

OpenBSC - Feature #1824

don't try to deactivate the SACCH in case of deactivation of PDCH channels

10/15/2016 09:32 PM - laforge

Status:	Closed	Start date:	10/15/2016
Priority:	Normal	Due date:	
Assignee:	dexter	% Done:	0%
Category:	libbsc		
Target version:	RBS2000 with BSC-located PCU		
Resolution:		Spec Reference:	
Description			
<p>When attempting to manually enable a PDCH on a RBS2000 without a PCU connected to the CCU, the RBS2000 will inform us using a RSL CONNECTION FAIL message. We respond by RF CHANNEL RELEASE, but we also send a DEACTIVATE SACCH from our channel deactivation logic.</p> <p>As the PDCH has no SACCH (only dedicated circuit-switched channels do), this is of course bogus.</p> <p>So what needs to be done is probably to check if this channel is currently activated as PDCH, and if it is, skip the deactivate SACCH step.</p> <p>I believe this is not only valid for RBS2000 but also for osmocom-style dynamic PDCH support. Neels: Any thoughts on this?</p>			

History

#1 - 10/15/2016 09:34 PM - laforge

- File `rsl_chan_act_pdch.pcap` added

attaching pcap file with example.

don't be surprised if wireshark doesn't fully decode the messages. The non-standard RSL extensions require a patch (see my commit 89c71c635f0e85a66de3a176b0a0df2970d8058b in [wireshark.git](https://github.com/osmocom/wireshark) on osmocom.org, branch `laforge/om2000-201610`). However, a full decode of the RSL CHAN ACT message is not required to solve this bug.

#2 - 10/16/2016 10:38 PM - neels

With my default wireshark I basically see only "malformed packet", so I guess I'd need the extensions.

I'm not familiar with SACCH, but if you say that SACCH only exists for CS channels, then the obvious answer is that we can't deact SACCH on a PDCH channel.

The conclusion would be that `rsl_rx_conn_fail()` calling `rsl_rf_chan_release_err()` in `abis_rsl.c` must not pass `SACCH_DEACTIVATE` unconditionally. This patch could be all that's needed:

```
diff --git a/openbsc/src/libbsc/abis_rsl.c b/openbsc/src/libbsc/abis_rsl.c
index b9132a5..eb33b3a 100644
--- a/openbsc/src/libbsc/abis_rsl.c
+++ b/openbsc/src/libbsc/abis_rsl.c
@@ -859,7 +859,9 @@ static int rsl_rf_chan_release_err(struct gsm_lchan *lchan)
 {
     if (lchan->state != LCHAN_S_ACTIVE)
         return 0;
-    return rsl_rf_chan_release(lchan, 1, SACCH_DEACTIVATE);
+    return rsl_rf_chan_release(lchan, 1,
+                               ts_is_pdch(lchan->ts) ?
+                               SACCH_NONE : SACCH_DEACTIVATE);
 }
```

(haven't tested nor compiled it yet)

BTW...

From 08.58 8.4.4:

CONNECTION FAILURE INDICATION

This message is sent from BTS to BSC to indicate that an active connection has been broken for some reason.

That sounds like a channel was already fully established and then fails.

From a failure to switch on a PDCH lchan, I would expect an 'Channel Activation Nack' instead?

Anyway, if the RBS2000 replies with a conn failure, we have to be able to deal with it...

#3 - 02/28/2017 01:58 PM - dexter

- Status changed from New to Resolved

#4 - 02/28/2017 05:32 PM - laforge

- Status changed from Resolved to Closed

Files

rsl_chan_act_pdch.pcap	318 Bytes	10/15/2016	laforge
------------------------	-----------	------------	---------