

OsmoBTS - Bug #3032

(Wrong?) measurement of both RSSI and ToA on TCH channels

03/05/2018 08:01 AM - fixeria

Status:	New	Start date:	03/05/2018
Priority:	High	Due date:	
Assignee:	tnt	% Done:	0%
Category:	osmo-bts-trx		
Target version:			
Spec Reference:			

Description

Unlike xCCH and PDTCH, where we measure both RSSI and ToA values **for each burst**:

```
// ...

/* update mask + RSSI */
*mask |= (1 << bid);
*rssi_sum += rssi;
(*rssi_num)++;
*toa256_sum += toa256;
(*toa_num)++;

// ...

/* Send uplink measurement information to L2 */
llif_process_meas_res(llt->trx, tn, *first_fn,
    trx_chan_desc[chan].chan_nr | tn, n_errors, n_bits_total,
    *rssi_sum / *rssi_num, *toa256_sum / *toa_num);

ber10k = compute_ber10k(n_bits_total, n_errors);
return _sched_compose_ph_data_ind(llt, tn, *first_fn, chan,
    l2, l2_len,
    *rssi_sum / *rssi_num,
    4 * (*toa256_sum) / *toa_num, 0, ber10k,
    PRES_INFO_UNKNOWN);

// ...
```

both TCH/F and TCH/H are currently sending the measurements **for the last burst**:

```
// ...

/* Send uplink measurement information to L2 */
llif_process_meas_res(llt->trx, tn, *first_fn,
    trx_chan_desc[chan].chan_nr|tn, n_errors, n_bits_total,
    rssi, toa256);

// ...

/* FACCH */
_sched_compose_ph_data_ind(llt, tn,
    (fn + GSM_HYPERFRAME - 7) % GSM_HYPERFRAME, chan,
    tch_data + amr, GSM_MACBLOCK_LEN,
    rssi, 4 * toa256,
    0, ber10k, PRES_INFO_UNKNOWN);

// ...
```

```
return _sched_compose_tch_ind(11t, tn,
    (fn + GSM_HYPERFRAME - 7) % GSM_HYPERFRAME,
    chan, tch_data, rc);
```

Moreover, in case of FACCH carried on a TCH/H channel the ToA256 is for some reason divided by 64, while in case of TCH/F it's passed as is:

```
/* TCH/F FACCH */
_sched_compose_ph_data_ind(11t, tn,
    (fn + GSM_HYPERFRAME - 7) % GSM_HYPERFRAME, chan,
    tch_data + amr, GSM_MACBLOCK_LEN,
    rssi, 4 * toa256,
    0, ber10k, PRES_INFO_UNKNOWN);

/* TCH/H FACCH */
_sched_compose_ph_data_ind(11t, tn,
    (fn + GSM_HYPERFRAME - 10 - ((fn % 26) >= 19)) % GSM_HYPERFRAME,
    chan, tch_data + amr, GSM_MACBLOCK_LEN,
    rssi, toa256/64,
    0, ber10k, PRES_INFO_UNKNOWN);
```

And finally, on TCH channels we only attach the measurements to FACCH indications, while speech frames don't have such info:

```
/* TCH/F FACCH */
_sched_compose_ph_data_ind(11t, tn,
    (fn + GSM_HYPERFRAME - 7) % GSM_HYPERFRAME, chan,
    tch_data + amr, GSM_MACBLOCK_LEN,
    rssi, 4 * toa256,
    0, ber10k, PRES_INFO_UNKNOWN);

// ...

/* TCH/F speech frame */
return _sched_compose_tch_ind(11t, tn,
    (fn + GSM_HYPERFRAME - 7) % GSM_HYPERFRAME,
    chan, tch_data, rc);
```

I wish I could resolve this issue myself, but I didn't find anything in GSM specifications about the BTS side measurement process...

According to the GSM TS 05.03, a single TCH/F frame is mapped over 8 bursts, while a TCH/H frame is mapped over 6 bursts. This should be taken into account.

Any ideas are welcome.

Related issues:

Related to OsmoBTS - Feature #2977: OsmoBTS measurment processing at L1SAP to...	Stalled	02/21/2018
Related to OsmoBTS - Bug #2329: frequent "no space for uplink measurement" me...	Resolved	06/18/2017

History

#1 - 03/05/2018 08:39 AM - laforge

- Assignee set to fixeria

Hi fixeria,

the measurement processing is specified in TS 48.058 4.5 ("The averaging period is one SACCH block period (same as the basic period for MS).")

We implement "basic measurement reporting" only, not the not-fully-specified pre-processed measurement reporting.

This means that we are averaging all ToA and RSSI information over the period of one SACCH multiframe, which is 480ms. For that we use all the measurement information passed up on L1SAP for that lchan.

So the question is if the input values to this (reported at each block on l1sap) should already be the averages over the four bursts comprising the block? I think yes. After all, we don't want the average of short samples, but the average over all bursts in the SACCH multiframe.

#2 - 03/05/2018 08:42 AM - laforge

As a side-note, I think the entire handling of measurements across L1SAP is currently too complicated, I will add it as a 'related issue' here. Ideally, we should attach all measurement information in the best resolution (ber10k, toa256, rssi) to the PH-DATA.ind and remove the somewhat strange INFO-IND that we use separately at the moment. After all, we have the following cases:

- measurements only, no valid data: Use PH-DATA.ind with zero-length payload (bad frame)
- measurements + data: Use fully filled-in PH-DATA.ind
- data without measurements: doesn't exist, doesn't make sense to support (but theoretically possible now).

#3 - 03/05/2018 08:43 AM - laforge

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

#4 - 03/05/2018 06:25 PM - fixeria

- Related to Bug #2329: *frequent "no space for uplink measurement" message added*

#5 - 10/20/2018 07:36 PM - laforge

- Assignee changed from fixeria to tnt

#6 - 11/26/2018 02:32 PM - laforge

- Priority changed from Normal to High