TCH Rx error flags: outline of presentation

* Abis TRAU-UL frames and mobile-side equivalent
* The simpler case: TCH/FS and TCH/EFS (FRv1 & EFR)
* BFI flag and Rx SID classification
* TCH/HS first complication: new UFI error flag
* Reference code for SID classification: what's that BCI flag?
* More clues from TI Calypso DSP implementation
* Why did ETSI lie by omission?
* Relevance for Osmocom

Presented by:

Mother Mychaela N. Falconia,
High Priestess of Telecommunications,
Women's Republic of Themyscira

Abis on E1: what does a traditional BTS put out for TCH Rx?

TRAU-UL frame format of GSM 08.60 (full-rate TCH/FS & TCH/EFS):

* The appropriate number of traffic frame payload bits
* BFI (Bad Frame Indicator) flag
* Rx SID classification status (valid SID / invalid SID / non-SID speech)
* Time Alignment Flag (TAF)

TRAU-UL frame format of GSM 08.61 (half-rate TCH/HS):

* All of the above, plus:
+ UFI (Unreliable Frame Indicator) flag

Interesting bits:

* What are the criteria for setting BFI and UFI?
* How is Rx SID classification done?
* What are frame payload bits set to when there is none?

GSM specs: abstract vs concrete definition of TCH Rx interface

TCH Rx interface is defined in _concrete_ form (exact bit definitions) for network side only, as in Abis - covered on previous slide.

TCH Rx interface is defined in _abstract_ form (semantics, flag definitions, but no frame formats) for both network and mobile sides of GSM! It is fully symmetric between uplink and downlink. :)

Abstract form defined in:

GSM 06.31: TCH/FS for FRv1
GSM 06.41: TCH/HS for HRv1
GSM 06.81: TCH/EFS for EFR

In each of these 3 specs, the block diagram of TCH Rx interface is the first figure at the beginning of Chapter 6 - so consistent!

Reference C code implementations from ETSI (HRv1 and EFR codecs) are designed for the abstract interfaces from these specs:

* Encoder test sequences follow Chapter 5 Tx interface
* Decoder test sequences follow Chapter 6 Rx interface

TI Calypso DSP: Abis-like interface on mobile side

On the network side:

```
+--------+                     +---------+---------+
| Radio  |   Abis interface    | Rx      |         |
| Sub    | ------------------> | DTX     | Speech  |
| System |   TRAU-UL frames    | Handler | Decoder |
+--------+                     +---------+---------+
(in the BTS)                   (in the TRAU)
```

On the mobile side (TI Calypso implementation):

```
+--------+                     +---------+---------+
| Radio  |   internal i/f      | Rx      |         |
| Sub    | ------------------> | DTX     | Speech  |
| System |   (but sniffable!)  | Handler | Decoder |
+--------+                     +---------+---------+
      (all inside one DSP, *seems* monolithic)
```

By sniffing the internal interface of TCH DL (reading words from shared memory at just right times) we can learn a lot!

The simple case: TCH/FS

TRAU-UL frames on network side:

* 260 bits of FRv1 payload, in natural order of codec parameters
* BFI and SID metadata bits as already covered

a_dd_0 buffer interface in TI Calypso DSP:

* a_dd_0[0]: word of status flags
* a_dd_0[1]: D_MACC unknown
* a_dd_0[2]: convolutional decoder bit error count
* remaining 17 words: FRv1 payload, in GSM 05.03 channel coding bit order

a_dd_0[0] status flag bits:

```
bits [1:0]: unused in TCH/F - we’ll see later what they do in TCH/HS
bit   2   : BFI just like metadata bit C12 in TRAU-UL
bits [4:3]: SID classification just like C13-C14 in TRAU-UL
...
additional bits: FACCH status and other stuff not covered here
```

The simple case: TCH/FS (continued)

What are the criteria for setting BFI on TCH/FS Rx, be it TRAU-UL bit C12 or TI DSP a_dd_0[0] bit 2?

- FACCH received instead of traffic frame ==> BFI
- Bad CRC-3 in GSM 05.03 channel decoding ==> BFI
- ... but other (unknown) criteria must be present too!

When a DTX half-block is received (4 out of 8 bursts missing), both Nokia InSite BTS and TI DSP set BFI - and standard Rx DTX handler logic depends on this behavior - but what is the actual criterion?

OTOH, Rx SID classification for TCH/FS (FRv1) and TCH/EFS (EFR) is straightforward:

```
* at most 1 bit error in SID field:   SID=2 output
* 2 to 15 bit errors in SID field:    SID=1 output
* 16 or more bit errors in SID field: SID=0 output
```

(codified in GSM 06.31 for TCH/FS and GSM 06.81 for TCH/EFS)

TCH/HS: the specs get muddier

Compared to just-covered TCH/(E)FS, the specs for TCH/HS get more complicated:

* In addition to Bad Frame Indicator (BFI), we now also have Unreliable Frame Indicator (UFI), with completely unknown criteria.

* This new UFI flag is included in TRAU-UL frame format of GSM 08.61.

GSM 06.06 C code tells us what the Rx DTX handler and the speech decoder do with this UFI flag, but no guidance is given as to when the Rx radio subsystem should set it!

* Rx SID classification is still the same 3 possibilities, but no more exact rules in the spec. :-(

GSM 06.41 spec refers to GSM 06.06 C code for "an example solution" of how to compute Rx SID classification - but it is treated as a non-normative example.

GSM 06.06: reference C code for GSM-HR codec

The full body of C code can be divided into 3 parts:

* The speech encoder with optional DTX: normatively bit-exact, exercised by GSM 06.07 test sequences.

* The speech decoder with built-in Rx DTX handler front end: also normatively bit-exact in all aspects other than ECU, likewise exercised by GSM 06.07 test sequences.

* Residual Error Insertion Device (reid.c): used to generate *.dec from *.cod in the official test sequence set, but NOT normative in itself!

The non-normative "example solution" for Rx SID classification resides in REID utility - the least documented, most glossed-over part... Nice job, ETSI! /s

A look at swSidDetection() function

The function that performs Rx SID classification - excerpt from comment block before the function body:

```
 |     Input:     pswParameters[18]                                          |
 |                           input parameters of the speech decoder          |
 |                                                                           |
 |                pswErrorFlag[3]                                            |
 |                           error flags, generated by channel decoder       |
```

Code snippet that calls this function:

```
    /* Evaluate SID flag */
    /* ----------------- */
    pswErrorFlag[0] = 0;   /* BFI flag */
    pswErrorFlag[1] = 0;   /* UFI flag */
    pswErrorFlag[2] = 0;   /* BCI flag */
    swOutPara[20] = swSidDetection(swOutPara, pswErrorFlag);
```

We know what BFI and UFI are - while the spec comes short of telling us how to set these flags, at least it describes what they are, and they appear in TRAU-UL frames - but what is that third error flag BCI?

BCI flag: how does it impact Rx SID classification?

All 3 classic codecs have class 1 bits (protected by convolutional coding) and class 2 bits (unprotected). In FRv1 and EFR, the SID field resides entirely in class 1 - hence all SID field bits are equal for counting and classification purposes.

HRv1 SID field uses class 1 and class 2 bits - the "example" code from ETSI gives them unequal weight in Rx SID classification.

Logic for BCI=0:

```
    if (swSidN1 < 3)
       swSid = 2;
    else if (swSidN1pN2 < 11)
       swSid = 1;
    else
       swSid = 0;
```

Logic for BCI=1:

```
    if (swSidN1 < 3)
       swSid = 2;
    else if (swSidN1pN2 < 16)
       swSid = 1;
    else
       swSid = 0;
```

The threshold between invalid SID and non-SID speech (which counts both class 1 and class 2 bits) changes based on this mystery BCI flag.

TI Calypso DSP to the rescue

When I look at TCH DL captures from HS channels, any time a BFI frame appears, I see C007 in the first status word, right where I would see C004 or C204 with TCH/FS or TCH/EFS. It sure looks like TI’s DSP implements all 3 error flags for TCH/HS, not just one or two. :)

After some experimentation, I determined that a_dd_x[0] flags for TCH/HS are:

bit 0: UFI
bit 1: BCI
bit 2: BFI
bits [4:3]: ternary SID classification

Furthermore, SID classification logic implemented by this DSP is exactly the same as ETSI reference (sorry, "example") version, including the effect of a_dd_x[0] bit 1 (BCI) on the 11 vs 16 threshold - yay!

Error severity hierarchy

The definition of BFI and UFI error flags makes it clear that BFI is a more severe error than UFI - but where does BCI fit in?

Experiment with R&S CMU200 test instrument:

* CMU200 configured to emulate a GSM network
* Coax from the CMU to the antenna connector on FCDEV3B
* Test call between FreeCalypso MS and the CMU, HR codec selected

As I turned the knob to reduce DL signal level, I exercised the full range of Rx conditions from perfectly good to radio link failure - and we can see how errors in TCH/HS channel-decoded frames progress.

TCH/HS received by Calypso DSP - error progression:

-> No errors
-> BCI
-> BCI+UFI
-> BCI+UFI+BFI

BCI "upgrade" to BFI

GSM 06.41 spec wording (section 6.1.1):

> NOTE: That the BFI flag, which is generated by the channel error correction scheme, will in some cases be modified by the SID frame detection unit.

Code path in GSM 06.06 reid.c:

```
  if (pswErrorFlag[2]) {        /* BCI */

    /* SID threshold logic for BCI=1, shown earlier */

    if ( (swSidN1pN2 >= 16) && (swSidN1pN2 <= 25) ) {
      pswErrorFlag[0] = 1;      /* BFI */
    }
  }
```

Effect: with certain bit patterns, lowest-severity BCI is "upgraded" to highest-severity BFI! But why?

TI Calypso DSP (ROM 3606, patch version 6840) omits this logic. :)

Why did ETSI lie by omission?

Why is BCI error flag not documented anywhere? We don’t even know what BCI stands for: Bad Channel Indicator maybe?

Omission of BCI from TRAU-UL frames makes sense: this flag only affects the Rx SID classification block, and that step is done in the CCU before TRAU-UL output – but still, it should have been *documented* as a channel decoder output flag, used by the SID detector block.

History books tell us that GSM-HR codec was developed by Motorola – supposedly there were several competing candidates, and Motorola’s proposal won the coveted standardization. But did Motorola develop only the codec itself, or also the associated TCH/HS channel coding and error flagging scheme? Were both UFI and BCI Motorola inventions?

Did ETSI then omit BCI from all official specs on the reasoning that it was a private detail of one particular implementation, not needed for interoperation?

If so, why is TI Calypso DSP implementation so close to this unpublished canon? Did TI license their implementation directly from Motorola?

Relevance for Osmocom

As of just recently, osmo_hr_sid_classify() in libosmocodec implements the same logic as ETSI’s swSidDetection(). BCI flag input is there, as well as the optional logic that "upgrades" BCI to BFI. Frame payload input is supplied in TS 101 318 format in this version - I designed it this way so it can be used to post-process the output of sysmoBTS PHY black box. Patches to osmo-bts-* still remain to be implemented.

We still don’t know the criteria for setting BCI or UFI, therefore:

* When we integrate osmo_hr_sid_classify() into osmo-bts-*, it will be called with fixed ’false’ for BCI argument;

* UFI bit (present in TW-TS-002 enhanced RTP format) will also be fixed at 0.

On GSM MS side: anyone who is implementing SDR-based replacement for Calypso are also welcome to use this function, and deal with the same problem of having no known criteria for BCI or UFI.

And we need to keep in mind: all this GSM-HR and TCH/HS work is only for a feel-good sense of completeness - practical deployment of HRv1 is not very likely...