Software Defined E1

Harald Welte <laforge@gnumonks.org>

E1 / T1 / TDM

- good old ISDN technology
- 2 Mbits/s (E1) or 1.54 Mbits/s (T1) synchronous, full-duplex
- not used much anymore in telephony (everything moves to SIP/IP)
- still used quite a bit in 2G/3G cellular networks, even in 2018!

E1/T1/TDM in 2G/3G networks

traditionally all interfaces over E1/T1

- Abis (RSL/OML over LAPDm) from BTS to BSC
- A (BSSAP/SCCP/MTP) from BSC to MSC
- ISUP/MTP for calls between MSCs and from/to PSTN
- MAP/TCAP/SCCP/MTP between MSC/VLR, SGSN, HLR, GW-MSC, IW-MSC, ...
- Gb (BSSGP/NS/FR) betewen PCU and SGSN
- Iub (Inverse ATM multiplex) over 4xE1 to RNC

E1/T1/TDM in 2G/3G networks today: Abis

- TDM based Abis on decline
  - back-haul networks increasingly switch TDM to IP as 4G is co-located with 2G
  - but: Lots of BTSs still have physical E1
  - Equipment like Ericsson SIU used to convert E1 to IP (proprietary protocols)
  - TDM link remains between BTS and SIU-style converter

E1/T1/TDM in 2G/3G networks today: A

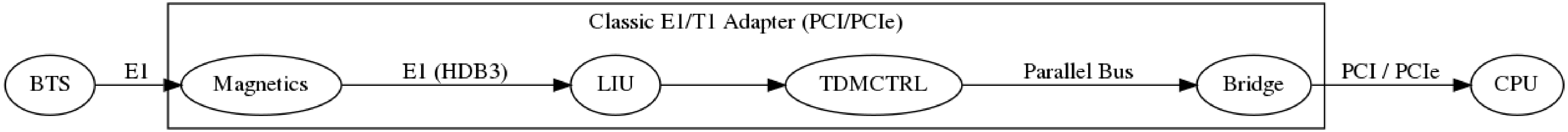
- TDM based A on the decline
  - 3GPP has an official interoperable protocol: AoIP
  - adopted by many more modern MSCs
  - OsmoBSC supports 3GPP AoIP (yay!)

E1/T1/TDM in 2G/3G networks today: Core

- TDM based core network connections still prevalent
  - lots of legacy switches (MSCs) and STPs around
  - signaling interconnect among MNOs and MVNOs often still TDM
  - full MAP+CAP over TCAP/SCCP/MTP stack required

E1/T1/TDM interfacing from Linux / Osmocom

- we’ve had E1/T1 based Abis for ages
- **libosmo-abis** supports mISDN + DAHDI drivers
  - PCI + PCIe cards readily available
  - still extremely expensive (OK in CN, not next to each BTS)
  - PCI cards of course require a rather large (ATX, ITX, …) computer



- TDMCTRL implements equalizer, elastic buffer, CRC, framing, HDLC, …

Osmocom E1/T1/TDM interfacing use-case

- many E1/T1 based BTSs decommissioned around the world
- refurbished traders have quantities in stock for **very** low price
- using those BTSs with OsmoBSC + friends is an inexpensive way of
  - deploying carrier-grade tier-1 BTS equipment
  - with excellent environmental, RF sensitiviy, RF power and high MTBF
  - for very low cost
- but: The E1/T1 card + associated PC are more expensive than your BTS :(

E1/T1/TDM interfacing wishlist

- in 2018, you just want a very small E1/USB or E1/Ethernet adapter
- can be used with laptop when on the road, debugging something
- can be used with Raspberry Pi, Beaglebone or whatever other mall, inexpensive embedded Linux board
- you want to pay a realistic price, not some fantasy price (Digium & co)

Building an E1/T1 adapter

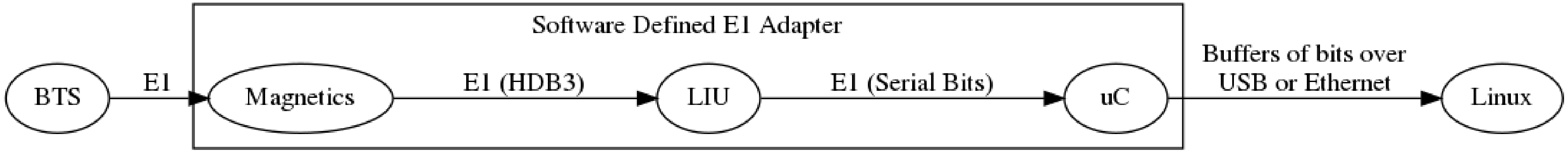
Ok, so let’s build an E1 adapter using existing chips…

- existing E1/T1 controllers
  - many (including Infineon) already EOL due to the demise of ISDN
  - have arcane bus interfaces (parallel Intel/Motorola bus like 1980)
  - are ridiculously expensive
  - come in very large acane packages

The SD-TDM Plan [tm]

The road to software-defined E1:

- Simply use a LIU (Line Interface Unit) + Magnetics
  - this converts the HDB3 ternary signal to a serial bit-stream
- serialize/deserialize that stream from a microcontroller
- do everything else in software, including framing, CRC4, …



Hardware Option A: TI PRU

- TI processors like the AM335x on the Beagleboard have PRU cores
    - PRU: *Programmable Realtime Unit*
- PRU allows high-speed "real time" bit banging
- PRU can serialize/deserialize and provide buffers to ARM core with Linux
- E1 adapter could be a beaglebone cape
- Beaglebone could actually run entire OsmoBSC + OsmoMGW, too, using 3gPP AoIP over back-haul

Hardware Option B: XMOS

- XMOS has a very unusual microcontroller architecture
- RISC CPU core @ 500 MHz with programmable serdes
  - except USB + Ethenet, no other hard peripherals
  - all peripherals (including I2C, SPI, ..) implemented in software!
  - could be a simple / small E1/T1 to USB or to Ethernet converter

Hardware Option C: Programmable Logic

- Using FPGA or CPLD one can of course synthesize a E1 core
- but that’s not really *software defined* anymore
- toolchain trouble (except yosys/arachne/ice40)
- just seems like overkill for a slow 2 Mbits/s signal

Further Reading

- http://osmocom.org/projects/e1-t1-adapter
  - http://osmocom.org/issues/2484
- XMOS
  - https://www.xmos.com/download/private/Introduction-to-XS1-ports%281.0%29.pdf
  - https://www.xmos.com/download/private/XMOS-Programming-Guide-%28documentation%29%28E%29.pdf
- TI PRU
  - http://processors.wiki.ti.com/index.php/Programmable_Realtime_Unit_Subsystem
  - http://processors.wiki.ti.com/images/1/18/PRUSS_Training_Slides.pdf

EOF

End of File