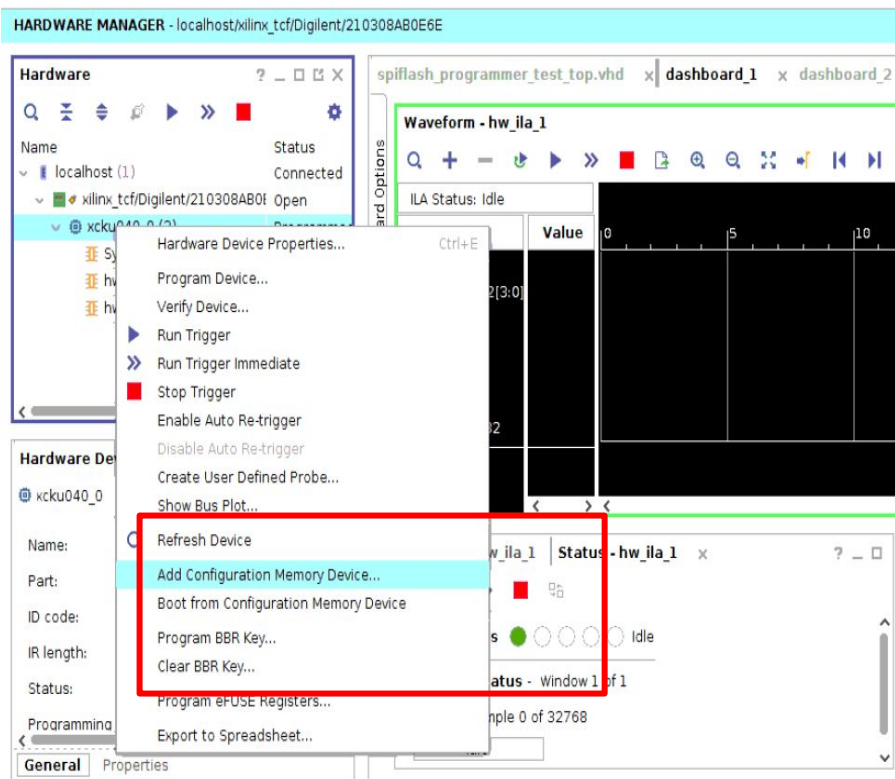
We can try to do both when testing the first pre-production boards

Produce firmware (.bin and .bit)

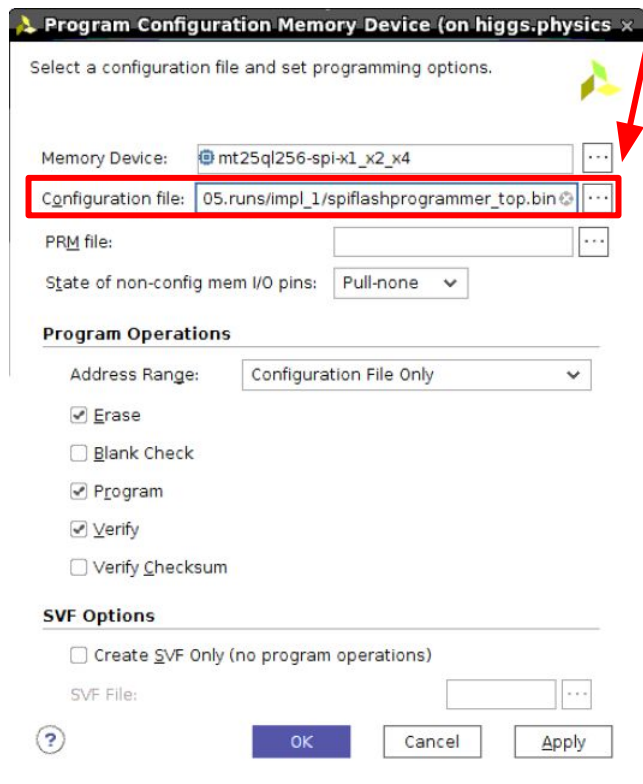
Select `-bin_file` in
bitstream setting
before generate
bitfile



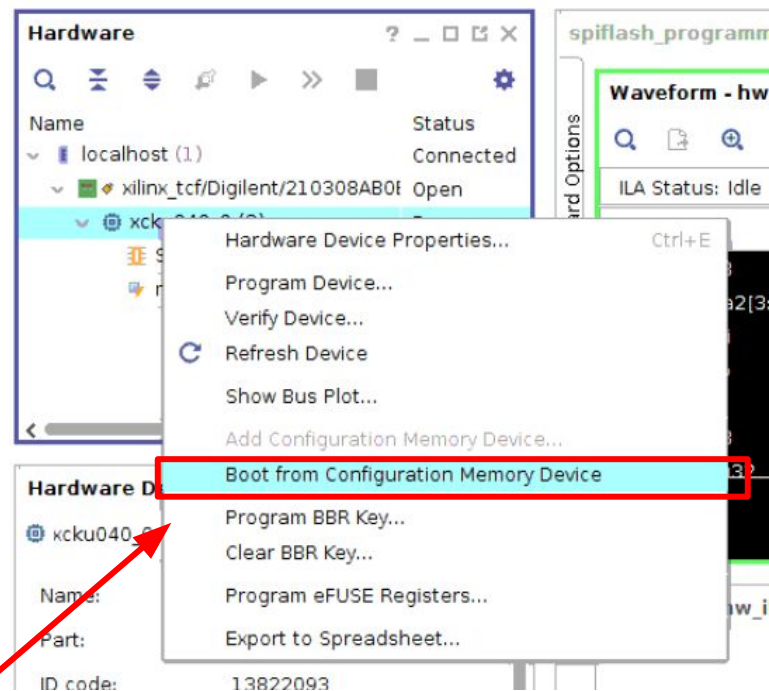
Open hardware manager, and add EPROM



Program EPROM with firmware



Specify location of .bin file



Load firmware to FPGA

Scripts for EPROM testing

master vivadoProjects / tcls / epromtesting /

Go to file Add file

mhl0116 fully working on kcu105 54b73ff in 5 minutes History

..		
doeprom.sh	add script to do read/erase/write	7 days ago
eraseEPROM.tcl	working for write	7 days ago
readEPROM.tcl	working for write	7 days ago
testeprom_all.sh	fully working on kcu105	now
writeEPROM.tcl	working for write	

```
0 Oct 23 08:39 epromaftererase_20201023_v1.txt
100M Oct 23 08:54 epromafterwrite_20201023_v1.txt
21M Oct 23 08:26 fwcontent_20201023_v1.txt
21M Oct 23 08:26 fwcontent_rdback_20201023_v1.txt
1.4K Oct 23 08:53 result_20201023_v1.txt
```

- Push one button, testprom_all.sh

- First read content of EPROM, compare with content of firmware loaded to EPROM
- Erase entire content of EPROM, read content of EPROM, verify they are all FFFF
- Write simple counter into EPROM, read content of EPROM, compare with expectation

Final result

This is the end of fw, I
readback until one line above ←

Empty is good, it means
whatever in eprom are all
FFFF ←

Currently a counter is
sent to EPROM, the
content of counter will be
updated (to a bit more
complex version), then
will also add a script to
compare ←

```
Fri Oct 23 08:29:15 PDT 2020 [INFO] Start read EPROM
1 Fri Oct 23 08:31:52 PDT 2020 [INFO] Read EPROM finished
2 Fri Oct 23 08:31:57 PDT 2020 [INFO] Compare content readback from EPROM to content of original firmware
3 447428d447427
4 < 06d3c30: 2000 0000 2000 0000 2000 0000 ... ..
5 Fri Oct 23 08:31:57 PDT 2020 [INFO] Start erase EPROM
6 Fri Oct 23 08:33:17 PDT 2020 [INFO] Erase EPROM finished
7 Fri Oct 23 08:33:17 PDT 2020 [INFO] Start read EPROM
8 Fri Oct 23 08:44:55 PDT 2020 [INFO] Read EPROM finished
9 Fri Oct 23 08:44:56 PDT 2020 [INFO] Check if there is only FFFF in EPROM
10 check if there is 0 in EPROM
11 check if there is 1 in EPROM
12 check if there is 2 in EPROM
13 check if there is 3 in EPROM
14 check if there is 4 in EPROM
15 check if there is 5 in EPROM
16 check if there is 6 in EPROM
17 check if there is 7 in EPROM
18 check if there is 8 in EPROM
19 check if there is 9 in EPROM
20 check if there is A in EPROM
21 check if there is B in EPROM
22 check if there is C in EPROM
23 check if there is D in EPROM
24 check if there is E in EPROM
25 Fri Oct 23 08:44:57 PDT 2020 [INFO] Start write EPROM
26 Fri Oct 23 08:45:54 PDT 2020 [INFO] Write EPROM finished
27 Fri Oct 23 08:47:45 PDT 2020 [INFO] Start read EPROM
28 Fri Oct 23 08:59:32 PDT 2020 [INFO] Read EPROM finished
```