OsmoPCU - Feature #1526

Acquire/update timing advance (TA)

02/22/2016 08:59 PM - zecke

Status:	Stalled	**Start date:**	02/22/2016
Priority:	Normal	**Due date:**	
Assignee:	lynxis	**% Done:**	30%
Category:			
Target version:			
Spec Reference:			

Description

Currently the TA is derived from the initial RACH request and never updated during the lifetime of a TBF.

Is this handled correctly for network initiated DL TBF establishment?

TS 44.018, 3.5.3.1.2 asks for TA determination on DL TBF establshment if the MS is idle and the TA is not known (see "If the network does not have a valid timing advance ...").

Other related issues:

- The burst timing info (qta) should be applied (see #1705)
- Support access bursts on PDCH which can be requested by the PCU if the current TA is unknown (44.060)

Related issues:

Related to OsmoPCU - Feature #1545: continuous timing advance loop using PTCCH	**Stalled**	**02/23/2016**
Related to OsmoPCU - Feature #1531: Use the burst timing information to compu...	**Closed**	**02/22/2016**
Related to OsmoPCU - Bug #1524: PACCH on the wrong timeslot	**Stalled**	**02/22/2016**
Related to OsmoPCU - Bug #3472: GPRS connection is in a state where pdp-conte...	**New**	**08/18/2018**
Related to OsmoBTS - Feature #2977: OsmoBTS measurment processing at L1SAP to...	**Stalled**	**02/21/2018**

History

#2 - 07/12/2016 11:04 AM - laforge

- Priority changed from Low to High

#3 - 07/12/2016 11:27 AM - msuraev

Reference to #1705 is probably wrong.

#4 - 07/12/2016 11:28 AM - msuraev

- Related to Feature #1545: continuous timing advance loop using PTCCH added

#5 - 07/13/2016 08:46 AM - msuraev

- Related to Feature #1531: Use the burst timing information to compute the timing advance added

#6 - 07/14/2016 12:49 AM - laforge

msuraev wrote:

> Reference to #1705 is probably wrong.

It's probably #1531

#7 - 07/17/2016 02:22 PM - msuraev

- Status changed from New to In Progress

- % Done changed from 0 to 10

TS 44.060 is referred to by 44.018 in "If the network does not have a valid timing advance ..." passage: "the network shall use the procedures defined in 3GPP TS 44.060". For both 1-phase (7.1.2.5) and 2-phase (7.1.3.5) in 44.060 the only procedure defined in continuous TA update described in #1545.

Alternatively 44.018 suggests to use "polling mechanism" to derive TA "if the PACKET CONTROL ACKNOWLEDGEMENT format is set to four access bursts".

The PACKET CONTROL ACKNOWLEDGEMENT format is described in TS 44.060 § 11.2.2 including 11-bit and 8-bit RACH.

The ACK format is specified in SI parameter CONTROL_ACK_TYPE and in Packet Polling Request message in § 11.2.12.

#8 - 07/17/2016 02:29 PM - msuraev

Currently TA is initialized with 0 by default which means we always begin with valid TA.

#9 - 07/18/2016 09:08 AM - msuraev

The SI message referred to above is Packet System Information Type 1 (§11.2.18) which has CONTROL_ACK_TYPE bit in GPRS Cell Options (§12.24) both in 3GPP TS 44.060.

#10 - 07/18/2016 09:22 AM - msuraev

Alternatively (our case as we do not support yet PBCCH) SI type 13 from 3GPP TS 44.018 §9.1.43a can be used as §10.5.2.37b refers to same IE specified for PSI1.

#11 - 07/19/2016 02:31 PM - msuraev

- Status changed from In Progress to Stalled

With gerrit #543, 544, 547 TA handling seems to be fine. So far I have not managed to reproduce situation in which PCU do not know TA - it's always derived from initial RACH qTA. The question is - is it worth adding support for access bursts on PDCH if there's no need for PCU to use polling which will trigger them? Note: it might also require changes to DSP firmware.

#12 - 07/20/2016 01:17 PM - msuraev

Apparently TA might be unknown in 2 cases:
- packet queuing notification is used
- packet UL/DL assignment sent without prior paging

#13 - 07/24/2016 08:53 PM - laforge

msuraev wrote:

> The question is - is it worth adding support for access bursts on PDCH if there's no need for PCU to use polling which will trigger them? Note: it might also require changes to DSP firmware.

Is there something specified in GPRS that permits access bursts on a PDTCH outside of the TCCH/PACCH?

Also, in terms of the DSP firmware, I would assume you just need to activate the "right" SAPI on the logical channel, and it should do the correlation/decoding of access bursts.

#14 - 07/27/2016 11:27 AM - msuraev

laforge wrote:

> Is there something specified in GPRS that permits access bursts on a PDTCH outside of the TCCH/PACCH?

It's possible to configure the use of access bursts (x4) instead of ctrl ack packets - see gerrit #546.

> Also, in terms of the DSP firmware, I would assume you just need to activate the "right" SAPI on the logical channel, and it should do the correlation/decoding of access bursts.

You mean pysical-layer SAPI or link-layer SAPI?

#15 - 07/30/2016 11:08 AM - laforge

On Wed, Jul 27, 2016 at 11:27:32AM +0000, msuraev [REDMINE] wrote:

> > Also, in terms of the DSP firmware, I would assume you just need to
> > activate the "right" SAPI on the logical channel, and it should do
> > the correlation/decoding of access bursts.
>
> You mean pysical-layer SAPI or link-layer SAPI?

physical-layer SAPI (such as RACH).

--
- Harald Welte <laforge@gnumonks.org> http://laforge.gnumonks.org/
==========
"Privacy in residential applications is a desirable marketing option."
(ETSI EN 300 175-7 Ch. A6)

#16 - 08/01/2016 07:41 AM - laforge

why is this stalled at 10%? Please update ticket status

#17 - 08/01/2016 09:08 AM - msuraev

- Status changed from Stalled to In Progress

Was working on other tickets in a meantime.

#18 - 08/09/2016 05:31 PM - msuraev

Fixes for TA handling submitted for review in gerrit #547, 552. With the code from gerrit #546 we can test decoding of 4xRACH in place of CTRL_ACK (once the decoding is available). See also OS#1545.

#19 - 10/12/2016 04:39 PM - msuraev

- Status changed from In Progress to Stalled

Gerrit #552 is still under review.

#20 - 11/09/2016 09:31 AM - laforge

Gerrit 552 is now merged.

#21 - 01/06/2017 01:27 PM - msuraev

Turning on "gprs control-ack-type-rach" option in OpenBSC breaks GPRS (expectedly), 4xRACH response is not visible in the logs (unexpectedly) although GsmL1_Sapi_Prach is activated for RxUplink - see pdtch_sapis in oml.c Investigation is ongoing.

#22 - 01/12/2017 08:19 PM - laforge

- Priority changed from High to Normal

#23 - 05/31/2017 12:24 PM - msuraev

- Related to Bug #1524: PACCH on the wrong timeslot added

#24 - 06/01/2017 08:37 AM - msuraev

The Packet Control Acknowledgement in RACH format described in 3GPP TS 44.060 Table 11.2.2.1, when trying to detect corresponding message type in PCU's pcu_rx_rach_ind() I've got nothing so far. I suspect that the reason is low-level encoding difference: I've got several consequitive RACH decoding errors from BTS' rx_rach_fn() (the tests were done using osmo-bts-trx as it exposes more information than DSP). It means that osmo_crc8gen_check_bits() fails in gsm0503_rach_decode() in libosmocoding - I'm not sure yet if different polynomial should be used compared to regular RACH. Note: at the time of tests 11-bit RACH is not supported by osmo-bts-trx.

#25 - 10/02/2017 12:49 PM - msuraev

Related fix is available in gerrit 3991.

#26 - 10/23/2017 12:09 PM - msuraev

- Status changed from Stalled to In Progress

Related gerrit 4336-4339, 4292 were sent for review.

#27 - 11/02/2017 09:16 AM - msuraev

- Status changed from In Progress to Stalled

- % Done changed from 10 to 20

Gerrit 4292 should be rewritten, the rest is merged.

#28 - 01/11/2018 09:35 AM - msuraev

- Status changed from Stalled to In Progress

- % Done changed from 20 to 30

The fixes for IA rest octets which seem to prevent 4xRACH reply from MS were posted for review in gerrit 5726 - 5729.

#29 - 01/17/2018 11:33 AM - msuraev

Fixes are merged, needs additional testing with "gprs control-ack-type-rach" and different BTS models.

#30 - 01/22/2018 10:33 AM - msuraev

- Status changed from In Progress to Stalled

#31 - 03/01/2018 11:13 PM - laforge

- Assignee changed from msuraev to sysmocom

#32 - 08/21/2018 02:54 PM - keith

I have found that when we create a new dl_tbf, we are not setting the TA to a valid TA at any time, this is resulting in:

```
DTBFDL <0009> tbf_dl.cpp:506 TBF(TFI=0 TLLI=0xf4b4d5ee DIR=DL STATE=NULL) Send dowlink assignment on PCH, no T
BF exist (IMSI=262423203000351)
DTBF <0008> tbf_dl.cpp:510 TBF(TFI=0 TLLI=0xf4b4d5ee DIR=DL STATE=NULL) changes state from NULL to ASSIGN
DTBF <0008> bts.cpp:799 TBF(TFI=0 TLLI=0xf4b4d5ee DIR=DL STATE=ASSIGN) TX: START Immediate Assignment Downlink
 (PCH)
DRLCMAC <0002> bts.cpp:807  - TRX=0 (69) TS=6 TA=220 pollFN=-1
```

Note: **TA=220**

And we are telling the MS to use an invalid TA in the resulting IMM.ASS?

Maybe MS some ignore this and continue to work, but others set their max valid TA?

This would explain behavior I have observed a lot; that the MS is transmitting in response, but the BTS/PCU is not seeing the response, because the TA is wrong.

#33 - 08/22/2018 10:46 AM - keith

- Related to Bug #3472: GPRS connection is in a state where pdp-context is active, but data TX cannot initiate from Network side. added

#34 - 10/02/2018 03:47 PM - laforge

#35 - 10/17/2018 09:48 AM - laforge

- Assignee changed from sysmocom to msuraev

#36 - 01/28/2019 09:26 AM - msuraev

- Related to Feature #2977: OsmoBTS measurment processing at L1SAP too complex / pass measurements along with data added

#37 - 04/15/2019 07:50 AM - laforge

- Assignee changed from msuraev to lynxis