

OsmoGSMTester - Support #3451

install LimeSDR-USB in 1U enclosure to production setup

08/07/2018 05:58 PM - laforge

Status:	Resolved	Start date:	08/07/2018
Priority:	Normal	Due date:	
Assignee:	pespin	% Done:	100%
Category:			
Target version:			
Spec Reference:			

Description

Some months ago we have built a 1U enclosed version for the osmo-gsm-tester setup at sysmocom. This unit for some reason has not yet been installed to the actual setup. I don't know why. Please add it soon, and hand this ticket over to pespin so he starts to use it from the tests.

History

#1 - 08/08/2018 05:27 PM - roh

- Status changed from New to In Progress
- Assignee changed from roh to pespin
- % Done changed from 0 to 50

i think the reasoning was that pespin wanted to get lime working locally (notebook) first.

i mounted the unit in the tester now.

rfport 1 is connected to the tester, 2 is unconnected.

port1 is tx1_1 and rx1_w

port2 is tx2_1 and rx2_w

since we do only have one 10mhz source in the tester i added a zapd20 splitter in the 10mhz line going to the b200 and using the second port for the lime.

i checked the clock-out of the b200 before, but i seemed completely inactive, thus the splitter. checking the attenuation i think it should be fine (still >3V rect. against 50ohm)

#2 - 08/27/2018 06:30 PM - pespin

I started adding support for it in osmo-gsm-tester prod setup.

LimeSDR unit there had no fw loaded, I had to run "LimeUtil --update" to flash latest one.

I added a pair of commits to osmo-ci to set it up and be able to compile LimeSuite:

<https://gerrit.osmocom.org/#/c/osmo-ci/+10628/>

<https://gerrit.osmocom.org/#/c/osmo-ci/+10629/>

I then have a pair of commits in osmo-gsm-tester branch pespin/modifiers with some refactoring of code in osmo-bts-trx/osmo-trx objects, and then another commit adding support to build and run osmo-trx-lms.

I am building LimeSuite master. I can run osmo-trx-lms in osmo-gsm-tester prod using "-s aoip_ussd:trx-lms", but osmo-trx-lms crashes. Last log line I

see before crash:

```
20180827195745662 DDEV <0001> LMSDevice.cpp:99 [tid=140178283848576] Opening LMS device..
```

Then, with the generated core file:

```
Core was generated by `/home/jenkins/workspace/osmo-gsm-tester_manual-run/trial-149/inst/osmo-trx/bin/'.  
Program terminated with signal SIGILL, Illegal instruction.
```

```
...  
(gdb) bt full  
#0 0x00007f7dcb803147 in lime::ConnectionFX3Entry::handle_libusb_events() () from /tmp/osmo-trx-lms/inst/lib/  
libLimeSuite.so.18.06-1  
No symbol table info available.  
#1 0x00007f7dcb4bce6f in ?? () from /usr/lib/x86_64-linux-gnu/libstdc++.so.6  
No symbol table info available.  
#2 0x00007f7dcc914494 in start_thread (arg=0x7f7dc87cb700) at pthread_create.c:333  
  __res = <optimized out>  
  pd = 0x7f7dc87cb700  
  now = <optimized out>  
  unwind_buf = {cancel_jmp_buf = {{jmp_buf = {140178211256064, -2066435507754056783, 0, 140726539151967,  
0, 140178284142656, 2137269339182646193,  
2137260355090039729}, mask_was_saved = 0}}, priv = {pad = {0x0, 0x0, 0x0, 0x0}, data = {prev =  
0x0, cleanup = 0x0, canceltype = 0}}}  
  not_first_call = <optimized out>  
  pagesize_ml = <optimized out>  
  sp = <optimized out>  
  freesize = <optimized out>  
  __PRETTY_FUNCTION__ = "start_thread"  
#3 0x00007f7dcac31acf in clone () at ../sysdeps/unix/sysv/linux/x86_64/clone.S:97  
No locals.
```

so the crash could be related to be using LimeSuite master. I'll try building an older one and see if I get similar issues.

TODO:

- Build LimeSuite with debug symbols.
- Get osmo-trx-lms working (not crashing) *

#3 - 08/27/2018 08:38 PM - laforge

On Mon, Aug 27, 2018 at 06:30:54PM +0000, pespin [REDMINE] wrote:

I am building LimeSuite master.

the question is **where** you are building it.

illegal instruction might imply that the compiler is emitting instructions which the target platform cannot execute. Maybe there is some broken code in the Lime build system which makes the assumption that the CPU on which you are compiling (packaging) is exactly the same feature set as the CPU on which you'll later run?

In any case, this kind of bug / core dump should be reported to the limesuite tracker on github.

#4 - 10/09/2018 01:38 PM - pespin

- Status changed from *In Progress* to *Feedback*

- % Done changed from 50 to 90

As we discussed a while ago, I moved to use a stable limesuite from debian repos, now that finally we have a new enough version in latest repos (18.10):

<https://gerrit.osmocom.org/#/c/osmo-ci/+11286> ansible: Install liblimesuite-dev and setup limesdr rights in gsm-tester

Submitted osmo-gsm-tester commit:

<https://gerrit.osmocom.org/#/c/osmo-trx/+10723> jenkins.sh: Enable build of osmo-trx-lms

with that commit, we can run tests with osmo-trx-lms and LimeSDR-USB by using scenario "trx-lms".

I configured osmo-trx-lms to use external clock as roh told me there was an external source connected to it.

Core network seems to be running fine, but MS is unable to find the network. Probably related to the bad signal that [roh](#) was seeing in the signal analyzer.

I guess we can close this ticket as it's specific to the setup, and the issue describe is more generic and being tracked in other tickets.

#5 - 11/12/2018 05:02 PM - pespin

- Status changed from *Feedback* to *Resolved*

- % Done changed from 90 to 100

I enabled tests here, though I expect them to fail due to current generic issues with LimeSDR. When at some point there are fixed, we should see

easily that LimeSDR tests are passing.

<https://gerrit.osmocom.org/#/c/osmo-gsm-tester/+11745> default-suites: Enable tests for LimeSDR-USB