OsmoBSC - Bug #2719

OsmoBSC doesn't send BCCH filling after RSL connection unless BTS sends unsolicited message

12/06/2017 06:42 PM - laforge

Status:	Resolved	**Start date:**	12/06/2017
Priority:	Normal	**Due date:**	
Assignee:	stsp	**% Done:**	50%
Category:	A-bis RSL		
Target version:			
Spec Reference:			

Description

The S_L_INP_TEI_UP signal is generated for Abis/IP in OsmoBSC only after the first RSL message is received from the BTS, see bts_ipaccess_nanobts:ipaccess_sign_link()

Usually this works as the known BTS models are sending something like an RF RESOURCE indication, but this is quite ugly

Related issues:

Related to libosmo-abis - Bug #2718: ipaccess_bts_handle_ccm() gets ID_REQ/ID...	**Resolved**	**12/06/2017**
Has duplicate OsmoBTS - Bug #3030: sysinfo.c:157 PH-RTS-IND: Unable to determ...	**Resolved**	**03/02/2018**

History

#1 - 12/06/2017 06:42 PM - laforge

- Related to Bug #2718: ipaccess_bts_handle_ccm() gets ID_REQ/ID_RESP/ID_ACK wrong added

#2 - 12/10/2017 03:35 PM - laforge

#3 - 12/10/2017 07:56 PM - laforge

- Project changed from OpenBSC to OsmoBSC

- Category deleted (libbsc)

#4 - 12/10/2017 08:15 PM - laforge

- Category set to A-bis RSL

#5 - 01/11/2018 05:08 PM - laforge

- Assignee changed from laforge to stsp

#6 - 02/12/2018 03:32 PM - stsp

- Status changed from New to In Progress

Initial patch proposed in https://gerrit.osmocom.org/#/c/6387/

#7 - 02/19/2018 01:57 PM - stsp

This issue is waiting for a test of above patch with a real BTS setup, which I don't have ready at the moment.
I will look into testing this change in a real BTS setup eventually if nobody else beats me to it.

#8 - 02/28/2018 05:18 PM - stsp

- File sysmobts-os2719-without-patch.pcapng added

- File sysmobts-os2719-with-patch.pcapng added

A sysmobts seems to work fine with this change.

I'm attaching two pcap files: One with stock osmo-bsc from master, and one with the patch applied to osmo-bsc.
Both show messages exchanged between osmo-bsc and the sysmobts while it is booting up.
The RF active LED on the sysmobts turned itself on in both cases.

#9 - 03/01/2018 04:25 PM - laforge

- Status changed from In Progress to Resolved

- % Done changed from 0 to 100

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

- Status changed from Resolved to New

- % Done changed from 100 to 50

Unfortunately the proposed patch breaks RSL establishment completely.

With osmo-bts-virtual and osmo-bts master, On BTS connect, I get:

```
DOML <0001> oml.c:441 Sending FOM ACK.
DOML <0001> oml.c:1025 OC=RADIO-CARRIER(02) INST=(00,00,ff) Rx CHG ADM STATE
DOML <0001> oml.c:1051 ADM state already was Unlocked
DOML <0001> oml.c:997 OC=RADIO-CARRIER(02) INST=(00,00,ff) Rx OPSTART
DOML <0001> oml.c:1008 ... automatic ACK, OP state already was Enabled
DOML <0001> oml.c:1025 OC=BASEBAND-TRANSCEIVER(04) INST=(00,00,ff) Rx CHG ADM STATE
DOML <0001> oml.c:997 OC=BASEBAND-TRANSCEIVER(04) INST=(00,00,ff) Rx OPSTART
DOML <0001> oml.c:344 OC=BASEBAND-TRANSCEIVER INST=(00,00,ff) AVAIL STATE OK -> OK
DOML <0001> oml.c:351 OC=BASEBAND-TRANSCEIVER INST=(00,00,ff) OPER STATE NULL -> Enabled
DOML <0001> oml.c:312 OC=BASEBAND-TRANSCEIVER INST=(00,00,ff) Tx STATE CHG REP
DOML <0001> oml.c:1408 OC=BASEBAND-TRANSCEIVER(04) INST=(00,00,ff) Rx IPACCESS(0xe0): DOML <0001> oml.c:1365 R
x IPA RSL CONNECT IP=127.0.0.1 PORT=3003 STREAM=0x00
DOML <0001> oml.c:441 Sending FOM ACK.
DLINP <0012> input/ipa.c:131 127.0.0.1:3003 connection done
DLINP <0012> input/ipaccess.c:708 received ID get from 1234/0/0
DABIS <000d> abis.c:113 RSL Signalling link for TRX0 up
DRSL <0000> rsl.c:271 Tx RSL RF RESource INDication
DL1P <0007> sysinfo.c:157 PH-RTS-IND: Unable to determine actual BS_AG_BLKS_RES value as SI3 is not available
yet, fallback to 1
DL1P <0007> sysinfo.c:157 PH-RTS-IND: Unable to determine actual BS_AG_BLKS_RES value as SI3 is not available
yet, fallback to 1
DL1P <0007> sysinfo.c:157 PH-RTS-IND: Unable to determine actual BS_AG_BLKS_RES value as SI3 is not available
yet, fallback to 1
DL1P <0007> sysinfo.c:157 PH-RTS-IND: Unable to determine actual BS_AG_BLKS_RES value as SI3 is not available
yet, fallback to 1
DL1P <0007> sysinfo.c:157 PH-RTS-IND: Unable to determine actual BS_AG_BLKS_RES value as SI3 is not available
yet, fallback to 1
```

No BCCH INFO and no SACCH FILL are sent on RSL establishment, rendering the cell completely unusable. Reverting the patch produces the expected output:

```
DOML <0001> oml.c:1408 OC=BASEBAND-TRANSCEIVER(04) INST=(00,00,ff) Rx IPACCESS(0xe0): DOML <0001> oml.c:1365 R
x IPA RSL CONNECT IP=127.0.0.1 PORT=3003 STREAM=0x00
DOML <0001> oml.c:441 Sending FOM ACK.
DLINP <0012> input/ipa.c:131 127.0.0.1:3003 connection done
DLINP <0012> input/ipaccess.c:708 received ID get from 1234/0/0
DABIS <000d> abis.c:113 RSL Signalling link for TRX0 up
DRSL <0000> rsl.c:271 Tx RSL RF RESource INDication
DL1P <0007> sysinfo.c:157 PH-RTS-IND: Unable to determine actual BS_AG_BLKS_RES value as SI3 is not available
yet, fallback to 1
DL1P <0007> sysinfo.c:157 PH-RTS-IND: Unable to determine actual BS_AG_BLKS_RES value as SI3 is not available
yet, fallback to 1
DRSL <0000> rsl.c:2543 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:322  Rx RSL BCCH INFO (SI1, 23 bytes)
```

```
DPAG <0005> paging.c:540 Paging SI update
DRSL <0000> rsl.c:2543 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:322  Rx RSL BCCH INFO (SI2, 23 bytes)
DPAG <0005> paging.c:540 Paging SI update
DRSL <0000> rsl.c:2543 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:383  RX RSL Disabling BCCH INFO (SI2bis)
DPAG <0005> paging.c:540 Paging SI update
DRSL <0000> rsl.c:2543 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:322  Rx RSL BCCH INFO (SI2ter, 23 bytes)
DPAG <0005> paging.c:540 Paging SI update
DRSL <0000> rsl.c:2543 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:383  RX RSL Disabling BCCH INFO (SI2quater)
DPAG <0005> paging.c:540 Paging SI update
DRSL <0000> rsl.c:2543 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:322  Rx RSL BCCH INFO (SI3, 23 bytes)
DPAG <0005> paging.c:540 Paging SI update
DRSL <0000> bts.c:405 Updated AGCH max queue length to 12
DPCU <0009> pcu_sock.c:123 Sending info
DPCU <0009> pcu_sock.c:138 BTS is up
DPCU <0009> pcu_sock.c:230 trx=0 ts=7: available (tsc=7 arfcn=871)
DPCU <0009> pcu_sock.c:689 PCU socket not connected, dropping message
DRSL <0000> rsl.c:2543 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:322  Rx RSL BCCH INFO (SI4, 23 bytes)
DPAG <0005> paging.c:540 Paging SI update
DPCU <0009> pcu_sock.c:123 Sending info
DPCU <0009> pcu_sock.c:138 BTS is up
DPCU <0009> pcu_sock.c:230 trx=0 ts=7: available (tsc=7 arfcn=871)
DPCU <0009> pcu_sock.c:689 PCU socket not connected, dropping message
DRSL <0000> rsl.c:2543 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:322  Rx RSL BCCH INFO (SI13, 23 bytes)
DPCU <0009> pcu_sock.c:689 PCU socket not connected, dropping message
DPCU <0009> pcu_sock.c:565 Failed to send SI13 to PCU: -5
DPAG <0005> paging.c:540 Paging SI update
DPCU <0009> pcu_sock.c:123 Sending info
DPCU <0009> pcu_sock.c:138 BTS is up
DPCU <0009> pcu_sock.c:230 trx=0 ts=7: available (tsc=7 arfcn=871)
DPCU <0009> pcu_sock.c:689 PCU socket not connected, dropping message
DRSL <0000> rsl.c:555  Rx RSL SACCH FILLING (SI5, 19 bytes)
DPAG <0005> paging.c:540 Paging SI update
DPCU <0009> pcu_sock.c:123 Sending info
DPCU <0009> pcu_sock.c:138 BTS is up
DPCU <0009> pcu_sock.c:230 trx=0 ts=7: available (tsc=7 arfcn=871)
DPCU <0009> pcu_sock.c:689 PCU socket not connected, dropping message
DRSL <0000> rsl.c:559  Rx RSL Disabling SACCH FILLING (SI5bis)
DPAG <0005> paging.c:540 Paging SI update
DPCU <0009> pcu_sock.c:123 Sending info
DPCU <0009> pcu_sock.c:138 BTS is up
DPCU <0009> pcu_sock.c:230 trx=0 ts=7: available (tsc=7 arfcn=871)
DPCU <0009> pcu_sock.c:689 PCU socket not connected, dropping message
DRSL <0000> rsl.c:555  Rx RSL SACCH FILLING (SI5ter, 19 bytes)
DPAG <0005> paging.c:540 Paging SI update
DPCU <0009> pcu_sock.c:123 Sending info
DPCU <0009> pcu_sock.c:138 BTS is up
DPCU <0009> pcu_sock.c:230 trx=0 ts=7: available (tsc=7 arfcn=871)
DPCU <0009> pcu_sock.c:689 PCU socket not connected, dropping message
DRSL <0000> rsl.c:555  Rx RSL SACCH FILLING (SI6, 13 bytes)
DPAG <0005> paging.c:540 Paging SI update
DPCU <0009> pcu_sock.c:123 Sending info
DPCU <0009> pcu_sock.c:138 BTS is up
DPCU <0009> pcu_sock.c:230 trx=0 ts=7: available (tsc=7 arfcn=871)
DPCU <0009> pcu_sock.c:689 PCU socket not connected, dropping message
^Csignal 2 received
```

#11 - 03/02/2018 04:02 PM - fixeria

- Has duplicate Bug #3030: sysinfo.c:157 PH-RTS-IND: Unable to determine actual BS_AG_BLKS_RES value as SI3 is not available yet, fallback to 1 added

#12 - 03/20/2018 01:15 PM - stsp

For the record, commit 383a059a123b1e0e5aab76423db47846e329f095 reverted the problematic patch from https://gerrit.osmocom.org/#/c/6387/ (commit faf0982ae20001519cf20c5d6345dad490a135f2).

#13 - 03/22/2018 12:03 PM - stsp

- Status changed from New to In Progress

#14 - 03/22/2018 03:47 PM - stsp

It seems the problem with the reverted patch was that messages queued on the RSL link during sign_link_up() are never transmitted.

I can get the reverted patch to work as it was, if I add the patch below to libosmo-abis.
This doesn't seem like the right approach (it triggers Tx during Rx processing), but maybe someone could point me into the direction of a proper fix?

```
diff --git a/src/input/ipaccess.c b/src/input/ipaccess.c
index 5eee57e..3e74a9e 100644
--- a/src/input/ipaccess.c
+++ b/src/input/ipaccess.c
@@ -84,6 +84,8 @@ static int ipaccess_drop(struct osmo_fd *bfd, struct e1inp_line *line)
     return ret;
 }

+static int handle_ts1_write(struct osmo_fd *bfd);
+
 static int ipaccess_rcvmsg(struct e1inp_line *line, struct msgb *msg,
                struct osmo_fd *bfd)
 {
@@ -204,6 +206,10 @@ static int ipaccess_rcvmsg(struct e1inp_line *line, struct msgb *msg,
                      "could not register FD\n");
                 goto err;
             }
+            while (!llist_empty(&sign_link->tx_list)) {
+                LOGP(DLINP, LOGL_NOTICE, "There is a message on sign_link->tx_list!\n");
+                handle_ts1_write(newbfd);
+            }
             /* now we can release the dummy RSL line. */
             e1inp_line_put(line);
         }
```

The osmo-bsc log then shows:

```
<0003> abis_rsl.c:277 sending SAACH filling to BTS0/TRX0
<0003> bsc_init.c:128 SI6: 2d 06 1e 00 00 09 f1 07 00 17 27 ff 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b

<0003> abis_rsl.c:277 sending SAACH filling to BTS0/TRX0
<0015> osmo_bsc_ctrl.c:149 TEI_UP received
<0013> input/ipaccess.c:212 There is a message on sign_link->tx_list!
<0013> input/ipaccess.c:212 There is a message on sign_link->tx_list!
<0013> input/ipaccess.c:212 There is a message on sign_link->tx_list!
<0013> input/ipaccess.c:212 There is a message on sign_link->tx_list!
<0013> input/ipaccess.c:212 There is a message on sign_link->tx_list!
<0013> input/ipaccess.c:212 There is a message on sign_link->tx_list!
<0013> input/ipaccess.c:212 There is a message on sign_link->tx_list!
<0013> input/ipaccess.c:212 There is a message on sign_link->tx_list!
<0013> input/ipaccess.c:212 There is a message on sign_link->tx_list!
<0013> input/ipaccess.c:212 There is a message on sign_link->tx_list!
<0013> input/ipaccess.c:212 There is a message on sign_link->tx_list!
<0013> input/ipaccess.c:212 There is a message on sign_link->tx_list!
<0008> a_reset.c:92 A-RESET(msc-0)[0x56493acd4850]{DISC}: (re)sending BSSMAP RESET message...
```

```
<0008> osmo_bsc_sigtran.c:92 Sending RESET to MSC: RI=SSN_PC,PC=0.23.1,SSN=BSSAP

<0008> osmo_bsc_bssap.c:201 RESET ACK from MSC: RI=SSN_PC,PC=0.23.1,SSN=BSSAP

<0008> a_reset.c:60 A-RESET(msc-0)[0x56493acd4850]{DISC}: SIGTRAN connection succeded.
```

And an osmo-virtual-bts shows:

```
DL1P <0007> sysinfo.c:158 PH-RTS-IND: Unable to determine actual BS_AG_BLKS_RES value as SI3 is not available
yet, fallback to 1
DL1C <0006> scheduler.c:585 Configuring multiframe with TCH/F+SACCH trx=0 ts=5
DL1P <0007> sysinfo.c:158 PH-RTS-IND: Unable to determine actual BS_AG_BLKS_RES value as SI3 is not available
yet, fallback to 1
DL1P <0007> sysinfo.c:158 PH-RTS-IND: Unable to determine actual BS_AG_BLKS_RES value as SI3 is not available
yet, fallback to 1
DL1C <0006> scheduler.c:585 Configuring multiframe with PDCH trx=0 ts=6
DL1C <0006> scheduler.c:585 Configuring multiframe with PDCH trx=0 ts=7
DL1C <0006> bts_model.c:93 Unimplemented vbts_set_trx
DOML <0001> oml.c:1049 ADM state already was Unlocked
DLINP <0012> input/ipa.c:131 127.0.0.1:3003 connection done
DLINP <0012> input/ipaccess.c:716 received ID get from 6969/0/0
DRSL <0000> rsl.c:271 Tx RSL RF RESource INDication
DRSL <0000> rsl.c:2554 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:322  Rx RSL BCCH INFO (SI1, 23 bytes)
DRSL <0000> rsl.c:2554 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:322  Rx RSL BCCH INFO (SI2, 23 bytes)
DRSL <0000> rsl.c:2554 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:383  RX RSL Disabling BCCH INFO (SI2bis)
DRSL <0000> rsl.c:2554 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:383  RX RSL Disabling BCCH INFO (SI2ter)
DRSL <0000> rsl.c:2554 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:383  RX RSL Disabling BCCH INFO (SI2quater)
DRSL <0000> rsl.c:2554 (bts=0,trx=0,ts=0,ss=0) Rx RSL BCCH_INFO
DRSL <0000> rsl.c:322  Rx RSL BCCH INFO (SI3, 23 bytes)
DRSL <0000> bts.c:400 Updated AGCH max queue length to 12
```

#15 - 03/22/2018 05:40 PM - laforge

Hi stsp,

I think you simply need to find the right place to set the BSC_FD_WRITE flag, at which time the socket will be marked in the select() readset, and the read-callback should drain the queue.

Where exactly to place that, I don't know withou t studying the details of the related code.

If it uses osmo_wqueue_enqueue(), this should take care of it?

Maybe we simply either

a) simply forget/overwrite BSC_FD_WRITE when a new connection comes in, or
b) never have encountered the case before that we have data pending for transmit before the connection arrives?

#16 - 03/22/2018 06:03 PM - stsp

Nice hint, thank you. This patch works as well, and I suppose this looks acceptable? If so, I'll submit it to gerrit.

```
diff --git a/src/input/ipaccess.c b/src/input/ipaccess.c
index 5eee57e..9a1485c 100644
--- a/src/input/ipaccess.c
+++ b/src/input/ipaccess.c
@@ -206,6 +206,11 @@ static int ipaccess_rcvmsg(struct e1inp_line *line, struct msgb *msg,
             }
             /* now we can release the dummy RSL line. */
             e1inp_line_put(line);
+
+            /* sign_link_up() might already have queued outbound messages. */
+            if (!llist_empty(&sign_link->tx_list))
+                newbfd->when |= BSC_FD_WRITE;
+
         }
         break;
     default:
```

#17 - 03/22/2018 06:38 PM - laforge

On Thu, Mar 22, 2018 at 06:03:25PM +0000, stsp [REDMINE] wrote:

> Nice hint, thank you. This patch works as well, and I suppose this looks acceptable? If so, I'll submit it to gerrit.

I still don't get why we do this in the receive/read path, and not when we enqueeu the message?
Or if we do it when enqueueing the message, why does it get lost?

#18 - 03/22/2018 07:02 PM - stsp

I have done some more digging and I think I have now figured out the root cause.

The underlying bug seems to be in this line of ipaccess_rcvmsg():

```
                        /* get rid of our old temporary bfd */
                        memcpy(newbfd, bfd, sizeof(*newbfd));
```

Before this memcpy() call newbfd->when equals 0x02 (BSC_FD_WRITE).
But bdf->when is 0x01 (BSC_FD_READ), so memcpy clears the FD_WRITE flag in newbfd->when.

```
<0013> input/ipaccess.c:196 (before memcpy) bfd->when = 0x1
<0013> input/ipaccess.c:197 (before memcpy) newbfd->when = 0x2
<0013> input/ipaccess.c:213 (after memcpy) bfd->when = 0x1
<0013> input/ipaccess.c:214 (after memcpy) newbfd->when = 0x1
```

It seems this code needs to be more careful about preserving information in newbfd. I'll try to write a patch for that.

#19 - 03/22/2018 07:29 PM - stsp

I have proposed a fix for libomos-abis in https://gerrit.osmocom.org/7462

#20 - 03/26/2018 11:21 AM - stsp

Above fix has been merged. Re-proposed the initial fix at https://gerrit.osmocom.org/7503

#21 - 04/09/2018 09:19 AM - stsp

- Status changed from In Progress to Resolved

Above fix has been merged.

Files

sysmobts-os2719-without-patch.pcapng	21.6 KB	02/28/2018	stsp
sysmobts-os2719-with-patch.pcapng	21.1 KB	02/28/2018	stsp