Data readout: rescue clock mode

Last check point: IDLE words read with ONE output line and saved in the spy buffer.

Now…:

- We are able to control the “START” and the “STOP” of the chip
- We find the Header word and the Frame Counter into the data saved into spy buffer
- We introduce the re-alignment of the PHASE
- Readout work ”well” only with ONE output line
- Odd Output lines do not work (we don’t know why)

```
 70  fcaafcaafcaafcaafcaafcaafcaafcaa
 71  fcaafcaafcaafcaafcaafcaafcaafcaa
 72  fcaafcaafcaafcaafcaafcaafcaafcaa
 73  fcaafcaafcaafcaafcaafcaafcaafcaa
 74  fcaafcaafefefe12fe38fe01fe01fc01
 75  fc01fc01fa743c01fa76f81c6ac0fcab
 76  fc02cc0bf2abf2abf2abf3ffe577e557
 77  e557e557e557e557fcaafcaafcaafcaa
 78  fcaafcaafcaafcaafcaafcaafcaafcaa
 79  fcaafcaafcaafcaafcaafcaafcaafcaa
 80  fcaafcaafcaafcaafcaafcaafcaafcaa
 81  fcaafcaafcaafcaafcaafcaafcaafcaa
 82  fcaafcaafcaafcaafcaafcaafcaafcaa
 83  fcaafcaafcaafcaafcaafcaafcaafcaa
 84  fcaafcaafcaafcaafcaafcaafcaafcaa
 85  fcaafcaafcaafcaafcaafcaafcaafeff
 86  fe12fe71fe01fe00fe00fe00fe00fd3a
 87  3d2cfe9e0fc8fd3f30b9ff005987fd55
 88  fcd5f955fe55fc4bfe57f95bfc6bf8ab
 89  f157f95fcb5f8abf357e557caafcaafc
 90  aafcaafcaafcaafcfcaafcaafcaafeff
 91  7e557e557e557e557e57ff2abf2abf2a
 92  bf2afcaafcaafcaafcaafcaafcaafcaa
 93  fcaafcaafcaafcaafcaafcaafcaafcaa
 94  fcaafcaafcaafcaafcaafcaafcaafcaa
 95  fcaafcaafcaafcaafcaafcaafcaafcaa
 96  fcaafcaafcaafcaafcaafcaafcaafcaa
 97  fcaafcaafcaafcaafcaafcaafcaafcaa
 98  fcaafcaafcaafcaafcaafcaafe00fe13
 99  fe71fe01fe00fe00fe00fe00fd3b3b48
100   fe9e0db9ff00b769f9abe357fc03f84f
101   fb8ff00ff007e00fe00fe00fd388180c
102   8489d8cfba8fcaafd3942bc7eaafcaaf
103   caafcaafcaafcaafcaafcaaffcaafcaa
```

Data readout: PLL mode

			SYNC_IN_N SLVS pads
SEL_START	2-1	0	Start mode selection: Automatic: 0 (by default after PLL locked) Software: 1 (by sending 03 to INSTR CMDID see section 2 page 10) External: 2 or 3 (via START_IN_P and START_IN N SLVS pad)
SEL_CLOCK	0	0	Clock selection: Main @ 40 MHz (CLK_P and CLK_N SLVS pad): 0 ← Rescue @ 320 MHz[4] (CLKRESCUE_P CLKRESCUE_N SLVS pad): 1

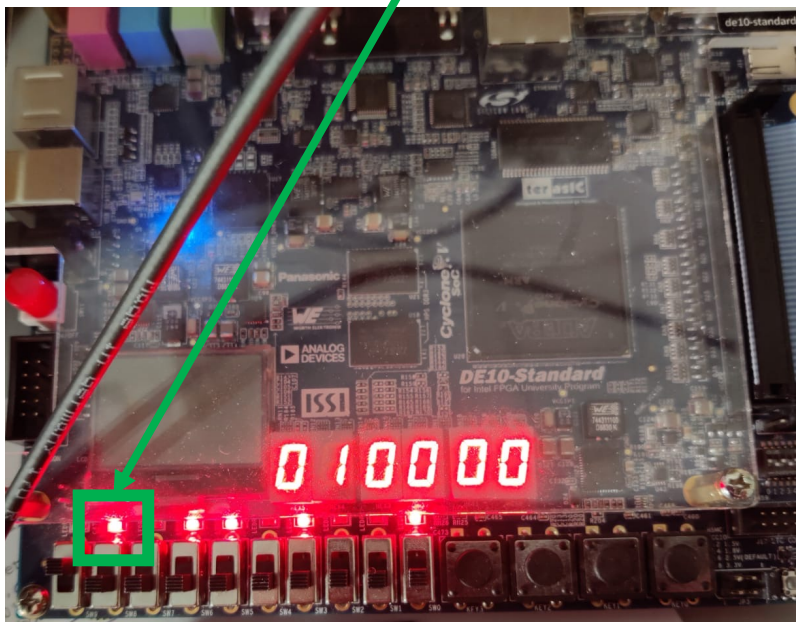
Name	Bits	Reset	Description
Not Used	7	0	
EN_ LOCKPLL	6	0	Enable LOCKPLL pad ←
EN_CLKPLL	5	0	Enable CLKPLL_P CLKPLL_N SLVS pad ←
TRIM_ REFCURPLL	4-2	4	Trimming bits for PLL current reference
DIS_PLL_LDO	1	0	Disable low-dropout PLL regulator (when disable VCO powered through AVDD and AVSS)
EN_PLL_VCO	0	1	Enable VCO

Name	Bits	Reset	Description
Not Used	7	0	
GAIN	6-3	7	Gain of current comparator to detect PLL lock from phase comparator
THRESHOLD	2-0	0	Threshold of current comparator to detect PLL lock from phase comparator

Scan to find the right value

Name	Bits	Reset	Description
EN_CLKRESCUE_TERM	7	0	Enable rescue clock (320 MHz) SLVS 100 Ω termination
EN_CLKSER	6	0	Enable output clock serializer (160 MHz) test option ←
EN_CUSTOM_FLV	5	0	Enable custom fill level for elastic buffer test option
EN_CUSTOM_BW	4	0	Enable custom bandwidth for frame generator test option
EN_PATTERN	3	0	Enable pattern for serializer test mode
EN_PULSEINJ	2	0	Enable external pulse injection test option
EN_PIXELMASK	1	1	Enable Pixel Masked (1) or Pulsed (0) mode for pixel array
EN_MFE	0	0	Enable Multi Frame Emulator test mode

Now the PLL is locked !!!



Data readout: PLL mode

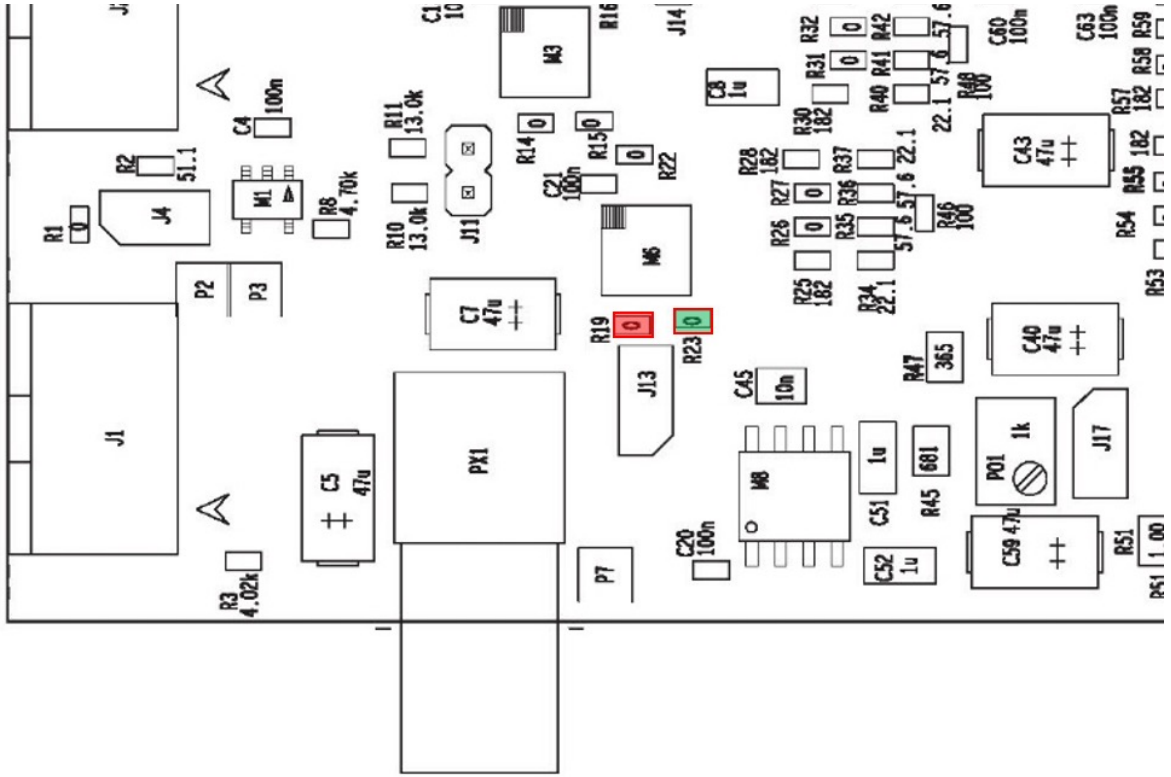
Same results as with the rescue clock:

- Only IDLE words when the chip is in STOP
- Frame when it is in START

```
47  fcaafcaafcaafcaafcaafcaafcaa
48  fcaafe2afe7ffe77fe7ffe2afe2afe2a
49  fe2afeaafcaafcaafcaafcaafcaafcaa
50  fec6fe06fe5cfe03fe00fe00fe00fe00
51  ff007ffdfc7dfceefceafcaafcaafcaa
52  fcaafcaafcaafcaafcaafceafcfffcea
53  fcaafcaafcaafcaafcaafcaafcaafcaa
54  fcaafcaafcaafcaafcaafcaafcaafcaa
55  fcaafcaafcaafcaafcaafcaafcaafcaa
56  fcaafcaafcaafcaafcaafcaafcaafcaa
57  fcaafcaafcaafcaafcaafcaafcaafcaa
58  fcaafcaafcaafcaafcaafcaafcaafcaa
59  fcaafcaafceafcaafcaafcaafcaafcaa
60  fcaafcaafcaafcaafcaafeaafe2afe3a
61  fe2afe2afeaafeaafeaafeaafeaafcaa
62  fec7fe06fe5cfe03fe00fe00fe00fe00
63  ff00577ffcaafceffcfffc55fcfffcaa
64  fcaafcaafcaafcaafcaafceafcfffcff
65  fcfffc55fc7ffceafcaafcaafcaafcaa
66  fcaafcaafcaafcaafcaafcaafceafcaa
67  fcaafcaafcaafcaafcaafcaafcaafcaa
68  fcaafcaafcaafcaafceffcaafcaafcaa
69  fcaafcaafcaafcaafcaafcaafcaafcaa
70  fcaafcaafcaafcaafcaafcaafcaafcaa
71  fcaafcaafcaafcaafcaafcaafcaafcaa
72  fcaafcaafcaafcaafcaafcaafcaafcaa
73  fcaafcaafeaafe2afe2afeaafe2afe2a
74  fe2afe2afeaafcaafcaafcaafcaafcaa
75  fec8fe06fe5cfe03fe00fe00fe00fe00
76  ff00775dfcaafcaafcaafcaafcaafcaa
77  fcaafcaafcaafceefcfffcfffceafcaa
78  fcaafcaafcaafcaafcaafcaafcaafcaa
```

Next Step…

Switch R19 into R23 to remove the external clock and try again to lock the PLL



- Clock @40 MHz is sent to the chip from the FPGA
- Clock OUT not used for the deserializer
- Same configuration of the registers

First attempt

Clock IN @320 MHz ⟶ it is wrong! But the PLL is locked. No IDLE words in the Spy Buffer

Second attempt

Clock IN @40 MHz ⟶ Right configuration

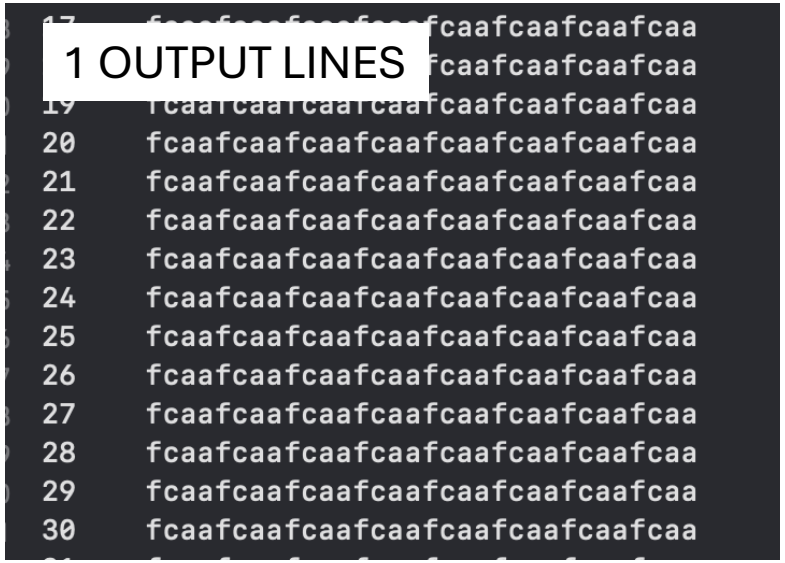
IDLE word decoded correctly ⟶ with 1 and 2 output lines

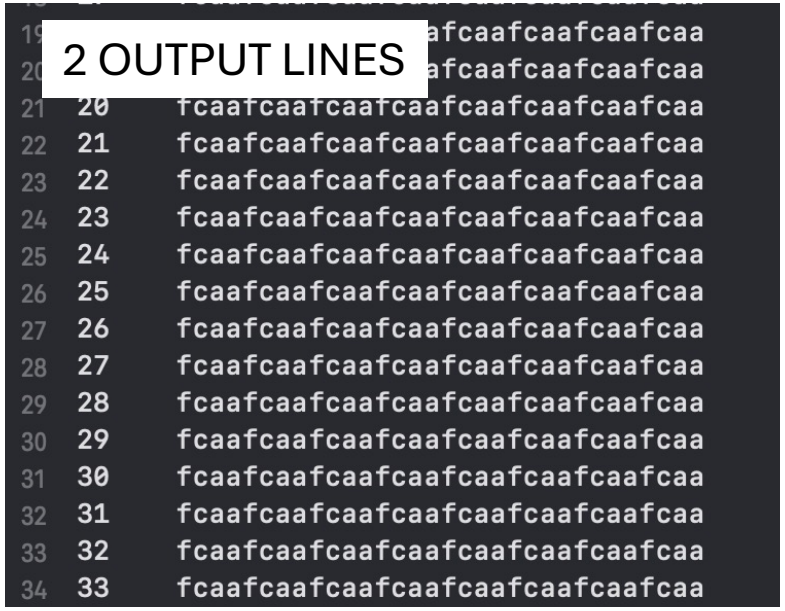
⟶ with 4, 2 or 3 output lines stop decoding the IDLE word

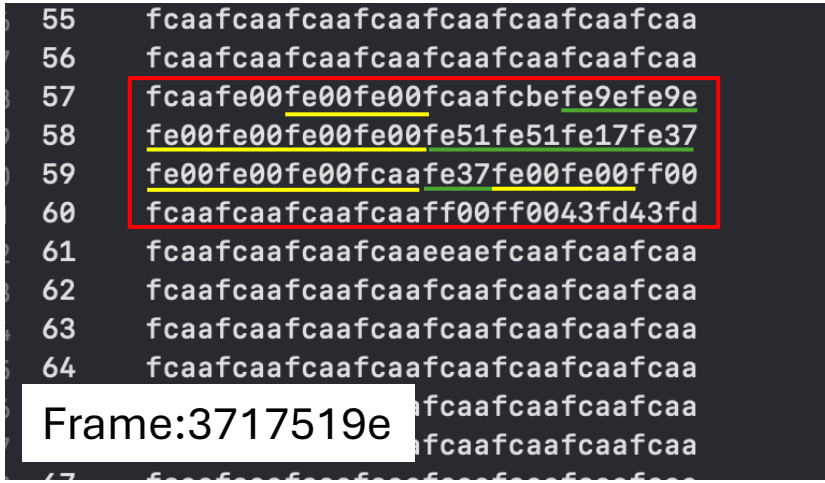
Chip in START ⟶ words seem duplicated but it's recognizable the frame

⟶ with 2 output lines, some problems in the deserializer, but it's still recognizable the frame

Some examples of outputs







…About ten lines of IDLE words…

…Many lines of IDLE words…