**OsmoBTS - Bug #4772**

**PDCH timeslots are always full-power on secondary TRX**

10/01/2020 12:15 PM - laforge

<table>
<thead>
<tr>
<th>Status:</th>
<th>Resolved</th>
<th>Start date:</th>
<th>10/01/2020</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
<td>Due date:</td>
<td></td>
</tr>
<tr>
<td>Assignee:</td>
<td>pespin</td>
<td>% Done:</td>
<td>100%</td>
</tr>
<tr>
<td>Category:</td>
<td>osmo-bts-trx</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spec Reference:</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Description**

The problem is that a dynamic TS (probably also a normal/static PDCH) is transmitting in downlink on a TRX != 0 even if there are no TBF on it (and hence no information transmitted).

The question is "how do we make sure a PDCH only transmits in DL when there is actual, real RLC/MAC blocks being transmitted".

Vadim wrote:

> If I remember correctly, osmo-bts-trx would not Tx anything on PDCH (idle state of TCH/F_TCH/H_PDCH) unless osmo-pcu is connected. As soon as you start osmo-pcu, it starts sending valid RLC/MAC blocks (PACKET_DOWNLINK_DUMMY_CONTROL_BLOCK in Wireshark) to osmo-bts-trx.

In my understanding, we must ensure constant Tx power on a PDCH slot if there is at least one MS/UE is listening to it, because it needs to decode every RLC/MAC block and check the USF assignments, paging, etc. And the problem is that on the BTS side it's impossible to know if there is any TBF activity in the PCU or not, because they're basically separate entities.

understood. So it's osmo-pcu which would need to take that decision: Do we have any TBF allocations on a given timeslot? If yes -> send valid RLC/MAC blocks. If no -> send nothing (or some type of idle request) so osmo-bts doesn't transmit on those timeslots (if on TRX != 0).

**Associated revisions**

Revision 4210e61f - 08/25/2021 02:49 PM - pespin
sched: energy saving: Avoid Tx dummy blocks on empty PDCH TS
Related: SYS#4919
Related: OS#4772
Change-Id: l8d66ddf5e838748611e7b7b504fc86295f02c019

Revision 8e86671f - 08/25/2021 03:52 PM - pespin
pcu: Introduce test TC_pdch_energy_saving
Related: SYS#4919
Related: OS#4772
Change-Id: lc5ccf4e67ecabe3cbfb86e0f4caf497669b76fb42

Revision a89a2388 - 08/26/2021 10:51 AM - pespin
sched: energy saving: Avoid Tx dummy blocks on empty PDCH TS
Related: SYS#4919
Related: OS#4772
Change-Id: l8d66ddf5e838748611e7b7b504fc86295f02c019

Revision 3aa61c66 - 08/31/2021 10:10 AM - pespin
pcu: Introduce test TC_pdch_energy_saving
Related: SYS#4919
Related: OS#4772
Change-Id: lc5ccf4e67ecabe3cbfb86e0f4caf497669b76fb42

**History**

09/18/2022
Vadim Yanitskiy wrote:

Hi Harald,

understood. So it's osmo-pcu which would need to take that decision: Do we have any TBF allocations on a given timeslot? If yes - send valid RLC/MAC blocks. If no - send noting (or some type of idle request) so osmo-bts doesn't transmit on those timeslots (if on TRX != 0).

yes, this approach sounds good. On the [PCUIF] protocol level, I see two potential solutions:

- Add a new NOPE.req message that would be sent in response to RTS.req in case if there is no TBF activity on a given PDCH slot. This would require us to either bump the PCUIF version [again], or add a new flag to INFO.ind indicating NOPE.req support to the PCU.

- Abuse one of the fields in DATA.req, which is not used by the current osmo-bts-trx version. We do have DATA.ind specific fields (Uplink measurements) in DATA.req message, so one of them could indicate a frame that is not necessary to transmit.

I would go for adding the flag to INFO.ind. So the BTS indicates this flag to the PCU, and the PCU can then start sending those new requests in response to RTS.req (on TRX != 0). This is fully backwards compatible to both older BTS with newer PCU and vice-versa.

pespin

AFAICS, all your patches have been merged. Closing this ticket.