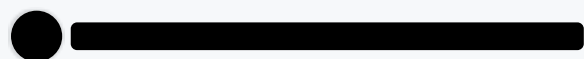


master ▾

proxmark3 / doc / trace_notes.md

Go to file

...



🔗 Notes about the tracelog

🔗 Table of Contents

- [Notes about the tracelog](#)
- [Table of Contents](#)
 - [Trace command](#)
 - [Timing](#)
 - [Sources](#)
 - [Data](#)
 - [CRC](#)
 - [Annotation](#)
 - [Tracelog format](#)
 - [Trace and Wireshark](#)

[↗](#) Trace command

[^Top](#)

The `trace` command lists the data exchange by the proxmark3 and a tag or a reader in human readable form.

With `trace list` a table is shown which gives timing information, the src of the data bytes, the transmitted/received bytes itself, a check if the CRC was correct and some decoding of the command.

[↗](#) Timing

[^Top](#)

The Start and the End column lists timestamps when the transmission of the shown data started (time of first bit) and when it ended (end of last modulation).

The unit for this time information depends on the protocol in use:

- ISO14443A and Thinfilm: all times are in carrier periods (1/13.56MHz)
- For Legic timing information depends also on direction:
- Reader Mode: Timings are in ticks (1us == 1.5ticks)
- Tag Mode: Timings are in sub carrier periods (1/212 kHz == 4.7us)
- Hitag1 / Hitag2 / HitagS: Elementary Time Unit (ETU) is 8μs
- iCLASS, ISO15693, ISO18092 and FeliCa have no accurate timing information at the moment
- For others timing is not available

By specifying the option `f` (e.g. `trace list -t 14a -f`) the frame delay times are shown. (So you don't have to do the math by your own).

[↗](#) Sources

[^Top](#)

If the data is marked as a response the source is shown as Tag. Otherwise it is marked as Reader (Rdr).

[↗](#) Data

[^Top](#)

This column shows the raw bytes transmitted over the air. With option **c** CRC bytes are marked in square brackets.

[↗](#) CRC

[^Top](#)

Marks if the transmitted CRC matches with the calculated CRC.

[↗](#) Annotation

[^Top](#)

Annotations provide a rough decoding of the transmitted data. For ISO14443A a more detailed decoding is available with Wireshark (s. next chapter)

[↗](#) Tracelog format

[^Top](#)

The binary format for the dynamic tracelog is as following.

```
/*
  Traceformat:
  32 bits timestamp (little endian)
  16 bits duration (little endian)
  15 bits data length (little endian) (0x7FFF)
  1 bit isResponse (0=reader to tag, 1=tag to reader)
  data length Bytes data
  x Bytes parity, where x == ceil(data length/8)
```

```

*/

typedef struct {
    uint32_t timestamp;
    uint16_t duration;
    uint16_t data_len : 15;
    bool isResponse : 1;
    uint8_t frame[];
    // data_len          bytes of data
    // ceil(data_len/8) bytes of parity
} PACKED tracelog_hdr_t;

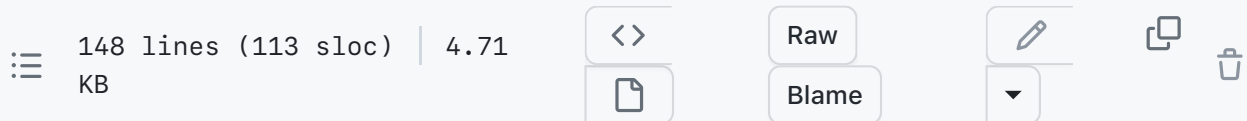
#define TRACELOG_HDR_LEN      sizeof(tracelog_hdr_t)
#define TRACELOG_PARITY_LEN(x) (((x)->data_len - 1) / 8 + 1)

```

Trace and Wireshark

[^Top](#)

To get a more detailed explanation of the transmitted data for ISO14443A traces the output can be converted to a pcapng file to read it with [Wireshark](#).



- copy the output (starting with the timestamp) into a textfile
- run `text2pcap -t "%S." -l 264 -n <input-text-file> <output-pcapng-file>`
- now open your pcapng file in Wireshark or read it with the CLI version `tshark`

An example frame

with `trace list -t 14a :`

```
19072 |          29536 | Rdr | 93  70  88  04  cf  ff  bc  7f  bb
|  ok  | SELECT_UID
```

the same data with `tshark -r foo.pcapng -V -x`:

```

Frame 5: 13 bytes on wire (104 bits), 13 bytes captured (104
bits) on interface 0
    Internet Protocol Version 4, Src: 10.0.0.1, Dest: 10.0.0.2
    Transmission Control Protocol, Src Port: 80, Dest Port: 80
    Hypertext Transfer Protocol
    1
    2
    3
    4
    5
    6
    7
    8
    9
    10
    11
    12
    13
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24
    25
    26
    27
    28
    29
    30
    31
    32
    33
    34
    35
    36
    37
    38
    39
    40
    41
    42
    43
    44
    45
    46
    47
    48
    49
    50
    51
    52
    53
    54
    55
    56
    57
    58
    59
    60
    61
    62
    63
    64
    65
    66
    67
    68
    69
    70
    71
    72
    73
    74
    75
    76
    77
    78
    79
    80
    81
    82
    83
    84
    85
    86
    87
    88
    89
    90
    91
    92
    93
    94
    95
    96
    97
    98
    99
    100
    101
    102
    103
    104
    105
    106
    107
    108
    109
    110
    111
    112
    113
    114
    115
    116
    117
    118
    119
    120
    121
    122
    123
    124
    125
    126
    127
    128
    129
    130
    131
    132
    133
    134
    135
    136
    137
    138
    139
    140
    141
    142
    143
    144
    145
    146
    147
    148
    149
    150
    151
    152
    153
    154
    155
    156
    157
    158
    159
    160
    161
    162
    163
    164
    165
    166
    167
    168
    169
    170
    171
    172
    173
    174
    175
    176
    177
    178
    179
    180
    181
    182
    183
    184
    185
    186
    187
    188
    189
    190
    191
    192
    193
    194
    195
    196
    197
    198
    199
    200
    201
    202
    203
    204
    205
    206
    207
    208
    209
    210
    211
    212
    213
    214
    215
    216
    217
    218
    219
    220
    221
    222
    223
    224
    225
    226
    227
    228
    229
    230
    231
    232
    233
    234
    235
    236
    237
    238
    239
    240
    241
    242
    243
    244
    245
    246
    247
    248
    249
    250
    251
    252
    253
    254
    255
    256
    257
    258
    259
    260
    261
    262
    263
    264
    265
    266
    267
    268
    269
    270
    271
    272
    273
    274
    275
    276
    277
    278
    279
    280
    281
    282
    283
    284
    285
    286
    287
    288
    289
    290
    291
    292
    293
    294
    295
    296
    297
    298
    299
    300
    301
    302
    303
    304
    305
    306
    307
    308
    309
    310
    311
    312
    313
    314
    315
    316
    317
    318
    319
    320
    321
    322
    323
    324
    325
    326
    327
    328
    329
    330
    331
    332
    333
    334
    335
    336
    337
    338
    339
    340
    341
    342
    343
    344
    345
    346
    347
    348
    349
    350
    351
    352
    353
    354
    355
    356
    357
    358
    359
    360
    361
    362
    363
    364
    365
    366
    367
    368
    369
    370
    371
    372
    373
    374
    375
    376
    377
    378
    379
    380
    381
    382
    383
    384
    385
    386
    387
    388
    389
    390
    391
    392
    393
    394
    395
    396
    397
    398
    399
    400
    401
    402
    403
    404
    405
    406
    407
    408
    409
    410
    411
    412
    413
    414
    415
    416
    417
    418
    419
    420
    421
    422
    423
    424
    425
    426
    427
    428
    429
    430
    431
    432
    433
    434
    435
    436
    437
    438
    439
    440
    441
    442
    443
    444
    445
    446
    447
    448
    449
    450
    451
    452
    453
    454
    455
    456
    457
    458
    459
    460
    461
    462
    463
    464
    465
    466
    467
    468
    469
    470
    471
    472
    473
    474
    475
    476
    477
    478
    479
    480
    481
    482
    483
    484
    485
    486
    487
    488
    489
    490
    491
    492
    493
    494
    495
    496
    497
    498
    499
    500
    501
    502
    503
    504
    505
    506
    507
    508
    509
    510
    511
    512
    513
    514
    515
    516
    517
    518
    519
    520
    521
    522
    523
    524
    525
    526
    527
    528
    529
    530
    531
    532
    533
    534
    535
    536
    537
    538
    539
    540
    541
    542
    543
    544
    545
    546
    547
    548
    549
    550
    551
    552
    553
    554
    555
    556
    557
    558
    559
    560
    561
    562
    563
    564
    565
    566
    567
    568
    569
    570
    571
    572
    573
    574
    575
    576
    577
    578
    579
    580
    581
    582
    583
    584
    585
    586
    587
    588
    589
    590
    591
    592
    593
    594
    595
    596
    597
    598
    599
    600
    601
    602
    603
    604
    605
    606
    607
    608
    609
    610
    611
    612
    613
    614
    615
    616
    617
    618
    619
    620
    621
    622
    623
    624
    625
    626
    627
    628
    629
    630
    631
    632
    633
    634
    635
    636
    637
    638
    639
    640
    641
    642
    643
    644
    645
    646
    647
    648
    649
    650
    651
    652
    653
    654
    655
    656
    657
    658
    659
    660
    661
    662
    663
    664
    665
    666
    667
    668
    669
    670
    671
    672
    673
    674
    675
    676
    677
    678
    679
    680
    681
    682
    683
    684
    685
    
```

```
Interface id: 0 (unknown)
```

```
Interface: unknown
```

```

Interface name: unknown
Encapsulation type: ISO 14443 contactless smartcard
standards (177)

```

Arrival Time: Aug 17, 2019 23:17:00.000002606 CEST

```
[Time shift for this packet: 0.000000000 seconds]
```

Epoch Time: 1566076620.000002606 seconds

```
Epoch Time: 15000/6020.0000002000 seconds
[Time delta from previous captured frame: 0.000000840
seconds]
```

```
seconds]
      [Time delta from previous displayed frame: 0.000000840
seconds]
```

```
seconds]
      [Time since reference or first frame: 0.000001907
seconds]
```

Frame Number: 5

Frame Length: 13 bytes (104 bits)

Capture Length: 13 bytes (104 bits)

```
Capture Length: 15 bytes
[Frame is marked: False]
```

```
[Frame is marked: False]
[Frame is ignored: False]
```

```
[Frame is ignored: false]
[Protocols in frame: iso14443]
```

ISO 14443

Pseudo header

```

Pseudo header
  Version: 0x00

```

```

Event: Data transfer PCD -> PICC (0xfe)

```

Length field: 9

Message: Select

```
SEL: 0x93
```

NVB: 0x70

CT: 0x88

UID CLn: 04cfff

```
BCC: 0xbc
```

```

CRC: 0xbb7f [correct]

```

```
[CRC Status: Good]
```

```
0000  00 fe 00 09 93 70 88 04 cf ff bc 7f bb  
.....p.....
```

If the Wireshark ISO14443a dissector is missing some commands or needs some other rework please [file a bug](#).