Enabling `GsmL1_Sapi_Ptch` on direction `GsmL1_Dir_RxUplink` fails for the black-box DSP based models:

```
<0006> oml.c:511 activating (bts=0,trx=0,ts=7,ss=0) pchan=PDCH ts_connect_as(PDCH) logChComb=pdtch
<0006> oml.c:808 Successful activation of L1 SAPI PDTCH on TS 7  <-- Downlink
<0006> oml.c:808 Successful activation of L1 SAPI PTCCH on TS 7  <-- Downlink
<0006> oml.c:808 Successful activation of L1 SAPI PRACH on TS 7  <-- Uplink
<0006> oml.c:808 Successful activation of L1 SAPI PDTCH on TS 7  <-- Uplink
<0006> oml.c:813 Error activating L1 SAPI PTCCH on TS 7: Invalid parameter  <-- Uplink
```

This is needed in order to support Continuous Timing Advance procedures on PDCH (see #1545).

At the moment we can only send PTCCH blocks on Downlink, while the Access Burst (PTCCH/U) detection task is not enabled:

```
static const struct sapi_dir pdtch_sapis[] = {
    { GsmL1_Sapi_Pdtch,     GsmL1_Dir_TxDownlink  },
    { GsmL1_Sapi_Pdtch,     GsmL1_Dir_RXUplink   },
    { GsmL1_Sapi_Ptch,      GsmL1_Dir_TxDownlink  },
    { GsmL1_Sapi_Prach,     GsmL1_Dir_RXUplink   },
    #if 0
    { GsmL1_Sapi_Ptch,      GsmL1_Dir_RXUplink   },  // <-- PTCCH/U
    { GsmL1_Sapi_Pacch,     GsmL1_Dir_TxDownlink  },
    #endif
};
```

This SAPI has been disabled on purpose, because the error indication breaks dynamic timeslots in the BSC (see #1796).

My best guess is that the DSP does not allow to enable both `GsmL1_Sapi_Prach` (Uplink) and `GsmL1_Sapi_Ptch` (Uplink) at the same time. As was pointed out by Max, according to the DSP's docs the only channel combination with PTCCH support is `GsmL1_LogChComb_XIII`. As per 3GPP TS 45.002, this combination XIII includes the following channels:

- PDTCH/F
- PACCH/F
- PTCCH/F

and PRACH is not a part of it! Perhaps we should enable `GsmL1_Sapi_Pacch` on `GsmL1_Dir_RXUplink` instead of `GsmL1_Sapi_Prach` in order to support Packet Control Ack in form of four Access Bursts, and try enabling `GsmL1_Sapi_Ptch` on `GsmL1_Dir_RXUplink`:

```
static const struct sapi_dir pdtch_sapis[] = {
    { GsmL1_Sapi_Pdtch,     GsmL1_Dir_TxDownlink  },
    { GsmL1_Sapi_Pdtch,     GsmL1_Dir_RXUplink   },
    /* PTCCH/D and PTCCH/U for Continuous Timing Advance loop */
    { GsmL1_Sapi_Ptch,      GsmL1_Dir_TxDownlink  },
    { GsmL1_Sapi_Ptch,      GsmL1_Dir_RXUplink   },
    /* Packet Control Ack in form of four Access Bursts */
    { GsmL1_Sapi_Pacch,     GsmL1_Dir_RXUplink   },
};
```

I don't have a possibility to verify my (potentially wrong) assumption, so waiting for remote access to be provided by @laforge.

**Related issues:**

- #1545
- #1796
Unfortunately my assumption turned out to be wrong, the DSP still returns an error:

```<0006> oml.c:813 Error activating L1 SAPI PTCCH on TS 7: Invalid parameter```

but the error message itself ('Invalid parameter') probably means that MPH-ACTIVATE.req we're sending is incomplete or incorrect:

```static int mph_send_activate_req(struct gsm_lchan *lchan, struct sapi_cmd *cmd){
    struct femtol1_hdl *fl1h = trx_femtol1_hdl(lchan->ts->trx);
    struct msgb *msg = l1p_msgb_alloc();
    int sapi = cmd->sapi;
    int dir = cmd->dir;
    GsmL1_MphActivateReq_t *act_req;
    GsmL1_LogChParam_t *lch_par;

    act_req = prim_init(msgb_l1prim(msg), GsmL1_PrimId_MphActivateReq, fl1h, l1p_handle_for_lchan(lchan));
    lch_par = &act_req->logChPrm;
    act_req->u8Tn = lchan->ts->nr;
    act_req->subCh = lchan_to_GsmL1_SubCh_t(lchan); // <-- NOTE
    act_req->dir = dir;
    act_req->sapi = sapi;
    act_req->hLayer2 = l1if_lchan_to_hLayer(lchan);
    act_req->hLayer3 = act_req->hLayer2;

    /* ... */
}```

According to 3GPP TS 45.002, table 6, PTCCH/U has 16 sub-channels (TAI=0..15), and `GsmL1_SubCh_t` defines exactly 16 sub-channels!

```typedef enum GsmL1_SubCh_t
{
    GsmL1_SubCh_0 = 0,    ///< Sub-channel 0
    GsmL1_SubCh_1,        ///< Sub-channel 1
    GsmL1_SubCh_2,        ///< Sub-channel 2
    /* ... */
    GsmL1_SubCh_14,       ///< Sub-channel 14
    GsmL1_SubCh_15,       ///< Sub-channel 15

    GsmL1_SubCh_NA = 0x1F  ///< No sub-channel (not applicable)
} GsmL1_SubCh_t;```

but currently we always send `GsmL1_SubCh_NA` (see `lchan_to_GsmL1_SubCh_t()`). We should either activate all 16 sub-channels in osmo-bts-[sysmo,oc2g,litecell15], or do it directly from osmo-pcu as soon as we assign a TAI to a subscriber, i.e. on demand. This also needs to be tested though.

### #4 - 04/18/2020 11:15 AM - fixeria
- Related to Bug #4501: osmo-bts-[sysmo,oc2g,litecell15]: Packet Control Ack (in form of Access Bursts) arrives on PRACH?? added

### #5 - 04/18/2020 03:45 PM - keith

From experiments yesterday, I confirm that adding
if (sapi & GsmL1_Sapi_Ptcch && dir & GsmL1_Dir_RxUplink) 
    act_req->subCh = GsmL1_SubCh_0;

in mph_send_activate_req() for case GsmL1_Sapi_Ptcch

This results in Activation success, but immediately followed by:

Error activating L1 SAPI PACCH on TS 6: Invalid parameter

As if once we have set GsmL1_SubCh_0 on the PTCCH/U it no longer accepts the PACCH activation.

I tried to bypass the PACCH activation as you suggested on IRC (I faked an activation confirmation) but that just runs into more problems later.

#6 - 04/18/2020 04:00 PM - fixeria

Hi keith,

Error activating L1 SAPI PACCH on TS 6: Invalid parameter

do you see this message with GsmL1_Sapi_Pacch enabled on GsmL1_Dir_RxUplink? If so, let's comment it out (as it's done in the current master) and focus on GsmL1_Sapi_Ptcch for now. The "PACCH/U vs PRACH" problem to be discussed in #4501.

#7 - 04/18/2020 04:23 PM - keith

fixeria wrote:

Hi keith,

Error activating L1 SAPI PACCH on TS 6: Invalid parameter

do you see this message with GsmL1_Sapi_Pacch enabled on GsmL1_Dir_RxUplink?

Ah, now i see you mean in the struct sapi_dir pdtch_sapis[]

So yes, that's correct.

If so, let's comment it out (as it's done in the current master)

OK. Yes, now I can activate the PTCCH/U and phone can GPRS attach.

EDIT: and PCU logs now: (I know what to do, but got to go AFK for a few hours)

osmo-bts-sysmo/sysmo_l1_if.c:213 Rx PH-DATA.ind for unknown L1 SAPI PTCCH

will try to get more on this later today.

#8 - 04/18/2020 04:51 PM - fixeria

- % Done changed from 10 to 40

- OK. Yes, now I can activate the PTCCH/U and phone can GPRS attach.

Yay! One more step towards the light in this SAPI-hell ;)

This also means that we can now (de)activate PTCCH/U for particular TAI on demand. Nice!

osmo-bts-sysmo/sysmo_l1_if.c:213 Rx PH-DATA.ind for unknown L1 SAPI PTCCH

Huh, I thought it would be PH-RA.ind. This is not a big problem though. Would be nice to see the whole payload of this indication (using gdb).

#9 - 04/18/2020 08:52 PM - keith
I wish I knew about the ‘dsp-trace-flag’ before. The DSP actually tells us (see /dev/rtfifo/dsp_trace) the exact cause of error:

MphActivateReq =>$ hLayer1 = 0x87080020, hLayer2 = 0x600BB, u8Tn = 6, sapi = 21, dir = 2, subCh = 31, bfi = 0.00 dBm
cfgParam = 0000003F 00000000 00000000 00000000
[ERROR] : LogChMng_GetTypeXIIIBufAddr() => Invalid SubCh (31) Valid is [GsmL1_SubCh_0 to _15]

MphActivateCnf =>$ hLayer3 = 0x600BB, hLayer1 = 0x87080020, TN = 6, sapi = 21, dir = 2, subCh = 31, status = -4

achieved by adding the following to osmo-bts-sysmo.conf:

phy 0
  instance 0
    dsp-trace-flag error
    dsp-trace-flag debug
    dsp-trace-flag mph_req
    dsp-trace-flag mph_ind
    dsp-trace-flag mph_cnf

P.S. make sure to apply https://gerrit.osmocom.org/c/osmo-bts/+17946 before using it.