AGIPD Firmware Development

Igor Sheviakov
Qingqing Xia

12st Dec, 2015

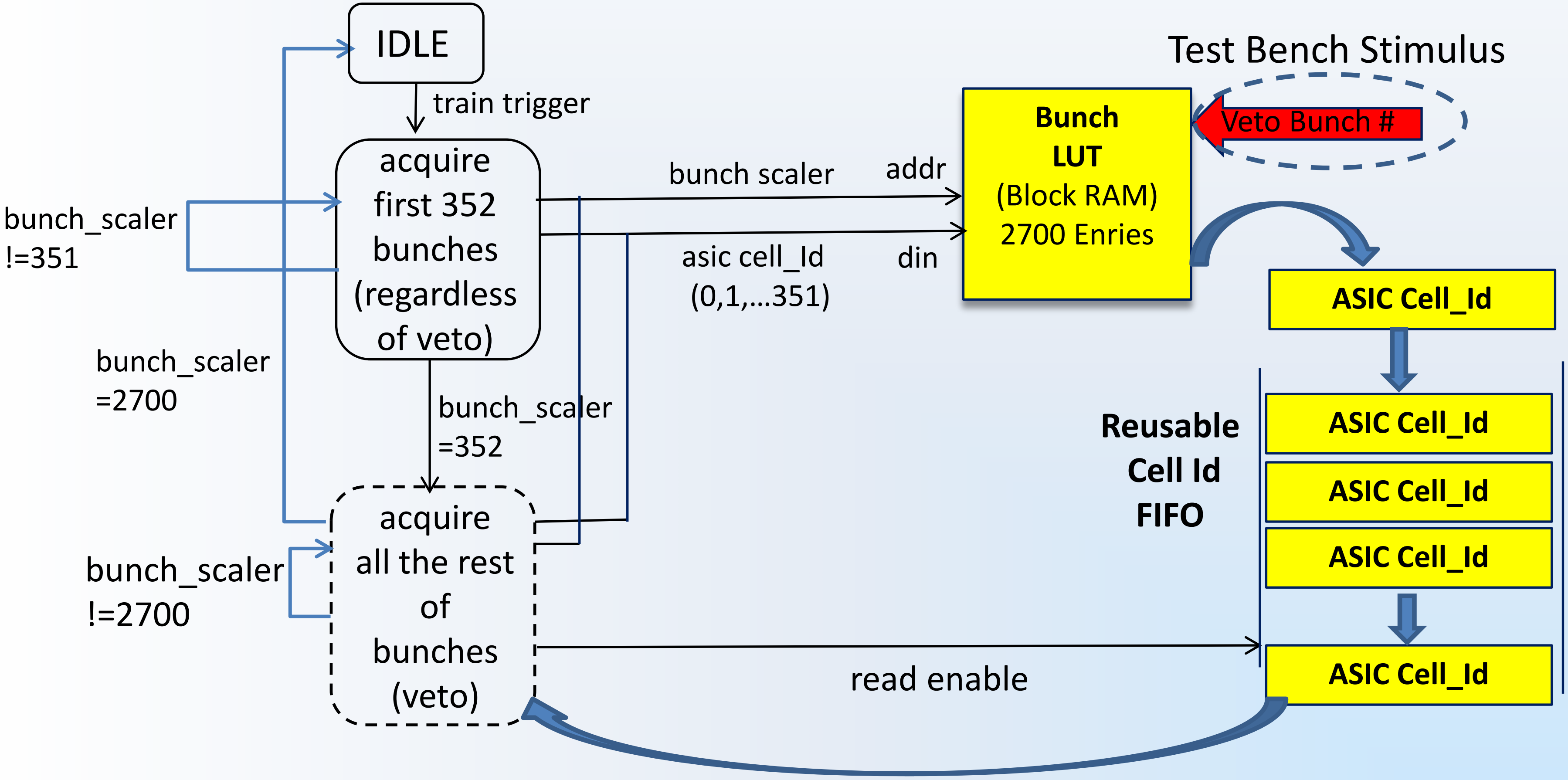
@DESY

Outline

- VETO Handling
  - Data flow diagram
- Train Builder format
  - Train header
  - Descriptor
  - Image Resorting
  - Detector Specific
  - Train trailer
- Multi-Module System
- Summary

Firmware Implementation: VETO Handling

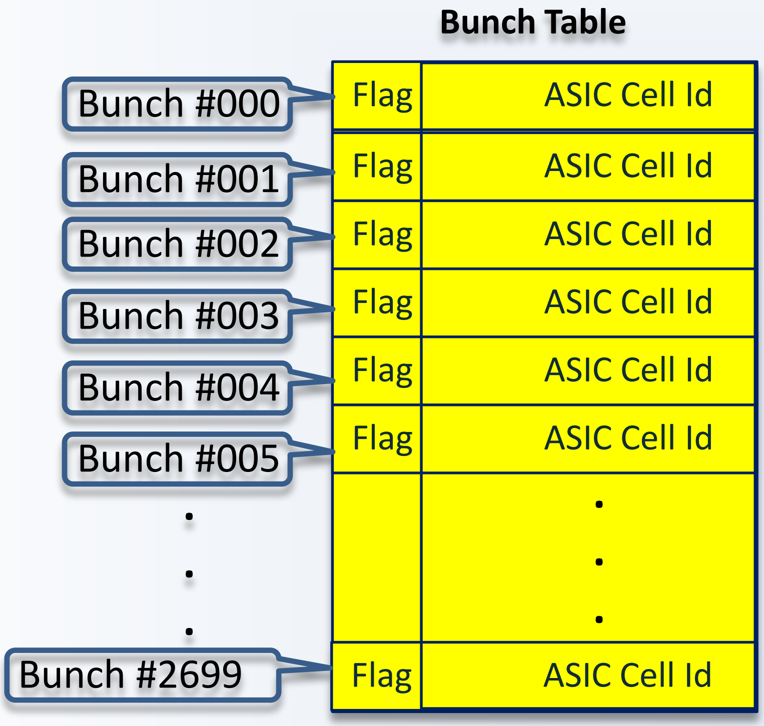
- Data flow diagram (preliminary & simplified)



Veto: is the bunch reject data encoded on the 99 MHz FEM clock.

Firmware Implementation: VETO Handling

- Generated bunch Table



Bunch Table

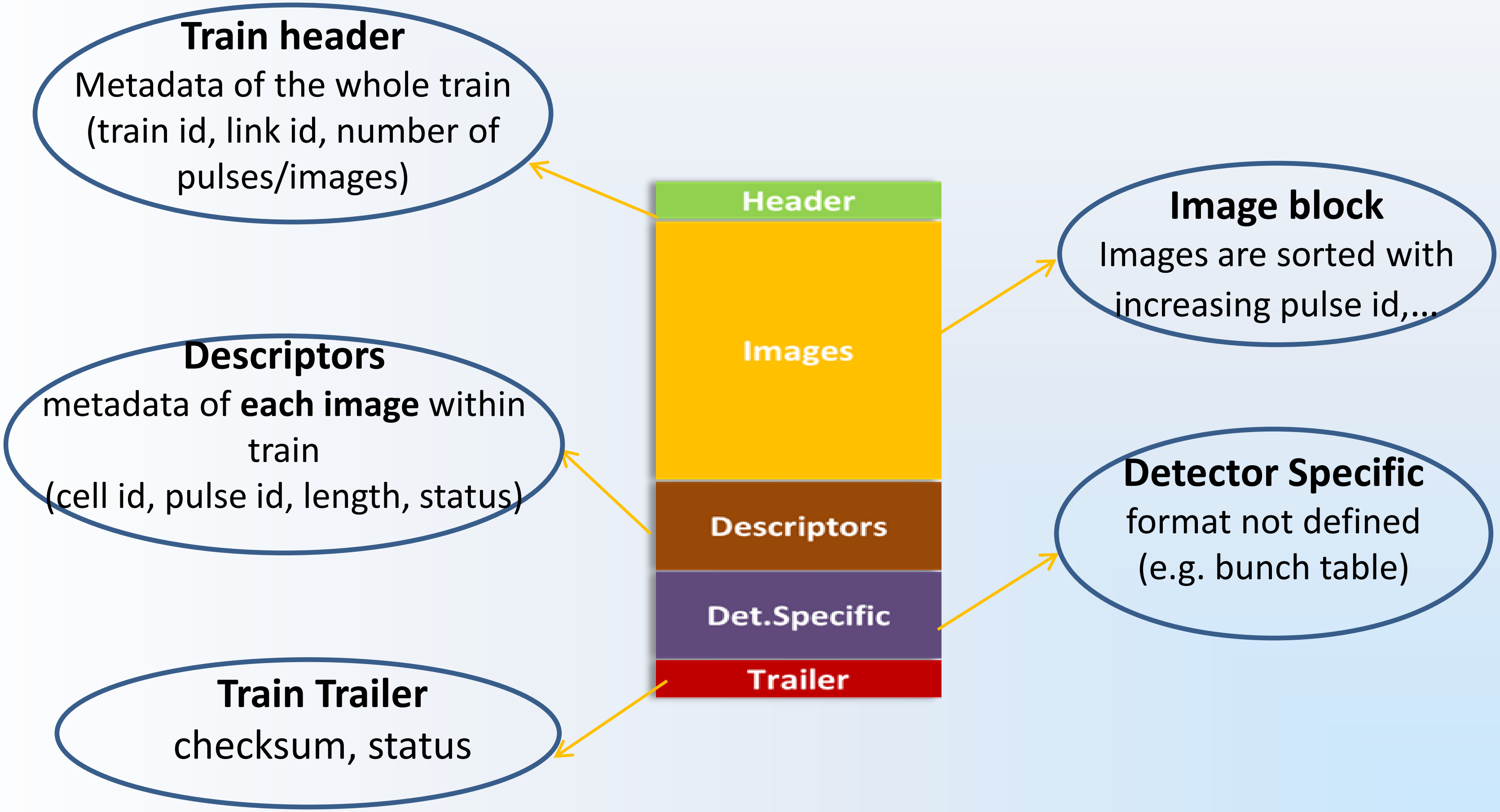
Bunch #000	Flag	ASIC Cell Id
Bunch #001	Flag	ASIC Cell Id
Bunch #002	Flag	ASIC Cell Id
Bunch #003	Flag	ASIC Cell Id
Bunch #004	Flag	ASIC Cell Id
Bunch #005	Flag	ASIC Cell Id
. . .		. . .
Bunch #2699	Flag	ASIC Cell Id

Flag=1 VETO(bad)
Flag=0 NO VETO(good)

e.g
0 x 8000→ bunch # 0: bad
0 x 0001→ bunch # 1: good

Firmware Implementation: Train Builder(TB) Format

XFEL Train Format Overview



Header

Images

Descriptors

Det.Specific

Trailer

Train header
Metadata of the whole train
(train id, link id, number of pulses/images)

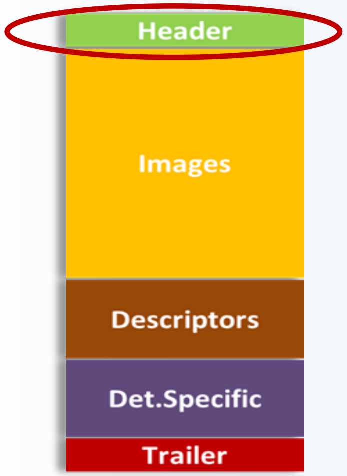
Descriptors
metadata of **each image** within train
(cell id, pulse id, length, status)

Train Trailer
checksum, status

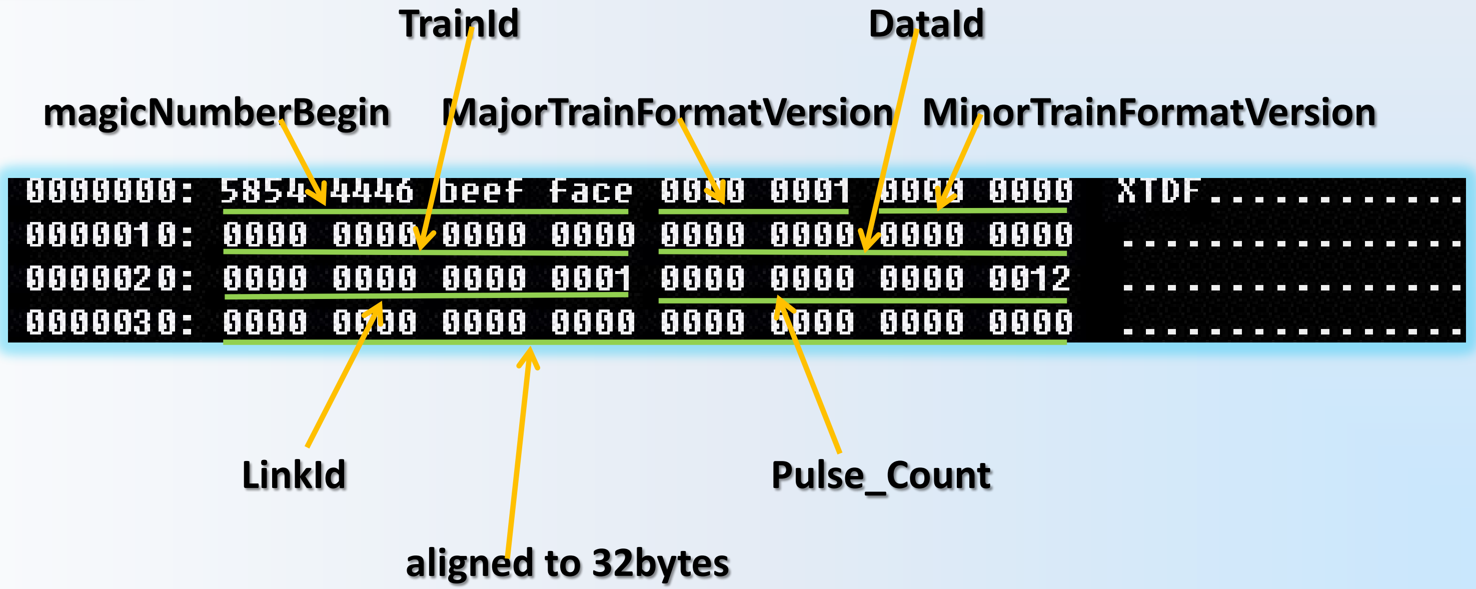
Image block
Images are sorted with increasing pulse id,...

Detector Specific
format not defined
(e.g. bunch table)

Firmware Implementation: Train Builder(TB) Format



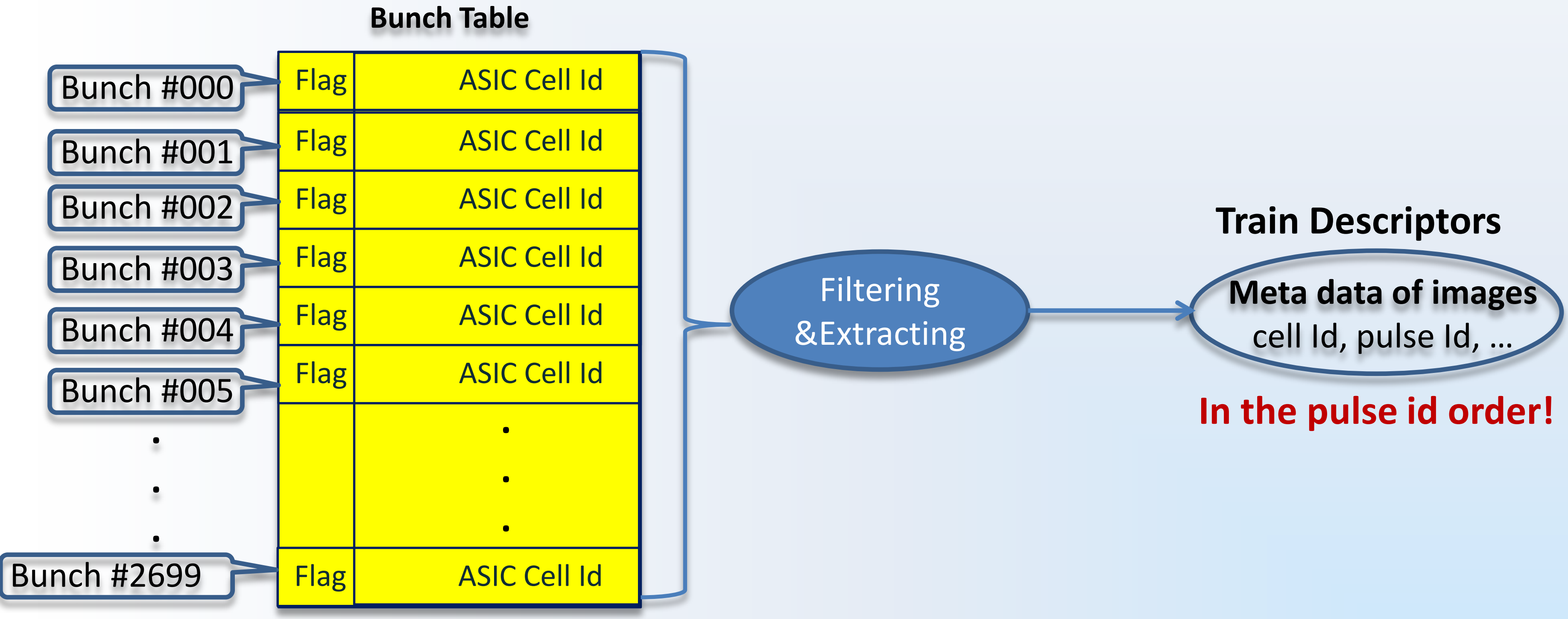
- Size=64Bytes
- Train Id now is a internal counter, later should be synchronized with C&C system



```
0000000: 5854 4446 beef face 0000 0001 0000 0000  XTDF............
0000010: 0000 0000 0000 0000 0000 0000 0000 0000  ................
0000020: 0000 0000 0000 0001 0000 0000 0000 0012  ................
0000030: 0000 0000 0000 0000 0000 0000 0000 0000  ................
```

Firmware Implementation:
VETO Handling

- Extracting Information



Bunch Table

Bunch #000 Flag ASIC Cell Id

Bunch #001 Flag ASIC Cell Id

Bunch #002 Flag ASIC Cell Id

Bunch #003 Flag ASIC Cell Id

Bunch #004 Flag ASIC Cell Id

Bunch #005 Flag ASIC Cell Id

. . .

Bunch #2699 Flag ASIC Cell Id

Filtering &Extracting

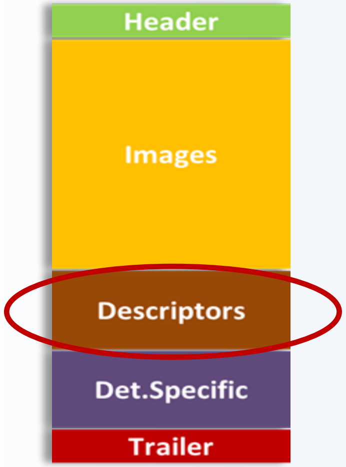
Train Descriptors

Meta data of images
cell Id, pulse Id, …

In the pulse id order!

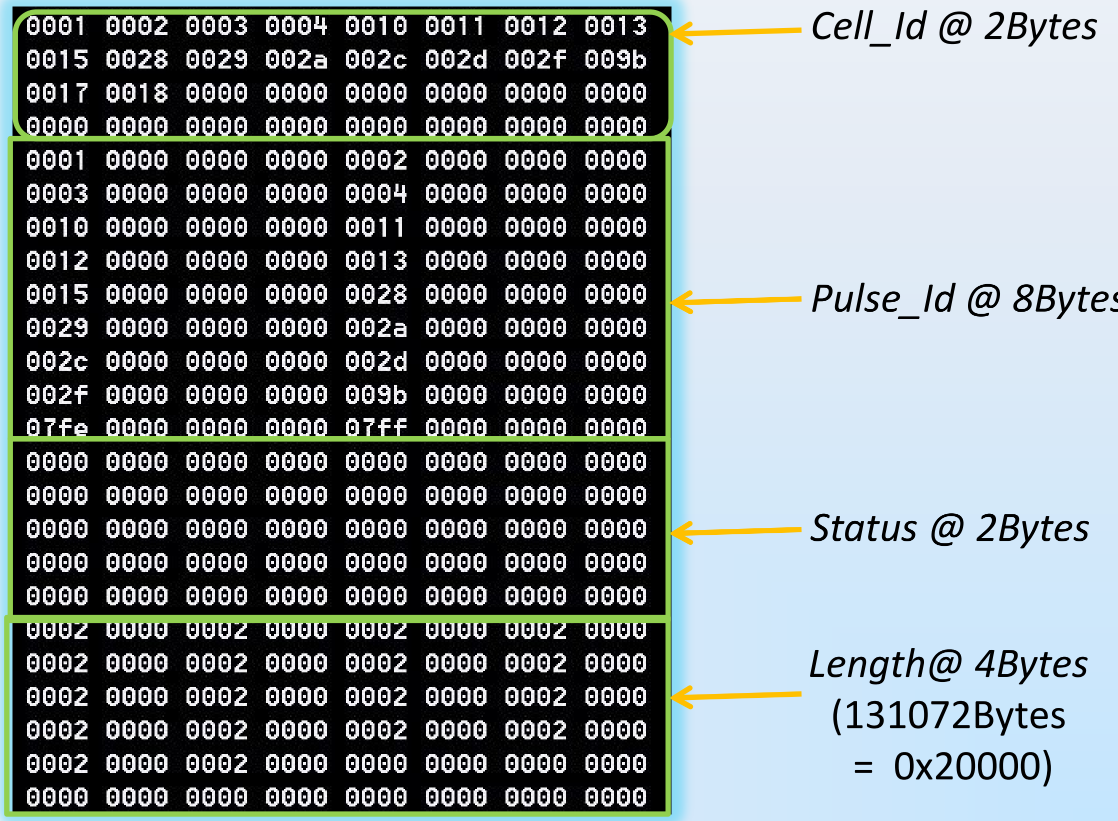
Flag=1 → VETO(bad)
Flag=0 → NO VETO(good)

Firmware Implementation:
Train Builder(TB) Format



- 4 Descriptors: Cell id & Pulse id & Status & Length
- Size: Each descriptor block aligned with 32 bytes (padding with zeros)
- Example

Example:Pulse_COUNT=18



```
0001 0002 0003 0004 0010 0011 0012 0013
0015 0028 0029 002a 002c 002d 002f 009b
0017 0018 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0001 0000 0000 0000 0002 0000 0000 0000
0003 0000 0000 0000 0004 0000 0000 0000
0010 0000 0000 0000 0011 0000 0000 0000
0012 0000 0000 0000 0013 0000 0000 0000
0015 0000 0000 0000 0028 0000 0000 0000
0029 0000 0000 0000 002a 0000 0000 0000
002c 0000 0000 0000 002d 0000 0000 0000
002f 0000 0000 0000 009b 0000 0000 0000
07fe 0000 0000 0000 07ff 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0002 0000 0002 0000 0002 0000 0002 0000
0002 0000 0002 0000 0002 0000 0002 0000
0002 0000 0002 0000 0002 0000 0002 0000
0002 0000 0002 0000 0002 0000 0002 0000
0002 0000 0002 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
```

Cell_Id @ 2Bytes

Pulse_Id @ 8Bytes

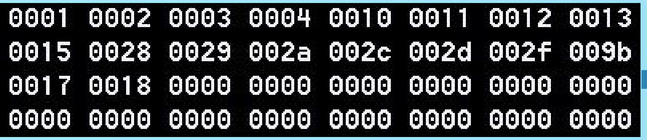
Status @ 2Bytes

Length@ 4Bytes
(131072Bytes
= 0x20000)

Firmware Implementation: Train Builder(TB) Format

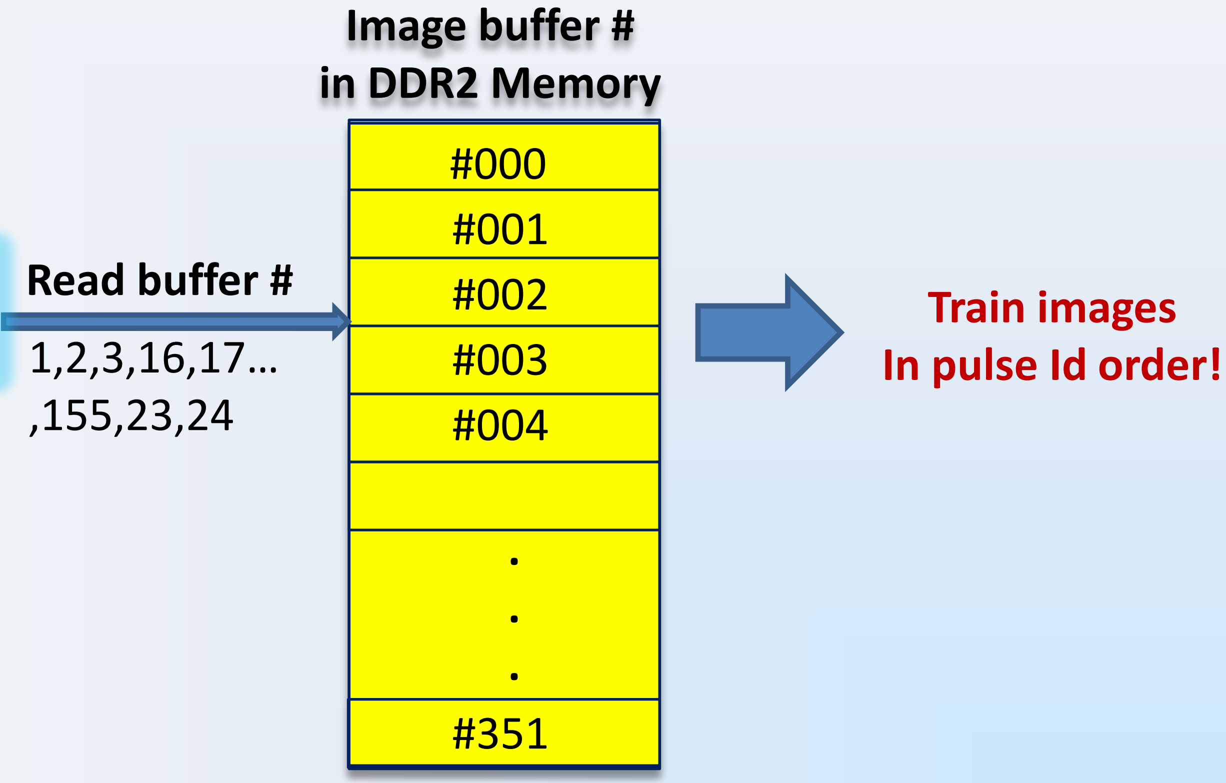
- Train images sorting

Descriptors:
Cell Id #



```
0001 0002 0003 0004 0010 0011 0012 0013
0015 0028 0029 002a 002c 002d 002f 009b
0017 0018 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
```

In pulse Id order!

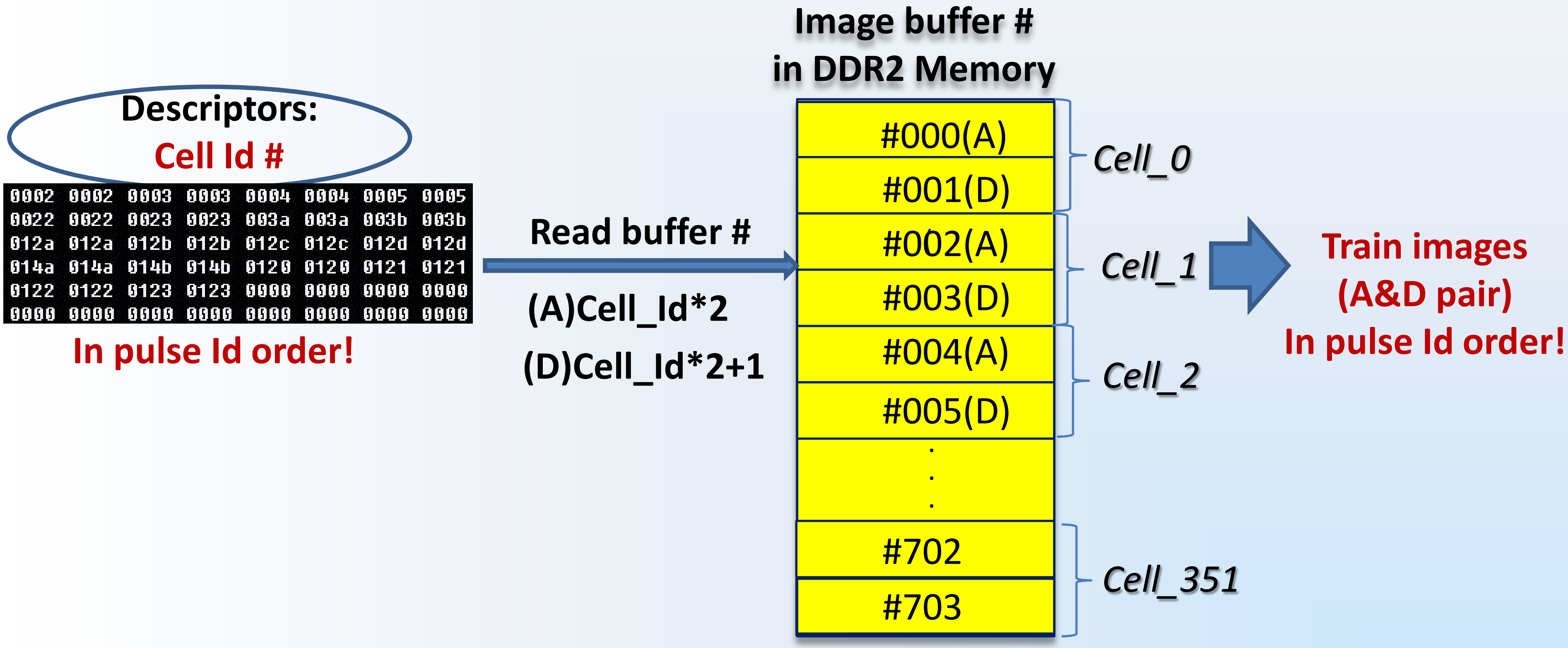


Train images
In pulse Id order!

Firmware Implementation:

Train Builder(TB) Format

- Train images sorting(separate frames of A,D,A,D,A,D…)



Descriptors:

Cell Id #

```
0002 0002 0003 0003 0004 0004 0005 0005
0022 0022 0023 0023 003a 003a 003b 003b
012a 012a 012b 012b 012c 012c 012d 012d
014a 014a 014b 014b 0120 0120 0121 0121
0122 0122 0123 0123 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
```

In pulse Id order!

Read buffer #

(A)Cell_Id*2

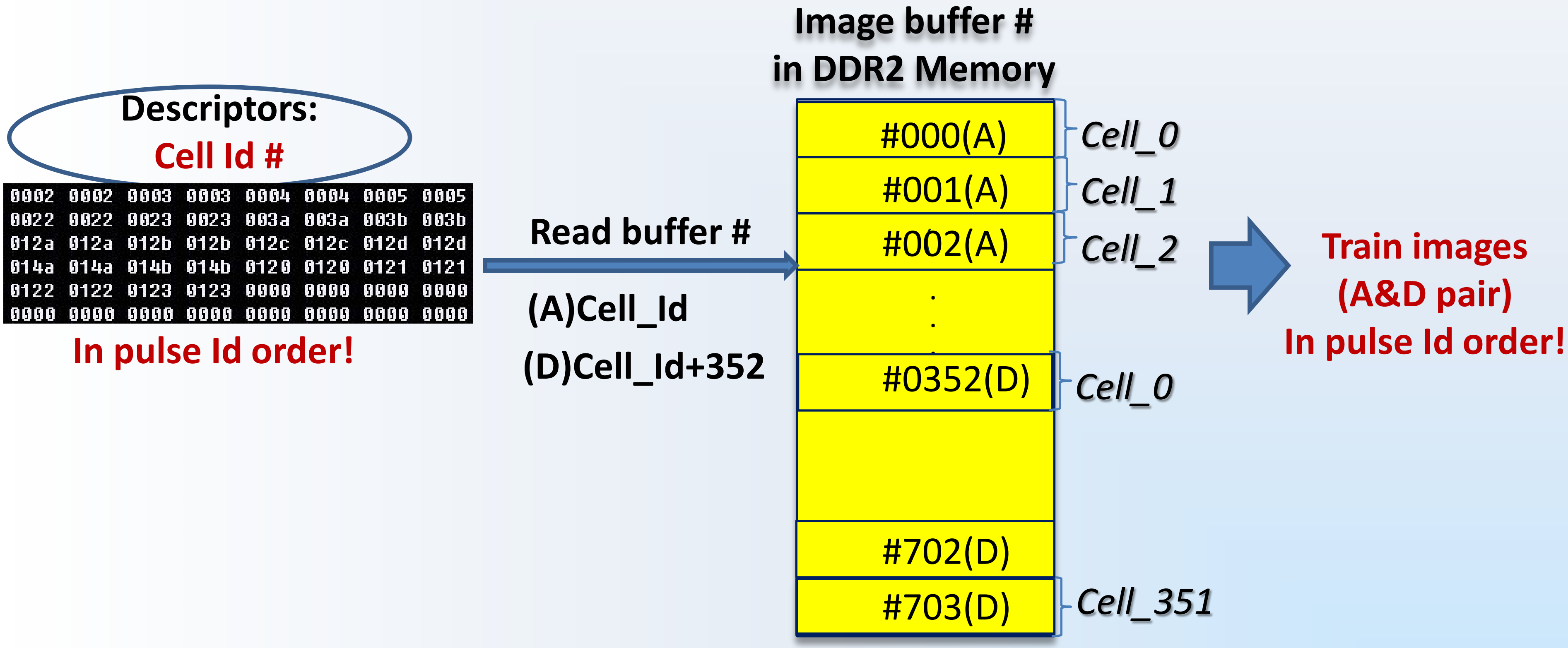
(D)Cell_Id*2+1

Image buffer #
in DDR2 Memory

Train images
(A&D pair)
In pulse Id order!

Firmware Implementation:
Train Builder(TB) Format

- Train images sorting(separate frames : all A, all D)



Descriptors:
Cell Id #

```
0002 0002 0003 0003 0004 0004 0005 0005
0022 0022 0023 0023 003a 003a 003b 003b
012a 012a 012b 012b 012c 012c 012d 012d
014a 014a 014b 014b 0120 0120 0121 0121
0122 0122 0123 0123 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
```

In pulse Id order!

Read buffer #

(A)Cell_Id

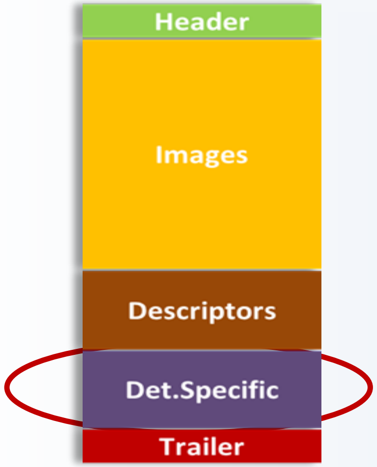
(D)Cell_Id+352

Image buffer #
in DDR2 Memory

Train images
(A&D pair)
In pulse Id order!

Firmware Implementation:

Train Builder(TB) Format



Header

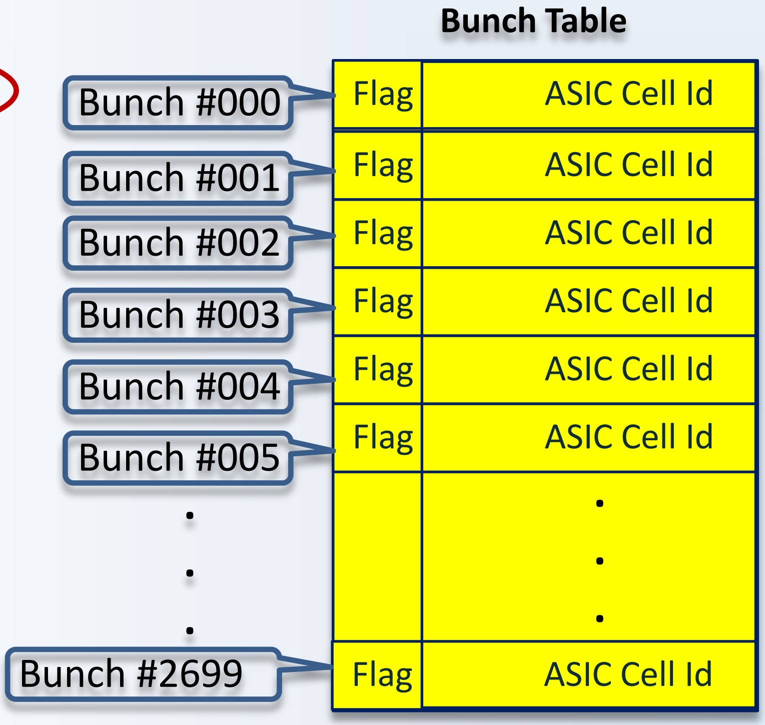
Images

Descriptors

Det.Specific

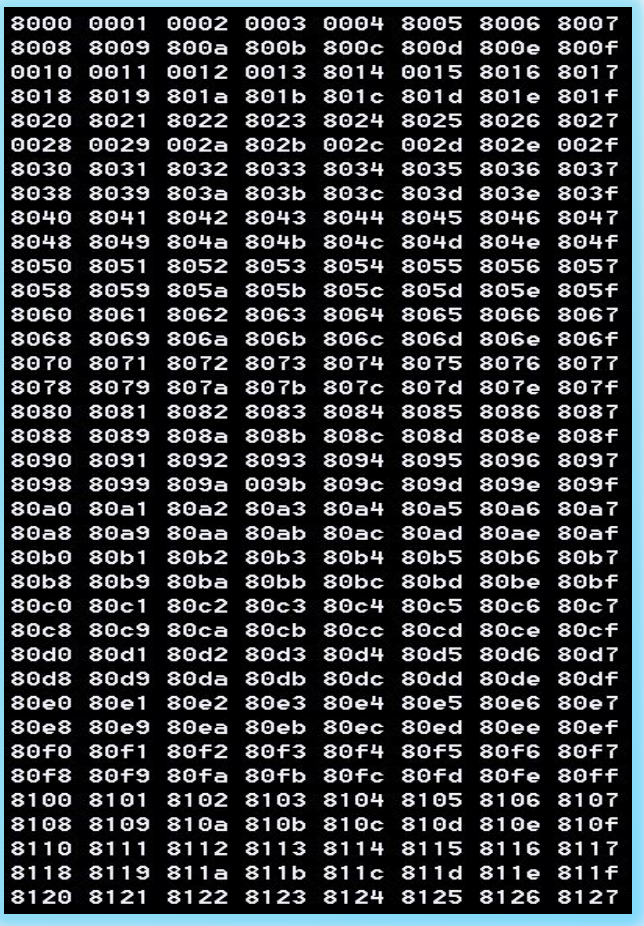
Trailer

- Up to Detector
- Bunch table
- other ideas?



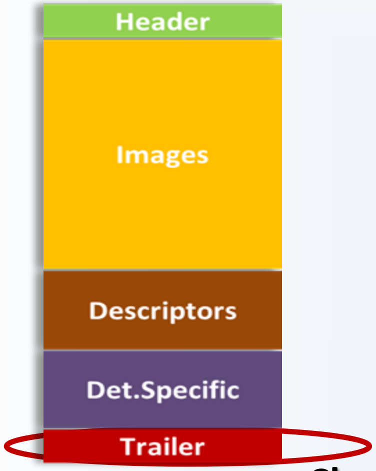
Bunch Table

Bunch #000	Flag	ASIC Cell Id
Bunch #001	Flag	ASIC Cell Id
Bunch #002	Flag	ASIC Cell Id
Bunch #003	Flag	ASIC Cell Id
Bunch #004	Flag	ASIC Cell Id
Bunch #005	Flag	ASIC Cell Id
. . .		. . .
Bunch #2699	Flag	ASIC Cell Id

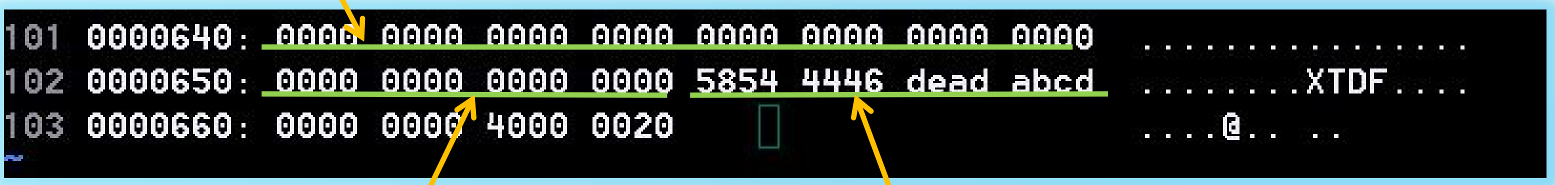


```
8000 0001 0002 0003 0004 8005 8006 8007
8008 8009 800a 800b 800c 800d 800e 800f
0010 0011 0012 0013 8014 0015 8016 8017
8018 8019 801a 801b 801c 801d 801e 801f
8020 8021 8022 8023 8024 8025 8026 8027
0028 0029 002a 802b 002c 002d 802e 002f
8030 8031 8032 8033 8034 8035 8036 8037
8038 8039 803a 803b 803c 803d 803e 803f
8040 8041 8042 8043 8044 8045 8046 8047
8048 8049 804a 804b 804c 804d 804e 804f
8050 8051 8052 8053 8054 8055 8056 8057
8058 8059 805a 805b 805c 805d 805e 805f
8060 8061 8062 8063 8064 8065 8066 8067
8068 8069 806a 806b 806c 806d 806e 806f
8070 8071 8072 8073 8074 8075 8076 8077
8078 8079 807a 807b 807c 807d 807e 807f
8080 8081 8082 8083 8084 8085 8086 8087
8088 8089 808a 808b 808c 808d 808e 808f
8090 8091 8092 8093 8094 8095 8096 8097
8098 8099 809a 009b 809c 809d 809e 809f
80a0 80a1 80a2 80a3 80a4 80a5 80a6 80a7
80a8 80a9 80aa 80ab 80ac 80ad 80ae 80af
80b0 80b1 80b2 80b3 80b4 80b5 80b6 80b7
80b8 80b9 80ba 80bb 80bc 80bd 80be 80bf
80c0 80c1 80c2 80c3 80c4 80c5 80c6 80c7
80c8 80c9 80ca 80cb 80cc 80cd 80ce 80cf
80d0 80d1 80d2 80d3 80d4 80d5 80d6 80d7
80d8 80d9 80da 80db 80dc 80dd 80de 80df
80e0 80e1 80e2 80e3 80e4 80e5 80e6 80e7
80e8 80e9 80ea 80eb 80ec 80ed 80ee 80ef
80f0 80f1 80f2 80f3 80f4 80f5 80f6 80f7
80f8 80f9 80fa 80fb 80fc 80fd 80fe 80ff
8100 8101 8102 8103 8104 8105 8106 8107
8108 8109 810a 810b 810c 810d 810e 810f
8110 8111 8112 8113 8114 8115 8116 8117
8118 8119 811a 811b 811c 811d 811e 811f
8120 8121 8122 8123 8124 8125 8126 8127
```

Firmware Implementation:
Train Builder(TB) Format



- Size=32 bytes
- Checksum(Not clear yet)



```
0000640: 0000 0000 0000 0000 0000 0000 0000 0000  ................
0000650: 0000 0000 0000 0000 5854 4446 dead abcd  ........XTDF....
0000660: 0000 0000 4000 0020                      ....@.. ..
```

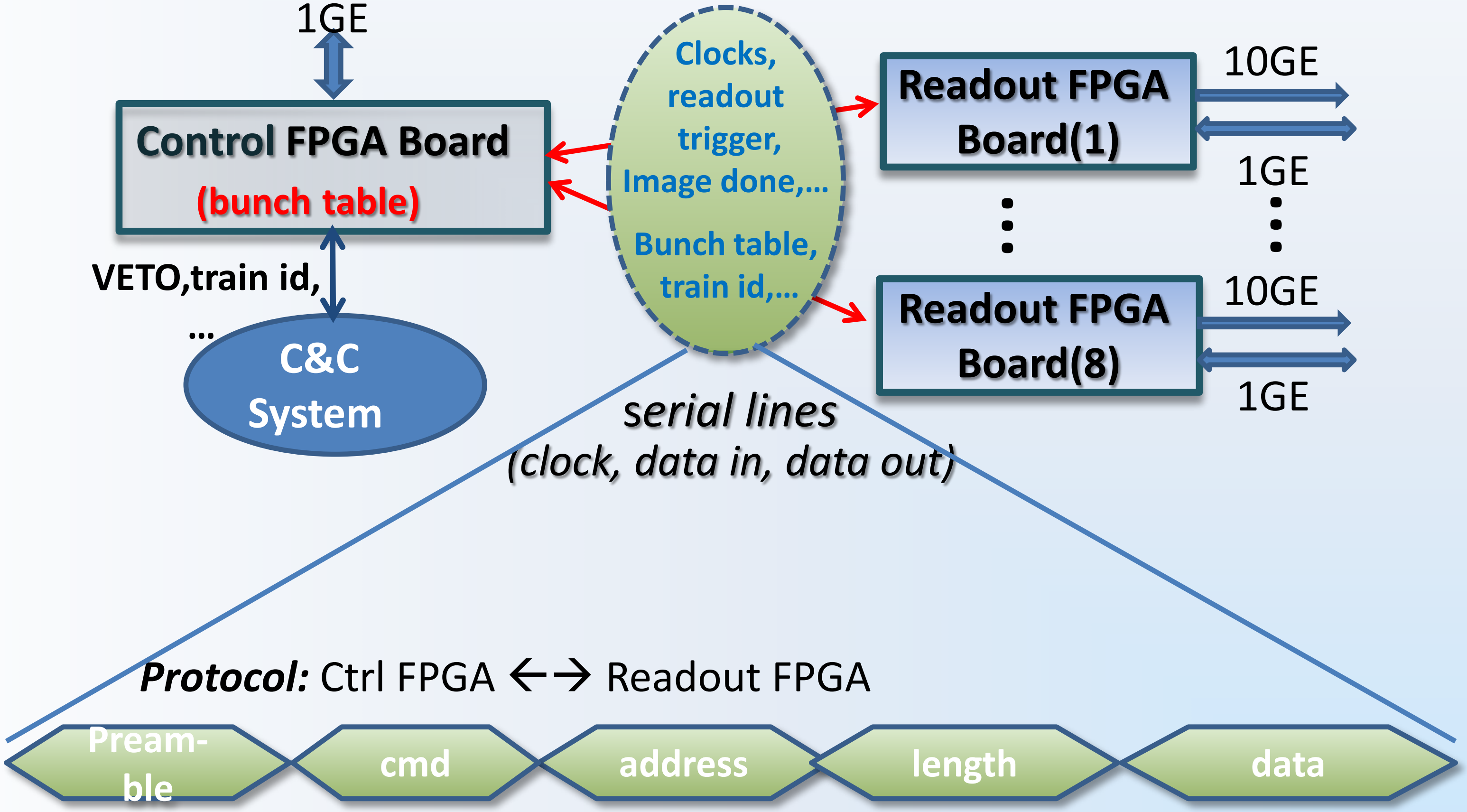
Checksum

Status

magicNumberEnd

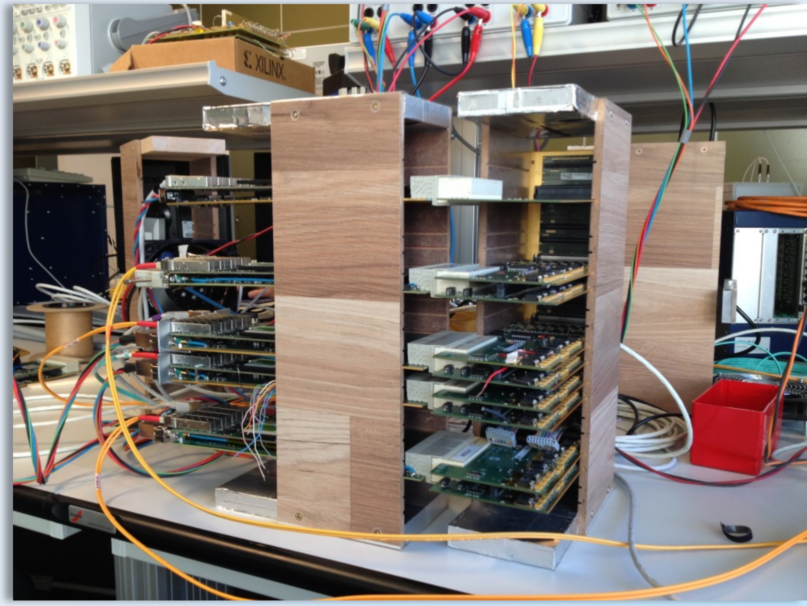
Multi-Module System

- Requirements



Multi-Module System

- Status
  - ❑ First Release of Ctrl FW, Readout FW & SW
  - ❑ Prototype setup (Ctrl FPGA ←→2x Readout) in two labs(FEA, FS-DS)
  - ❑ Tested max. number of multi-modules : 4x Readout *(hardware availability)*
    - ❑ All 4 simultaneously working modules behavior as expect
    - ❑ 10GE MAC&IP addresses configurable by user SW
    - ❑ No ASICs Module attached yet: More tests!



Summary

- Status
  - ❑ Pixel descrambling ✓
  - ❑ ADC delay automatic adjustment✓
  - ❑ Single Module 10GE integration(FW&Tango) ✓
  - ❑ 10GE train data in XFEL Train builder format ✓
  - ❑ Train **Descriptor & Specific** filled/generated with bunch table✓
  - ❑ 10GE Train images can be sorted in increasing **Puls Id** order✓
  - ❑ Multi-module system: Control FPGA←→ 4x Readout FPGA ✓
- Left to do
  - ❑ Test, Test, Test…
    - ❑ Multi-module system: ASIC Modules, Complete 8xReadout,
    - ❑ XFEL Train builder &backend PC
  - ❑ C&C System Integration →XFEL Timing adaption
  - ❑ Software Integration

Thank You!

Spare Slides

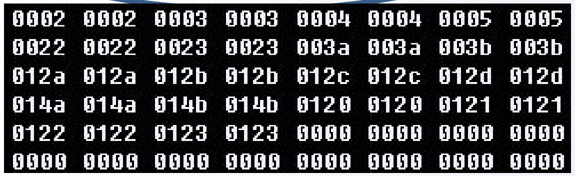
Firmware Implementation:

Train Builder(TB) Format

- Train images sorting(separate frames : all A, all D)

Descriptors:

Cell Id #



```
0002 0002 0003 0003 0004 0004 0005 0005
0022 0022 0023 0023 003a 003a 003b 003b
012a 012a 012b 012b 012c 012c 012d 012d
014a 014a 014b 014b 0120 0120 0121 0121
0122 0122 0123 0123 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
```

In pulse Id order!

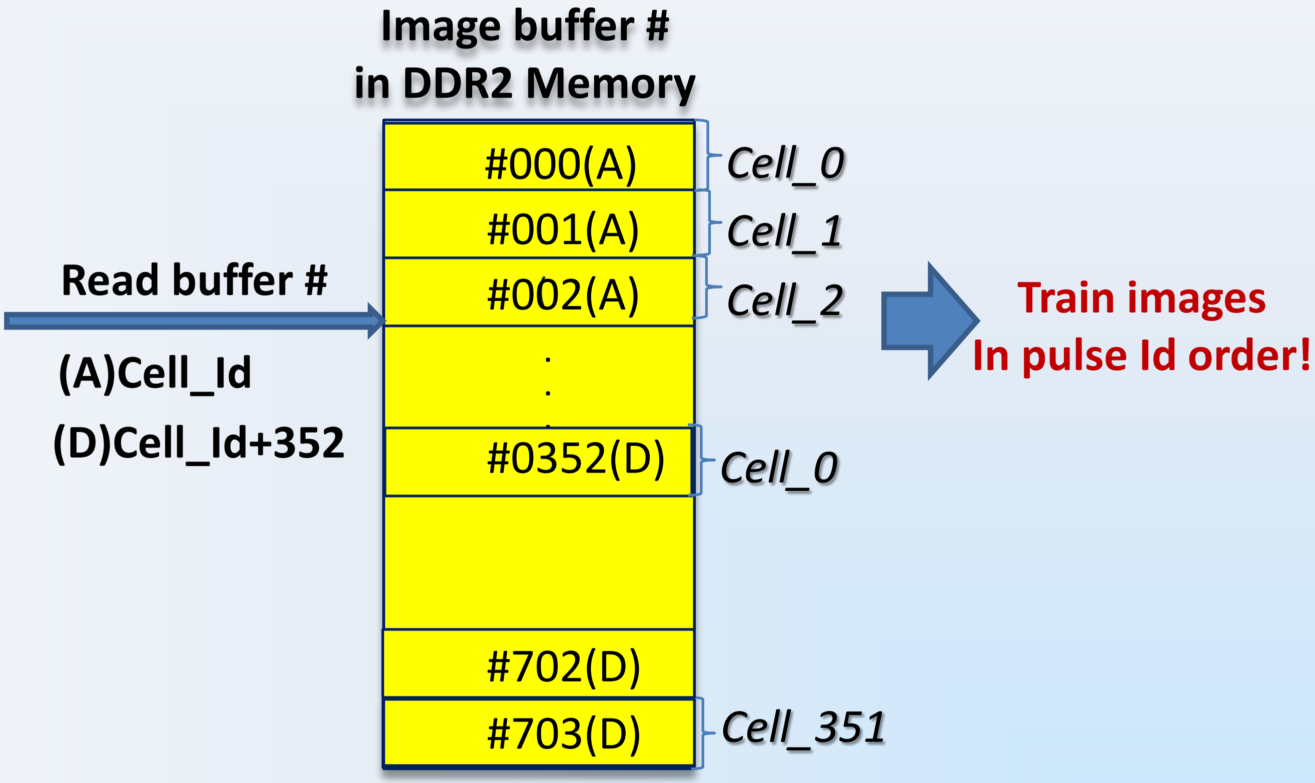


Image buffer #
in DDR2 Memory

Read buffer #

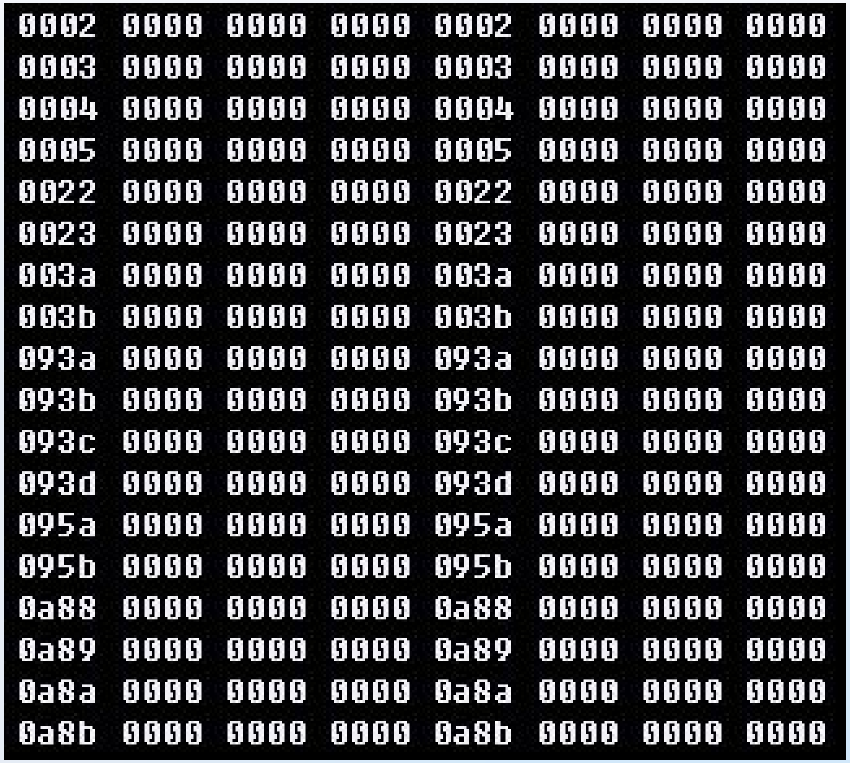
(A)Cell_Id

(D)Cell_Id+352

Train images
In pulse Id order!

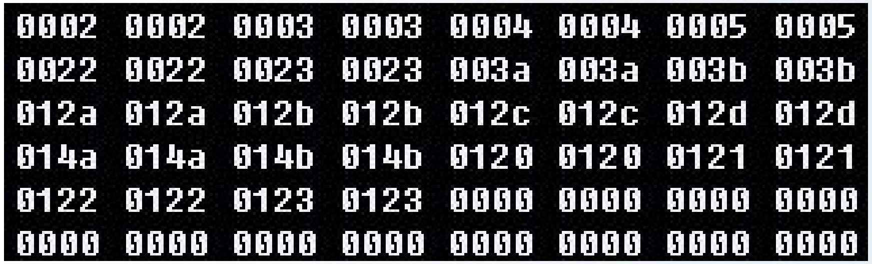
AGIPD Firmware Development

Pulse Id



```
0002 0000 0000 0000 0002 0000 0000 0000
0003 0000 0000 0000 0003 0000 0000 0000
0004 0000 0000 0000 0004 0000 0000 0000
0005 0000 0000 0000 0005 0000 0000 0000
0022 0000 0000 0000 0022 0000 0000 0000
0023 0000 0000 0000 0023 0000 0000 0000
003a 0000 0000 0000 003a 0000 0000 0000
003b 0000 0000 0000 003b 0000 0000 0000
093a 0000 0000 0000 093a 0000 0000 0000
093b 0000 0000 0000 093b 0000 0000 0000
093c 0000 0000 0000 093c 0000 0000 0000
093d 0000 0000 0000 093d 0000 0000 0000
095a 0000 0000 0000 095a 0000 0000 0000
095b 0000 0000 0000 095b 0000 0000 0000
0a88 0000 0000 0000 0a88 0000 0000 0000
0a89 0000 0000 0000 0a89 0000 0000 0000
0a8a 0000 0000 0000 0a8a 0000 0000 0000
0a8b 0000 0000 0000 0a8b 0000 0000 0000
```

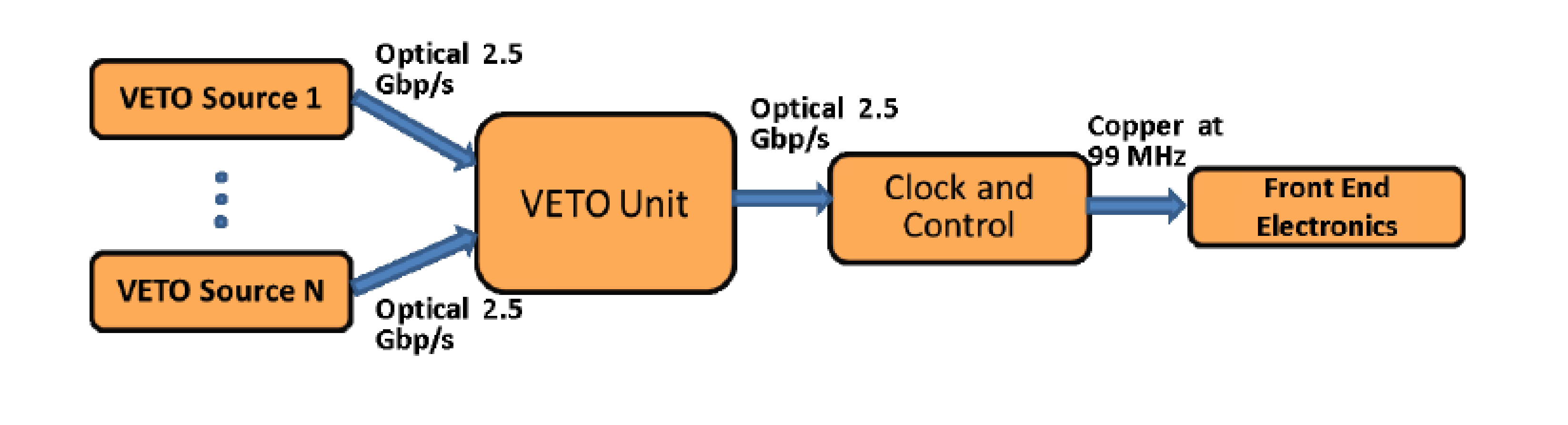
Cell Id

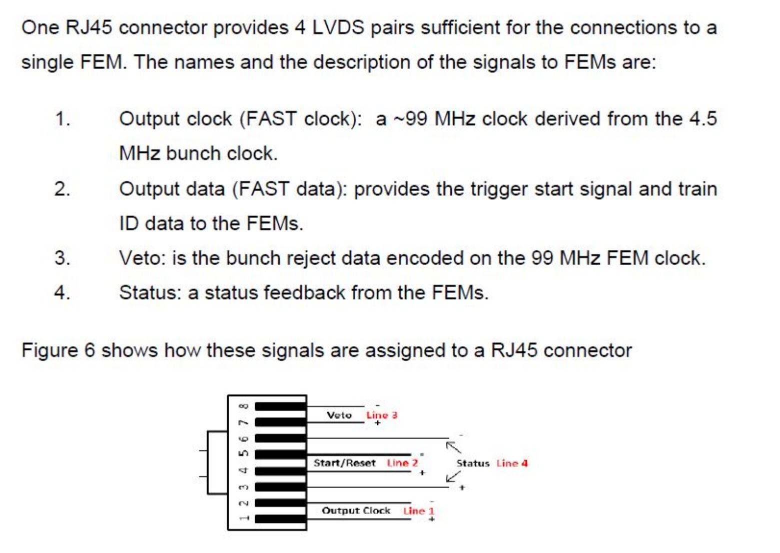


```
0002 0002 0003 0003 0004 0004 0005 0005
0022 0022 0023 0023 003a 003a 003b 003b
012a 012a 012b 012b 012c 012c 012d 012d
014a 014a 014b 014b 0120 0120 0121 0121
0122 0122 0123 0123 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
```

Firmware Implementation: VETO Handling

Veto: is the bunch reject data encoded on the 99 MHz FEM clock.





One RJ45 connector provides 4 LVDS pairs sufficient for the connections to a single FEM. The names and the description of the signals to FEMs are:

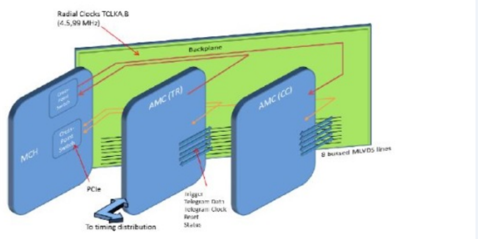
1. Output clock (FAST clock): a ~99 MHz clock derived from the 4.5 MHz bunch clock.
2. Output data (FAST data): provides the trigger start signal and train ID data to the FEMs.
3. Veto: is the bunch reject data encoded on the 99 MHz FEM clock.
4. Status: a status feedback from the FEMs.

Figure 6 shows how these signals are assigned to a RJ45 connector

Firmware Implementation:

VETO Handling

Veto: is the bunch reject data encoded on the 99 MHz FEM clock.



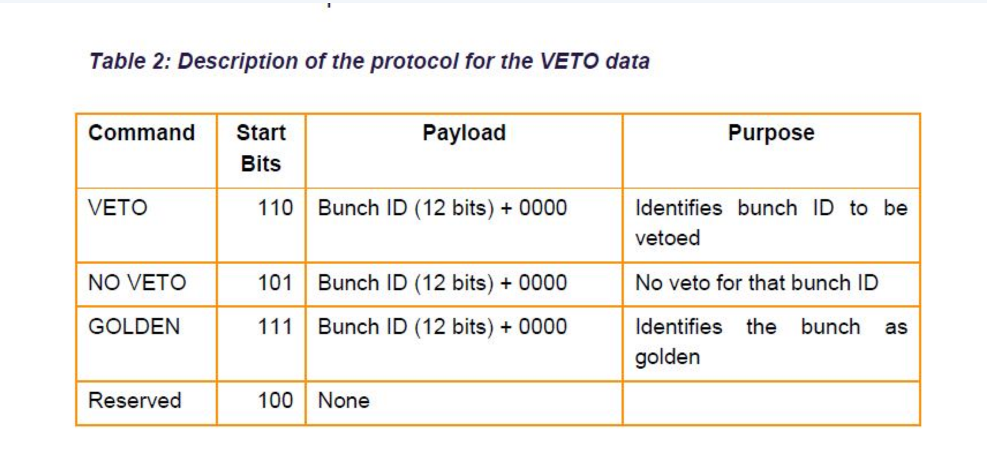


Table 2: Description of the protocol for the VETO data

Command	Start Bits	Payload	Purpose
VETO	110	Bunch ID (12 bits) + 0000	Identifies bunch ID to be vetoed
NO VETO	101	Bunch ID (12 bits) + 0000	No veto for that bunch ID
GOLDEN	111	Bunch ID (12 bits) + 0000	Identifies the bunch as golden
Reserved	100	None	

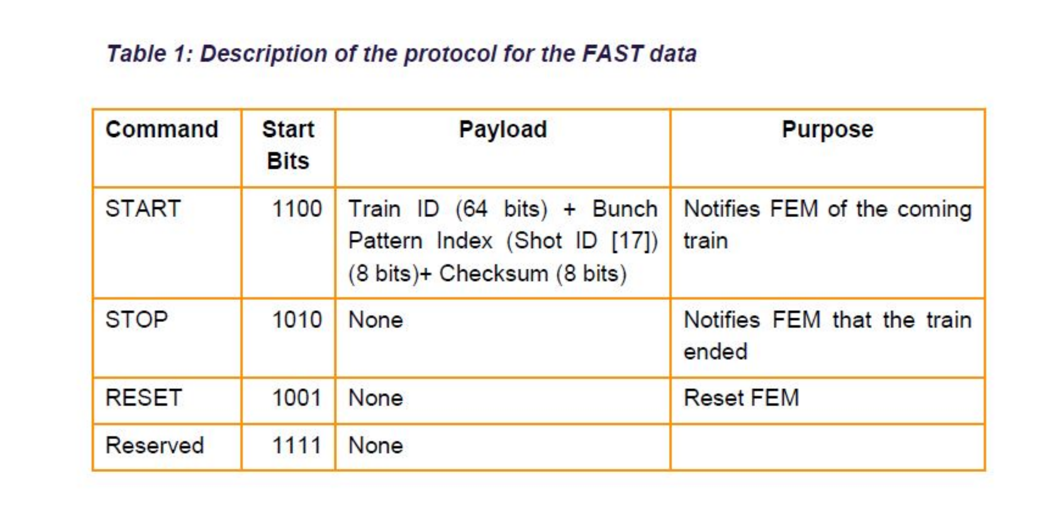
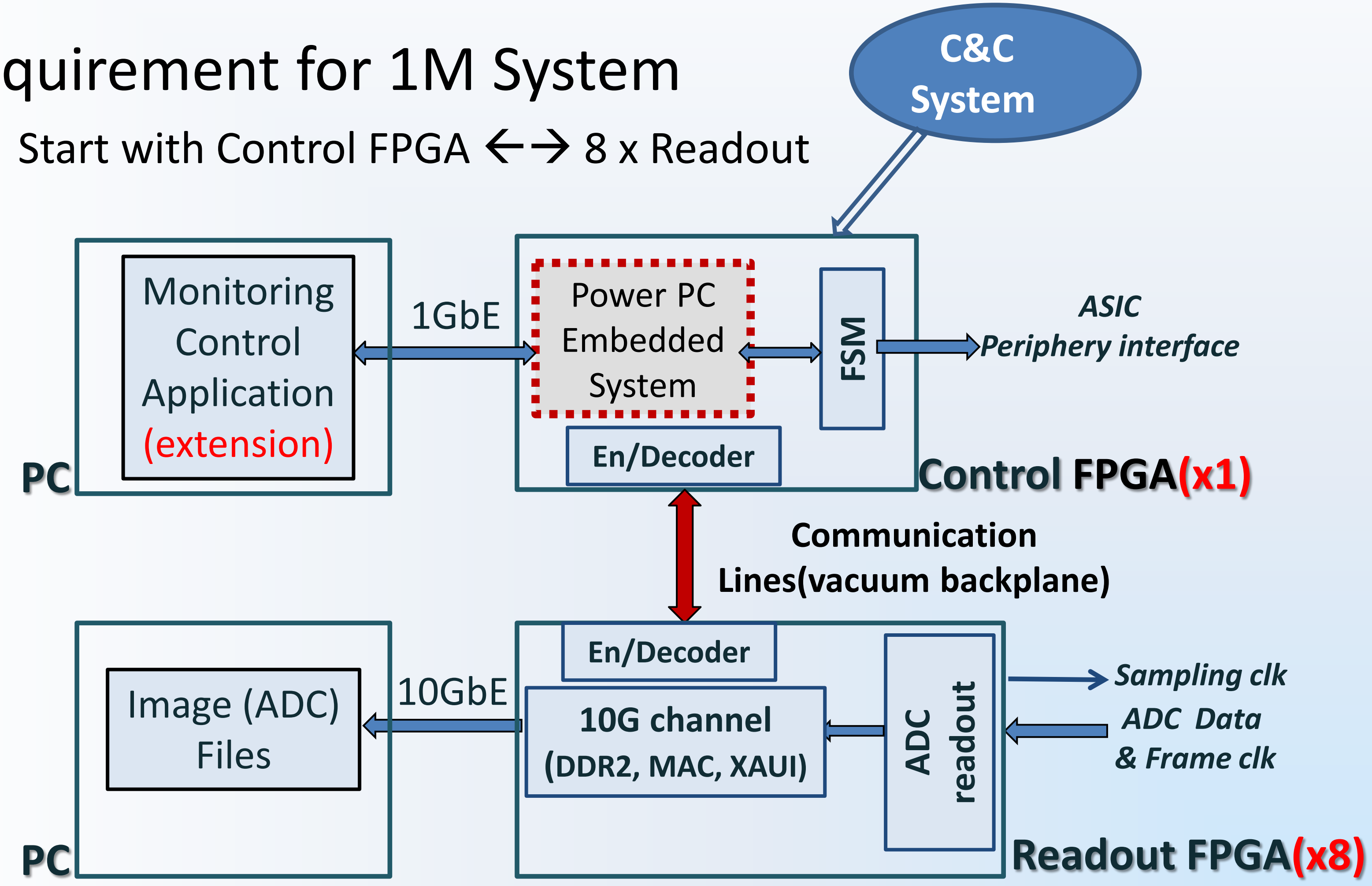


Table 1: Description of the protocol for the FAST data

Command	Start Bits	Payload	Purpose
START	1100	Train ID (64 bits) + Bunch Pattern Index (Shot ID [17]) (8 bits)+ Checksum (8 bits)	Notifies FEM of the coming train
STOP	1010	None	Notifies FEM that the train ended
RESET	1001	None	Reset FEM
Reserved	1111	None	

Firmware Development for 1M System

- Requirement for 1M System
  - Start with Control FPGA ←→ 8 x Readout



Summary

- Summary
    - Pixel descrambling ✓
    - ADC delay automatic adjustment✓
    - Single Module 10GE integration(FW&Tango) ✓
    - 10GE train data in XFEL Train builder format ✓
    - Train **Descriptor & Specific** filled/generated with bunch table✓
    - 10GE Train images can be sorted in increasing **Puls Id** order✓
    - Multi-module system: Control FPGA←→ 4x Readout FPGA✓

Outlook

- Outlook
  - Test, Test, Test…
    - Multi-module system: ASIC Modules, 8xReadout
    - XFEL Train builder & backend PC
  - C&C System Integration →Timing optimization
  - Software Integration