

OsmoBSC - Bug #3679

bsc_vty.c iterates trx to regen sysinfo manually, skips bcch_change_mark bump

10/31/2018 01:00 AM - neels

Status:	Resolved	Start date:	10/31/2018
Priority:	Normal	Due date:	
Assignee:	laforge	% Done:	100%
Category:			
Target version:			
Spec Reference:			
Description			
See <code>gsm_bts_set_system_infos()</code> in <code>bsc_init.c</code> , which increments <code>bcch_change_mark</code> . Compare <code>bts_resend_cmd</code> in <code>bsc_vty.c</code> which does			
<pre>l1list_for_each_entry_reverse(trx, &bts->trx_list, list) gsm_bts_trx_set_system_infos(trx);</pre>			
and doesn't increment <code>bcch_change_mark</code> .			
Does this need fixing?			

History

#1 - 06/21/2020 01:54 PM - laforge

- Status changed from *Feedback* to *In Progress*

- % Done changed from 0 to 70

Patch in <https://gerrit.osmocom.org/c/osmo-bsc/+18938>

#2 - 06/23/2020 10:02 PM - neels

related, I'm not sure about this at all, but I now also notice a possible similar problem in `bootstrap_rsl()` in `osmo_bsc_main.c`:

```
static void bootstrap_rsl(struct gsm_bts_trx *trx)
{
[...]
```

```
    gsm_bts_trx_set_system_infos(trx);
```

```
static int inp_sig_cb(unsigned int subsys, unsigned int signal,
void *handler_data, void *signal_data)
{
[...]
```

```
    switch (signal) {
    case S_L_INP_TEI_UP:
[...]
```

```
        if (isd->link_type == E1INP_SIGN_RSL)
            bootstrap_rsl(trx);
```

So whenever a BTS reconnects and brings a TRX up, there is no `bcch_change_mark` increment, because this only calls `gsm_bts_TRX_set_system_infos()` and not `gsm_BTS_set_system_infos()`. Is that fine?

#3 - 06/24/2020 05:26 AM - laforge

- Status changed from *In Progress* to *Resolved*

- % Done changed from 70 to 100

patch merged