OsmoBTS - Bug #4008

Channel Activation starts SACCH too early in Asynchronous Handover

05/19/2019 08:48 PM - laforge

Status:	In Progress	**Start date:**	05/19/2019
Priority:	High	**Due date:**	
Assignee:		**% Done:**	70%
Category:			
Target version:			
Spec Reference:	3GPP TS 48.058 Section 4.1.3		

Description

In case of an asynchronous hand-over related channel activation, 3GPP TS 48.058 Section 4.1.3 is very specific:

```
BTS starts transmission immediately on the main channel in the indicated mode and with encryption
if so indicated. If
the MS Power element is present the BTS may start transmission also on the SACCH.

When receiving a correct access burst with the correct handover reference, BTS starts the normal r
eception process on
the main channel in the indicated mode and starts receiving (and sending if not started earlier) o
n SACCH. Deciphering
is started if so indicated. The handover detection procedure towards BSC is also started.
```

However, as uncovered by an upcoming BTS_Tests.TC_sacch_chan_act_ho_async test case, we appear to be activating the SACCH unconditionally from the first moment.

The problem here is quite obvious: Until we have received the access burst from the MS, we don't yet know the timing offset, and hence the timing advance that we should advertise in the downlink SACCH. If we start SACCH transmission too early, it means that a wrong TA is advertised, which may be picked up by the MS, which will then apply a wrong TA value -> boom.

Related issues:

Related to OsmoBTS - Feature #3750: Extension of BTS_Tests.ttcn test coverage	**Resolved**	**01/08/2019**
Related to OsmoBSC - Feature #4010: Test RSL CHAN ACT related details for dif...	**New**	**05/19/2019**

Associated revisions

Revision 652ab250 - 01/28/2021 11:10 AM - Neels Hofmeyr

lchan activation: indicate whether TA is known

On lchan activation, we already know the Timing Advance in most situations: from the Channel Request RACH, or from a previous lchan in the same cell. Place this information in lchan->activate.info.ta.

So far, the lchan->last_ta (until recently called rqd_ta) was used to store the initial TA for channel activation -- move the initial TA to lchan->activate.info.ta, for proper scoping.

Only an inter-cell handover does not yet know a Timing Advance (until the Handover Detection RACH is received), so indicate activate.info.ta_known = false for that case.

If ta_known is false, do not include an Access Delay IE in the Channel Activation message, ensuring that the BTS does not use an arbitrary TA that is likely inaccurate.

The effect for OsmoBTS is that we will **not** start the downlink SACCH on channel activation for inter-cell handover, but will wait for a HO RACH first, and then use the correct TA when enabling downlink SACCH.

Related: OS#4008 SYS#5192
Change-Id: I986bf93e8acd6aef7eaf63ac962480b680aa894f

Revision 10272f77 - 01/28/2021 10:57 PM - Neels Hofmeyr

chan activ: activate DL SACCH only when TA is known

A channel activation for handover to another cell does not know the Timing Advance until the handover RACH is received. It does not make much sense to enable downlink SACCH without an accurate TA.

If the BSC omits the Access Delay IE (a.k.a. the Timing Advance), do not enable downlink SACCH. This is expected to happen only for inter-cell handover. In all other situations, the TA should be known either from a Channel Request RACH for Immediate Assignment, or from the previous lchan on the same cell upon Assignment / intra-cell handover.

Related: OS#4008 OS#4009 SYS#5192
Change-Id: I170b63c9856230d5f1a10654a9d950ada8e730d7

Revision b03e73f2 - 02/05/2021 07:24 PM - Neels Hofmeyr

lchan activation: indicate whether TA is known

On lchan activation, we already know the Timing Advance in most situations: from the Channel Request RACH, or from a previous lchan in the same cell. Place this information in lchan->activate.info.ta.

So far, the lchan->last_ta (until recently called rqd_ta) was used to store the initial TA for channel activation -- move the initial TA to lchan->activate.info.ta, for proper scoping.

Only an inter-cell handover does not yet know a Timing Advance (until the Handover Detection RACH is received), so indicate activate.info.ta_known = false for that case.

If ta_known is false, do not include an Access Delay IE in the Channel Activation message, ensuring that the BTS does not use an arbitrary TA that is likely inaccurate.

The effect for OsmoBTS is that we will **not** start the downlink SACCH on channel activation for inter-cell handover, but will wait for a HO RACH first, and then use the correct TA when enabling downlink SACCH.

Related: OS#4008 SYS#5192
Change-Id: I986bf93e8acd6aef7eaf63ac962480b680aa894f

History

#1 - 05/19/2019 08:50 PM - laforge

- Related to Feature #3750: Extension of BTS_Tests.ttcn test coverage added

#2 - 05/19/2019 08:58 PM - laforge

- Related to Feature #4010: Test RSL CHAN ACT related details for different scenarios added

#3 - 05/22/2019 07:18 AM - laforge

- Checklist item [ ] osmo-bts-sysmo added
Checklist item [ ] osmo-bts-oc2g added
Checklist item [ ] osmo-bts-lc15 added
Checklist item [ ] osmo-bts-trx added

There's good news and bad news.

The actual behavior of the BTS is very dependent on the specific PHY used. I've analyzed osmo-bts-trx and osmo-bts-sysmo as two representatives, where lc15 and oc2g are mostly like -sysmo.

osmo-bts-sysmo

osmo-bts-sysmo gets it half-way right. It

- checks if the activation is HO related, and only activates uplink RACH detection until a RACH is received
- then activates all other logical channels / SAPIs after the RACH was received

What it gets wrong:

- it doesn't activate DL main channel (FACCH/SDCCH) while waiting for the RACH
- it unconditionally delays activation of DL+UL SACCH, even if the MS Power IE and/or TA IE were present in RSL CHAN ACT

osmo-bts-trx

osmo-bts-trx gets it wrong in all cases:

- it always activates both main channel and SACCH in UL and DL from the very beginning, even before any RACH is received in UL

In fact, osmo-bts-trx and its scheduler don't even know the concept of L1 SAPI and hence don't have the infrastructure to enable/disable individual logical channels within one dedicated channel.

Summary

What does this all mean in practise for osmo-bts-trx? There is a significant risk of poor hand-over performance, as

- some phones could receive a massively wrong timing advance before we even know the TA
- some phones could simply refuse to send the RACH if there is no downlink FACCH/SDCCH visible

#4 - 06/19/2019 08:34 AM - laforge

- Priority changed from Urgent to High

#5 - 07/18/2019 05:10 AM - laforge

- Assignee deleted (laforge)

#6 - 07/18/2019 05:15 AM - laforge

- Priority changed from High to Normal

#7 - 11/03/2020 04:17 PM - neels

- Priority changed from Normal to High

#8 - 01/27/2021 03:19 PM - neels

- Checklist item [x] osmo-bts-sysmo set to Done

#9 - 01/27/2021 03:19 PM - neels

- Checklist item [x] osmo-bts-sysmo set to Done
Checklist item [x] osmo-bts-oc2g set to Done
Checklist item [x] osmo-bts-lc15 set to Done
Checklist item [x] osmo-bts-trx set to Done

#10 - 01/27/2021 03:38 PM - neels

- Checklist item [ ] osmo-bts-sysmo set to Not done
Checklist item [ ] osmo-bts-oc2g set to Not done
Checklist item [ ] osmo-bts-lc15 set to Not done
Checklist item [ ] osmo-bts-trx set to Not done

- Status changed from New to In Progress

- % Done changed from 0 to 70

Patches have been merged to keep SACCH deactivated in certain situations
(depending on what IEs the BSC sent during Channel Activation).

Still, OsmoBSC always sends the MS Power IE, so those patches will never have an effect when using OsmoBSC.
The rather non-trivial question is: what IEs should the BSC send in which situations?

The aim of this ticket is to keep SACCH deactivated as long as the TA is not known.
3GPP TS 48.058 4.1.3 and 4.1.4 define which SAPIs should be enabled for a handover target lchan.
Summary of that spec:

```
          |          | Access ||  transmit         |  activate
          | MS Power | Delay  ||  on main channel  |  dl SACCH
 ------------------------------------------------------------------
 async ho   no         *     -->  yes                no
 async ho   yes        *     -->  yes                may be started
 sync ho    no         no    -->  yes                no
 sync ho    yes        no    -->  yes                may be started
 sync ho    yes        yes   -->  yes                shall be started
```

So for this ticket we are mostly interested in the MS Power IE being present or not.

Looking at 3GPP TS 48.058 we get this information from the footnotes:

- MS Power IE has two different meanings:
  - when MS Power Parameters IE is **not** present: MS Power IE indicates the initial power the MS should use.
  - when MS Power Parameters IE **is** present: MS Power IE indicates the maximum permitted MS power value.

OsmoBSC transmits the MS Power Parameters IE only when in dynamic power control mode.
See https://git.osmocom.org/osmo-bsc/tree/src/osmo-bsc/abis_rsl.c?id=80184ae1714f52cbeed8d94472975f07122e806f#n205

For a new lchan assignment or handover to another cell, OsmoBSC sends the bts->ms_max_power in the MS Power IE.
For a handover within the same cell, OsmoBSC sends the previous lchan's MS power in the MS Power IE.
(see https://git.osmocom.org/osmo-bsc/tree/src/osmo-bsc/lchan_fsm.c?id=80184ae1714f52cbeed8d94472975f07122e806f#n577)

So, first of all, it seems that we have a bug in OsmoBSC:
If a channel activation for handover sends both MS Power IE as well as MS Power Parameters IE (in dynamic power control mode),
then the BTS should interpret the MS Power IE as the maximum permitted MS power. But we send the previous lchan's power.

The other question is whether we can or ever should omit the MS Power IE from a channel activation,
to get the effect that we wanted to achieve with this issue in the first place,
being to omit the DL SACCH from a handover channel activation until the TA is known.
I would have assumed that the lchan should omit DL SACCH when the Timing Advance is omitted from the Channel Activation,
but at least for async HO, the TA presence (the Access Delay IE) apparently does not make any difference in the decision
to activate DL SACCH or not on initial channel activation, according to the spec.

At this moment I am a bit confused / undecided on what IEs the BSC should send in what situations...
any help is welcome.

#11 - 01/27/2021 03:41 PM - neels

- Checklist item [x] osmo-bts-sysmo set to Done
Checklist item [x] osmo-bts-oc2g set to Done
Checklist item [x] osmo-bts-lc15 set to Done
Checklist item [x] osmo-bts-trx set to Done

the merged patches are
https://git.osmocom.org/osmo-bts/commit/?id=f733cf826375a3f348b6195b286af872beef0ff7
https://git.osmocom.org/osmo-bts/commit/?id=1c05ef15ecb1a9053112b2c11bb5c062cee06478
https://git.osmocom.org/osmo-bts/commit/?id=61585254338d16bb359987ad1f4537900e6288db
https://git.osmocom.org/osmo-bts/commit/?id=4ee4d6be2a66a151de0fc0ef258d89fad4135a91
https://git.osmocom.org/osmo-bts/commit/?id=def24f0d9af2463a5ef557d35f23abd5b4d07120

#12 - 01/27/2021 03:51 PM - neels

```
async ho   yes        *     -->  yes                 may be started
```

I just realize that OsmoBTS could **always** deactivate DL SACCH for async HO,
regardless of the MS Power IE presence, because the spec says the DL SACCH **may** be started, not **shall** be started.

For sync HO it's a bit different, but this ticket here is about async HO.

#13 - 01/27/2021 04:07 PM - neels

```
          |           | Access ||  transmit          |  activate
          | MS Power  | Delay  ||  on main channel   |  dl SACCH
 ------------------------------------------------------------------------
 async ho   no         *     -->  yes                 no
 async ho   yes        *     -->  yes                 may be started
 sync ho    no         no    -->  yes                 no
 sync ho    yes        no    -->  yes                 may be started
 sync ho    yes        yes   -->  yes                 shall be started
```

A possible overall solution for both async and sync handover:

- when both MS Power and Access Delay IEs are present, then activate SACCH.
- otherwise, do not activate SACCH.

This would adhere to all the "no" rows, would opt for not sending SACCH for all "may be started" rows,
and also adheres to the last "shall be started" row.

Hence OsmoBSC would in practice trigger DL SACCH activation or not by sending Access Delay or not.
Then we can implement in OsmoBSC that we mostly omit an Access Delay IE in HO channel activation.
Send Access Delay only when it is an intra-cell HO with an already known TA.

#14 - 01/27/2021 07:40 PM - laforge

Neels, I currently cannot dive into the last 'depty' of the topic, but one thing
to always keep in mind: OsmoBSC supports a vareity of BTS products out there, OsmoBTS
is only one of them.

So no matter what kind of changes we introduce, we should result in a spec-compliant
behavior. Relying on some special logic in OsmoBTS can only be considered as a last
resort, if the specs don't permit any generic solution.