

OsmoBTS - Bug #3780

existing test suite doesn't catch wrong measurement processing

02/04/2019 02:30 PM - laforge

Status:	New	Start date:	02/01/2019
Priority:	Normal	Due date:	
Assignee:	dexter	% Done:	0%
Category:			
Target version:			
Spec Reference:			

Description

As we discovered in [#3530](#) and particularly [#2977](#), there appear to be plenty of bugs in the way how measurements are computed.

Despite us having TTCN-3 tests in BTS_Tests.ttcn that also cover measurement processing, those bugs have not been caught so far. This means there are significant (unexpected) gaps in our test coverage which should be fixed. Preferably we should have test coverage that fails before fixing any related issues.

History

#1 - 02/06/2019 07:50 AM - bican304

Stupid spam. Deleted. Please block this account.

#2 - 02/13/2019 02:23 PM - dexter

- File TC_meas_res_sign_sdcch4_meas_expected.txt added
- File TC_meas_res_sign_sdcch4_meas_got.txt added
- File TC_meas_res_sign_sdcch8.meas_expected.txt added
- File TC_meas_res_sign_sdcch8.meas_got.txt added
- File TC_meas_res_sign_sdcch8.pcapng added
- File TC_meas_res_sign_tchh_meas_expected.txt added
- File TC_meas_res_sign_tchh_meas_got.txt added
- File TC_meas_res_sign_tchh_toa256_meas_expected.txt added
- File TC_meas_res_sign_tchh_toa256_meas_got.txt added

I am currently looking at what we already have. To check the measurement reporting we currently have the following tests in place:

BTS_Tests.TC_meas_res_sign_sdcch4
BTS_Tests.TC_meas_res_sign_sdcch8
BTS_Tests.TC_meas_res_sign_tchf
BTS_Tests.TC_meas_res_sign_tchh
BTS_Tests.TC_meas_res_sign_tchh_toa256

Unfortunately BTS_Tests.TC_meas_res_sign_tchf is the only one that is currently passing. I think we need to fix this first. The main reason why the tests are failing is because the measurement indications that comes back has unexpected values for rxq_f_u and rxq_s_u (see attached _expected.txt _got.txt files). The testsuite defines the tolerance range for both values as (0 .. 1), but the actual value that we get when the tests are running is 7. Technically this value may be even in the valid range but the testsuite narrows the valid range. The tests were passing before, probably the testsuite defines a correct test expectation here. We now should check how those values are computed and if this is correct.

When mp_tolerance_rxqual is set to 7, then the tests are more likely to pass, I could make them all pass instead of BTS_Tests.TC_meas_res_sign_sdcch8. However, there still seems to be a problem with other parts of the testsuite or of the iut itself. Sometimes there is no new channel opened and the tests times out because it sees no measurement reports within a certain amount of time. From what I can see, it is not the case that there are just no measurement reports sent, there is not even a channel established. All whats there are CCCH LOAD INDications. For this see attached trace: TC_meas_res_sign_sdcch8.pcapng.

Files

TC_meas_res_sign_sdcch4_meas_expected.txt	2.16 KB	02/13/2019	dexter
TC_meas_res_sign_sdcch4_meas_got.txt	2.26 KB	02/13/2019	dexter
TC_meas_res_sign_sdcch8.meas_expected.txt	2.16 KB	02/13/2019	dexter
TC_meas_res_sign_sdcch8.meas_got.txt	2.25 KB	02/13/2019	dexter
TC_meas_res_sign_sdcch8.pcapng	9.2 KB	02/13/2019	dexter
TC_meas_res_sign_tchh_meas_expected.txt	2.15 KB	02/13/2019	dexter
TC_meas_res_sign_tchh_meas_got.txt	2.25 KB	02/13/2019	dexter
TC_meas_res_sign_tchh_toa256_meas_expected.txt	2.35 KB	02/13/2019	dexter
TC_meas_res_sign_tchh_toa256_meas_got.txt	2.44 KB	02/13/2019	dexter