Introduction to Data Acquisition System

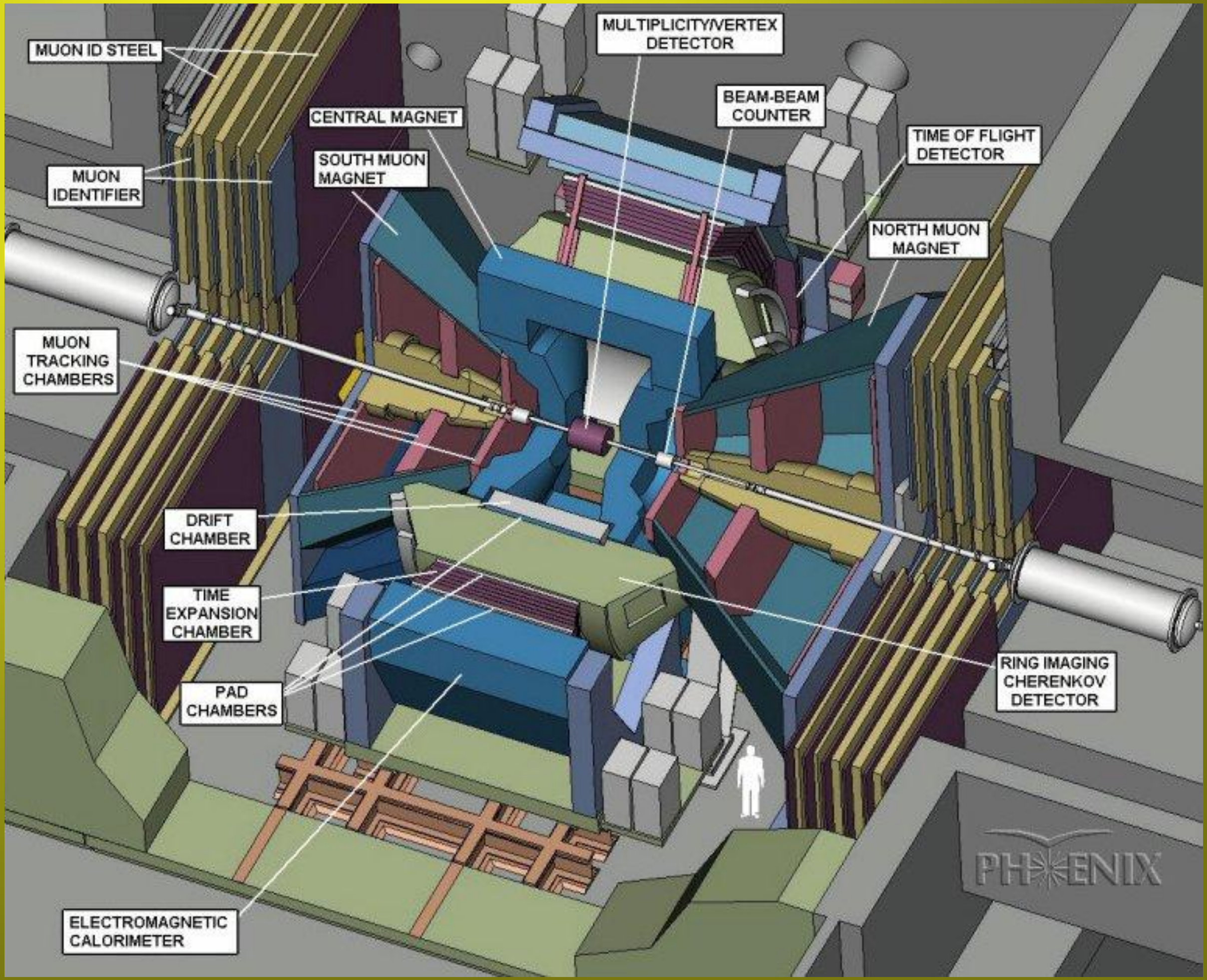
Tutorial for shift personnel

PHENIX Subsystems

PHENIX consists of many subsystems.

Each subsystem must be able to operate as part of PHENIX detector, working in sync with all other subsystems, and at the same time it must be able to make some specific measurements (calibration, time tuning, etc.) independently or as part of a smaller group of detectors.

To satisfy these demands a Granules & Partitions mechanism was developed and implemented in PHENIX DAQ system.



Granules and Partitions

- Here is the main idea of this approach.

For example, before the Run some subsystem(s) would like to tune their detectors. They may like to work with a part of their detector (for example, only with Muon Identifier North Arm (MUID.N)), with a detector as a whole (MUID.N + MUID.S), or with a group of the detectors (MUID.N + MUID.S + MUTR.N + MUTR.S). So we have to be able to "disassemble" PHENIX into some small parts (we call them **Granules**) that are able to operate independently, and then to be able to combine some or all of those granules into independently operating groups (we call them **Partitions**). To better understand Granule/Partition conception of the PHENIX, try to imagine PHENIX as an object assembled out of kid's LEGO bricks. The same way as in LEGO you may disassemble one shape down to individual bricks, and then reassemble different shape or shapes out of those bricks, with PHENIX you may disassemble detector into Granules and recombine them into Partitions.

The main LEGO law is as well applied to the PHENIX DAQ system: **in each taken moment each brick (Granule) may be constituent of only one shape (Partition), while one shape (Partition) may consist of any amount of bricks (Granules).**

The Global Level 1 (GL1) system defines the granules to partitions assignment, as well as triggers to partitions assignment.

Granules

- **Granule** is a minimal combination of DAQ hardware sufficient for data production.

In PHENIX it consists of:

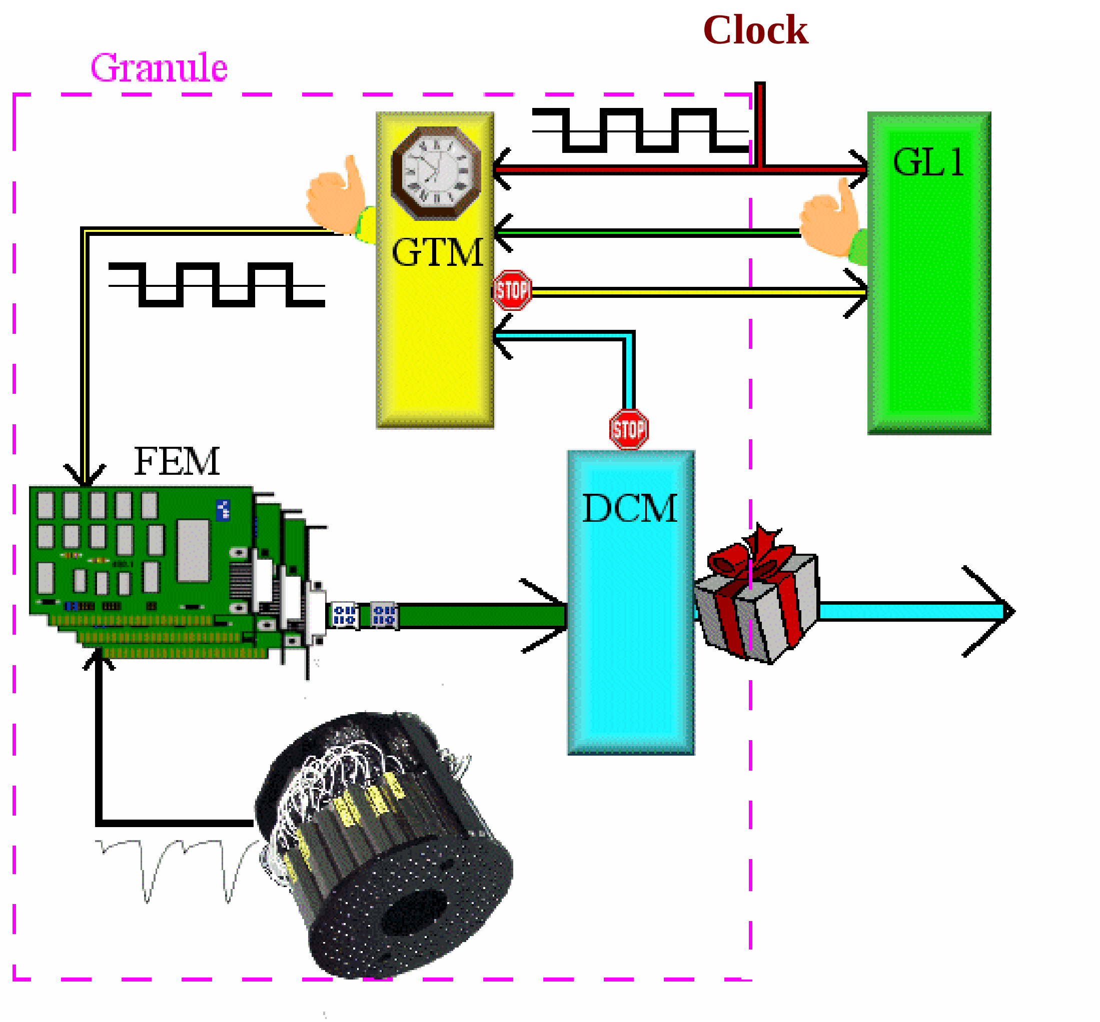
1. **F**ront **E**nd **M**odule(s)
2. **D**ata **C**ollection **M**odule(s)
3. **G**ranule **T**iming **M**odule

Granule may operate both from internal GTM Forced Accept (FA) trigger signal (so called "standalone mode"), as well as from an external Global Level1 trigger

Procedure of data collection:

Trigger signal (from GL1 or from GTM itself) is delayed (if necessary) in GTM, and then is sent to FEMs. On this signal FEMs start to convert and to transfer their data to DCMs. DCMs may write data directly to an output **P**HENIX **R**aw **D**ata **F**ormat (**PRDF**) file, or dispatch data to **S**ub **E**vent **B**uilder (**SEB**) if data are collected through **E**vent **B**uilder (**EvB**)

BUSY signal from DCM vetoes trigger signals when DCM is not ready to receive new data packets.



Partitions

- **Partition** is a combination of the granules, and such group works as one unit.

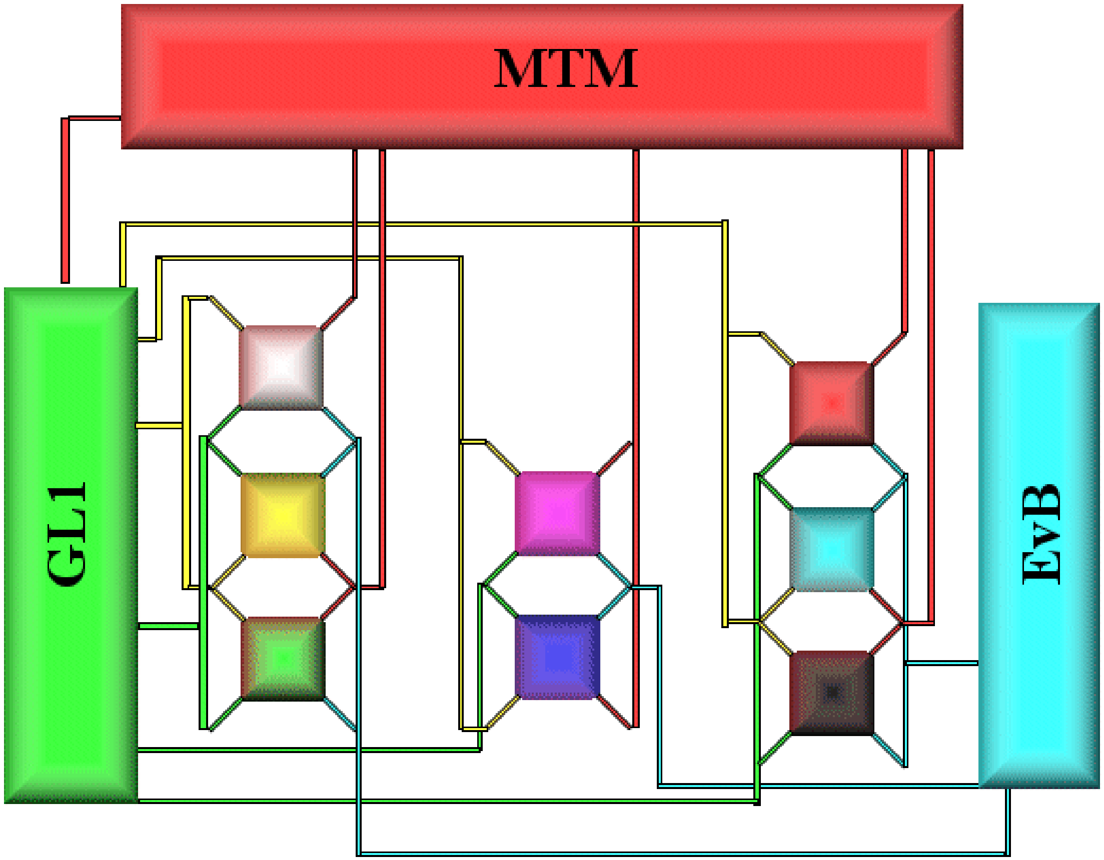
✓PHENIX may have up to **32 partitions**, that share up to **32 granules** and up to **128 triggers**.

✓Partitions operate in parallel and are independent on each other.

✓All granules in partition receive the same triggers assigned to this partition, and these triggers are vetoed by a common partition busy signal, that is an OR of Busy signals from all granules that belong to this partition.

✓If Event Builder is used, for each event data from all granules are combined into one Event packet from this partition. Such packets are written into a separate file.

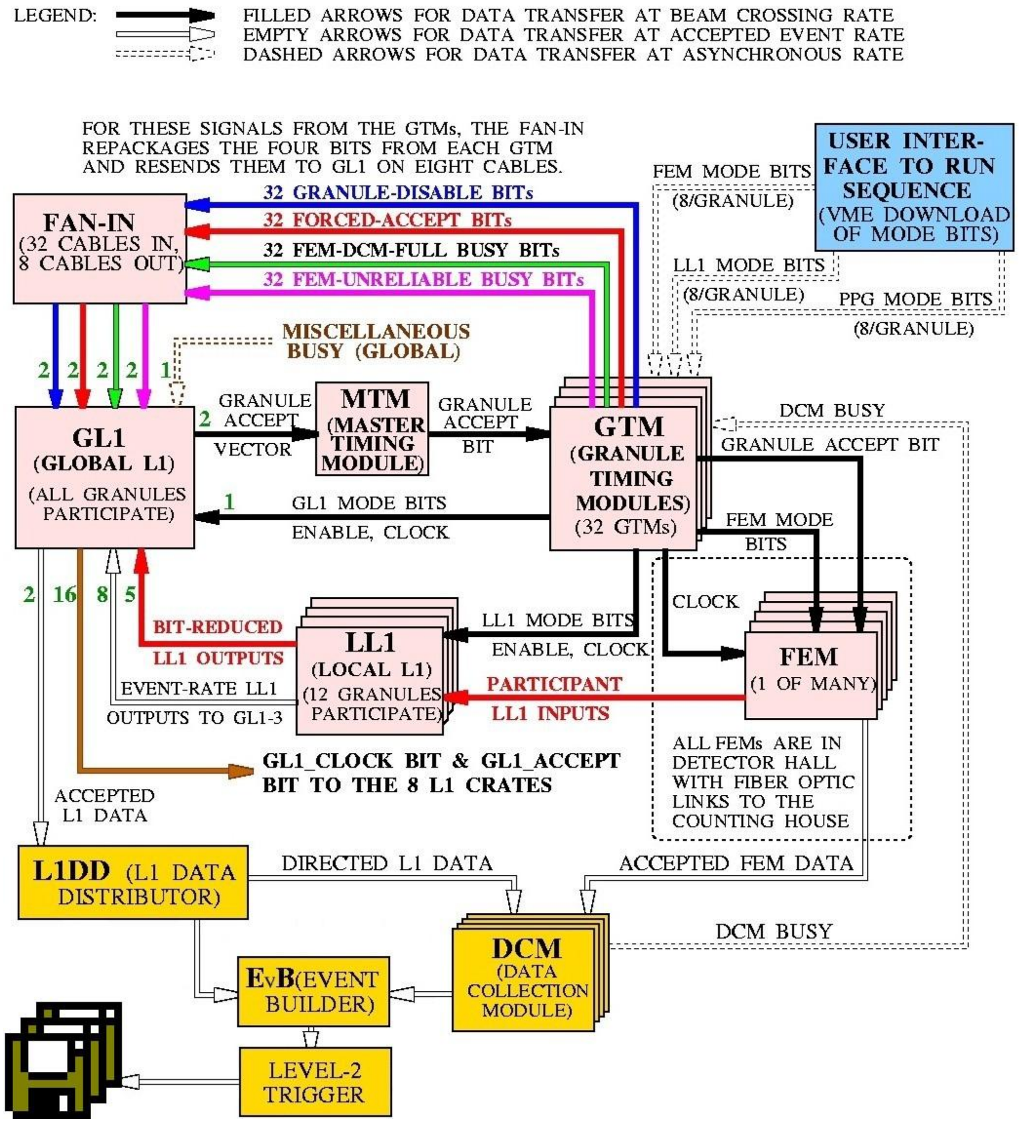
✓All partition configurations are fulfilled by **phTrigger** gui program.



HOW PHENIX DAQ WORKS

Signals flow:

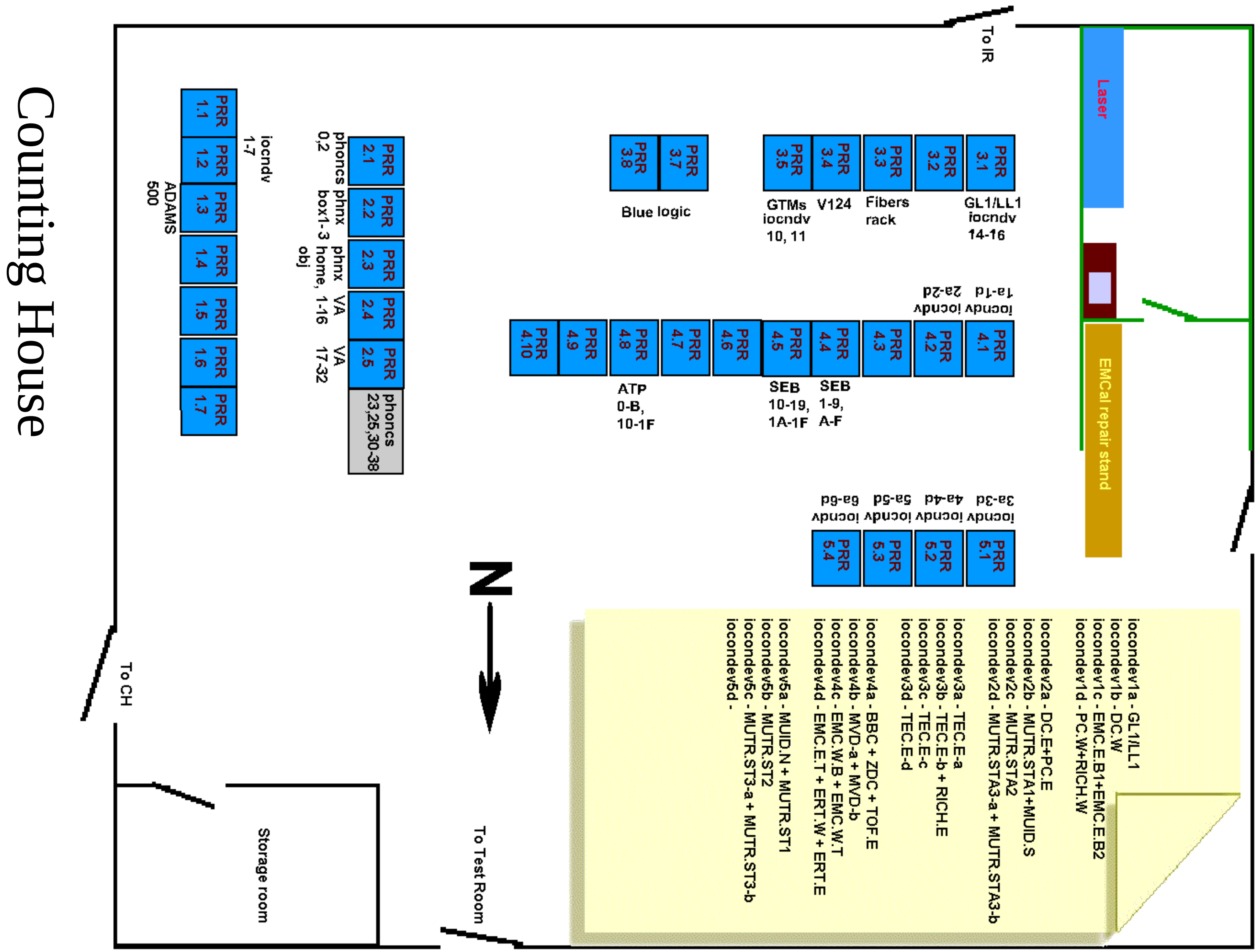
1. **MTM fans out RHIC sync signal (clock) to all GTMs (GTMs must be in the "External clock source" mode).**
2. **After adding (if necessary) some phase delay to the clock, GTM transmits clock to the its Granule FEMs.**
3. **FEMs process its input signals, using clock as start/stop/charge/discharge signal. They store the results in digital (BBC, ZDC, NTC, MUID, …) or analog (EMC, RICH, MVD, …) memory units.**
4. **Data from part of the subsystems – up to now BBC, ZDC, NTC, EMC, RICH, MUID are used by LL1 subsystems for fast analysis (within few clocks) of the data. The result of this analysis is sent to GL1 for final decision taking.**
5. **GL1 process those and other (130 in total) input signals, and if their combinations meet some of the triggers logic, corresponding triggers are fired. According to Trigger->Partition->Granule mapping these signals are transmitted to the corresponding GTMs via the MTM.**
6. **GTM delays the trigger signal so that FEM will extract data from the memory cell that correspond to this event. These data are sent then to DCM. Data from GL1 and all LL1 are sent to GL1 DCM too.**
7. **In accordance to Granules->Partitions mapping EvB combines DCMs data into Partition Event Packet.**
8. **Level-2 analyzes events and decides if they are worth to be stored on disks. Accumulated data (PHENIX runs) are then sent to HPSS and are stored on tapes.**



View of the Rack Room

Most of PHENIX DAQ hardware is located in the Rack Room. Below is a view of this room from the Counting House side entrance.

PHENIX Rack Room map

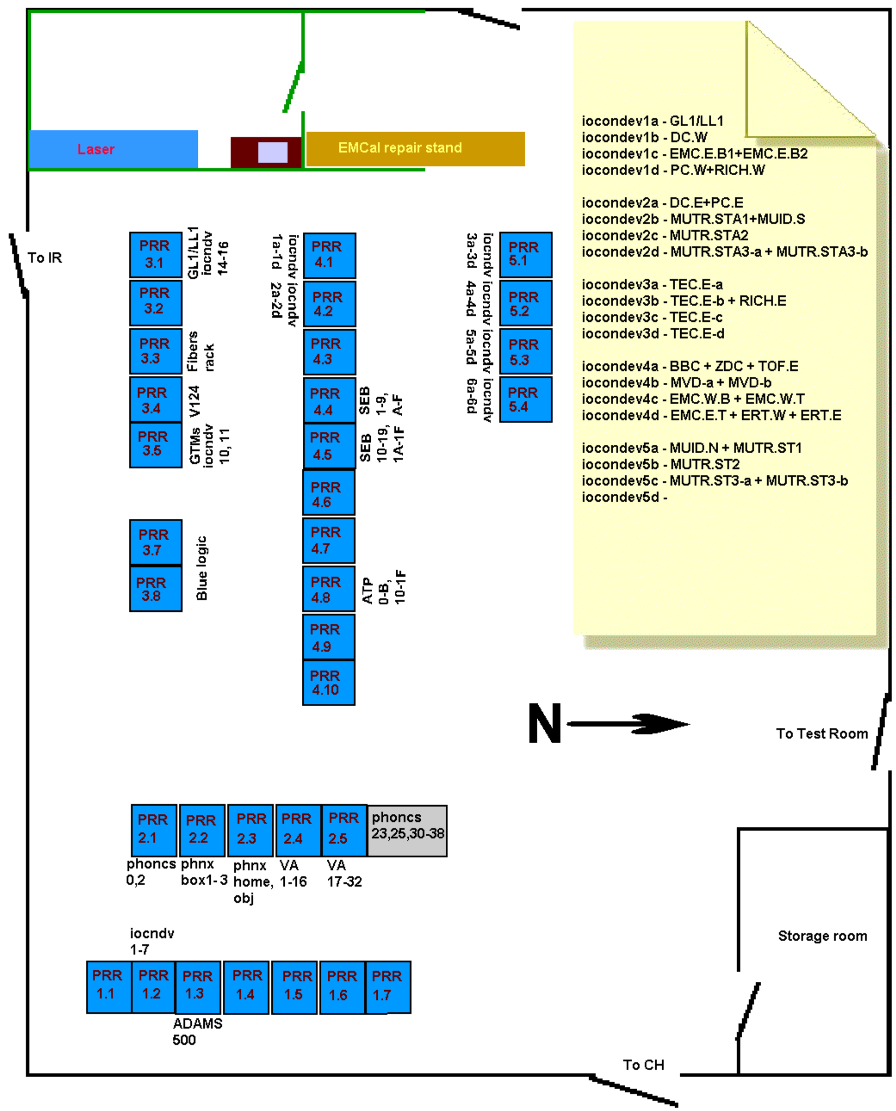


DAQ electronics location

- Subsystem FEMs are located in Interaction Region (IR) at the vicinity of their detectors or directly on the detectors. All signals from and to the FEMs are transmitted through optical fibers.

  The rest of DAQ electronics is located in the Rack Room. We name VME crates by an IP name of the Power PC installed in it.

- The **GTM**s reside in two VME crates: iocondev10 and iocondev11, mounted in rack **PRR 3.5**
- Crates with **DCM**s are mounted in racks **4.1, 4.2, 5.1-5.4**
- **G**lobal **L**evel **1** and **L**ocal **L**evel **1** electronics are located in the rack **3.1**
- **Blue logic** NIM crates are in racks **3.7, 3.8**
- **SEB** computers are in racks **4.4, 4.5**
- **ATP** computers reside in rack **4.8**
- Data storage phnxbox 1-3 are located in rack **2.2**



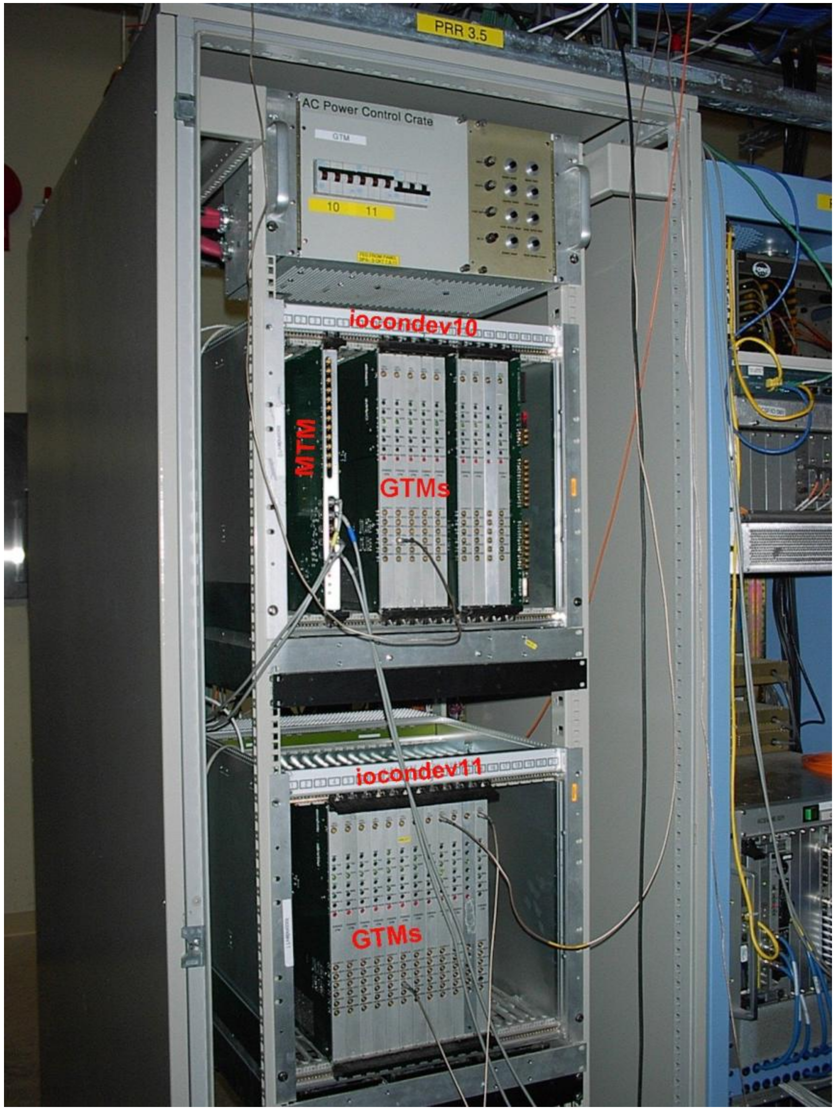
Granule Timing Module: location

GTMs are located in the two crates in the rack PRR 3.5,in crates iocondev10 and iocondev11. In the crate **iocondev10**:

Granule name	Granule #	Slot	Busy Cable
MTM		4	
BB	2	8	116
ZDC	3	9	121
MVD	4	10	120, 127
DC.W	5	11	5, 16
PC.W	6	12	6
TEC.W	7	13 [not installed]	
RICH.W	8	14	8
EMC.W.B	9	15	118
EMC.W.T	10	16	119
ERT.W	23	17	131
FCAL.N	24	18	
FCAL.S	25	19	

iocondev14:

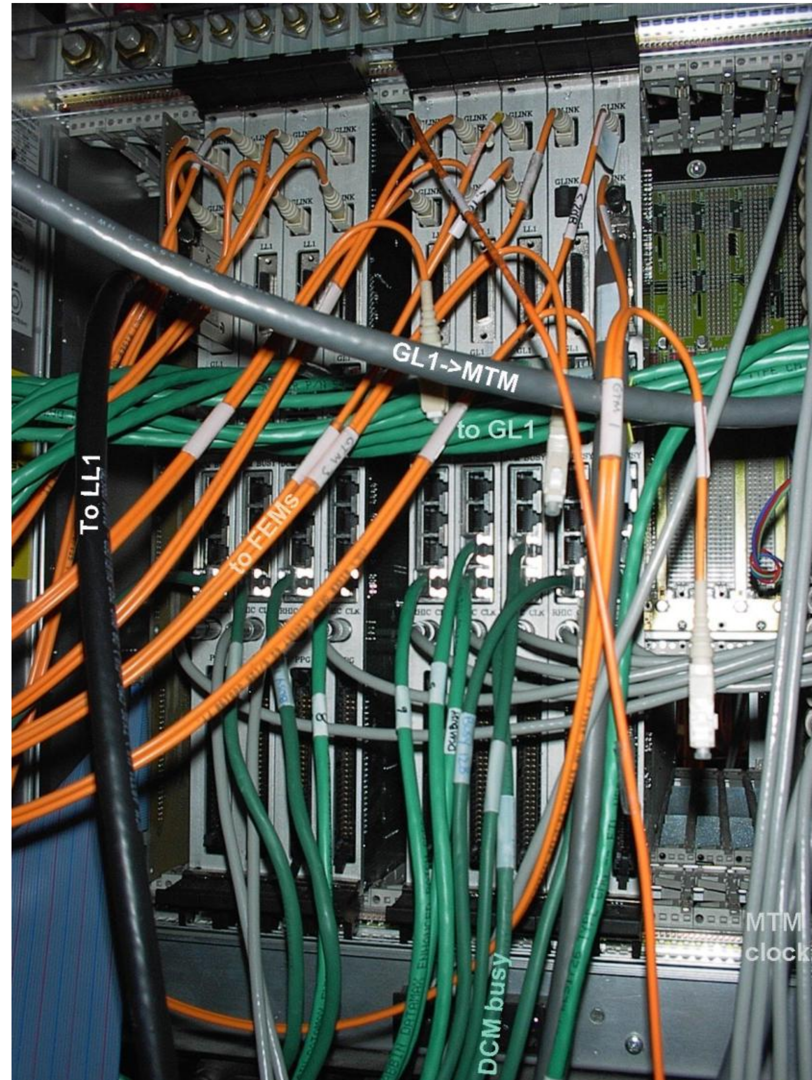
Granule name	Granule #	Slot	Busy Cable
GL1	0	5	



GTMs location (continued)

- **iocondev11:**

Granule name	Granule #	Slot	Busy Cables
DC.E	11	4	2, 4
PC.E	12	5	12
TEC.E	13	6	111,112,113,114
TOF.E	14	7	125
RICH.E	15	8	104
EMC.E.T	16	9	117
EMC.E.B	17	10	9,10
MUTR.S	18	11	11,15,17,18
MUID.S	29	12	19
MUTR.N	20	13	116,122,123,124
MUID.N	21	14	13
ERT.E	22	15	126



All communications of GTM with GL1, MTM, DCMs, FEMs go through a backplane transition card. Signals to the FEMs are sent via optical fibers (orange cables), all other signals (to GL1, LL1; from DCM, MTM) are sent via different types of wire cables.

GTMs are named according to their Granule name, for example, GTM.DC.E, GTM.EMC.E.B, GTM.GL1.

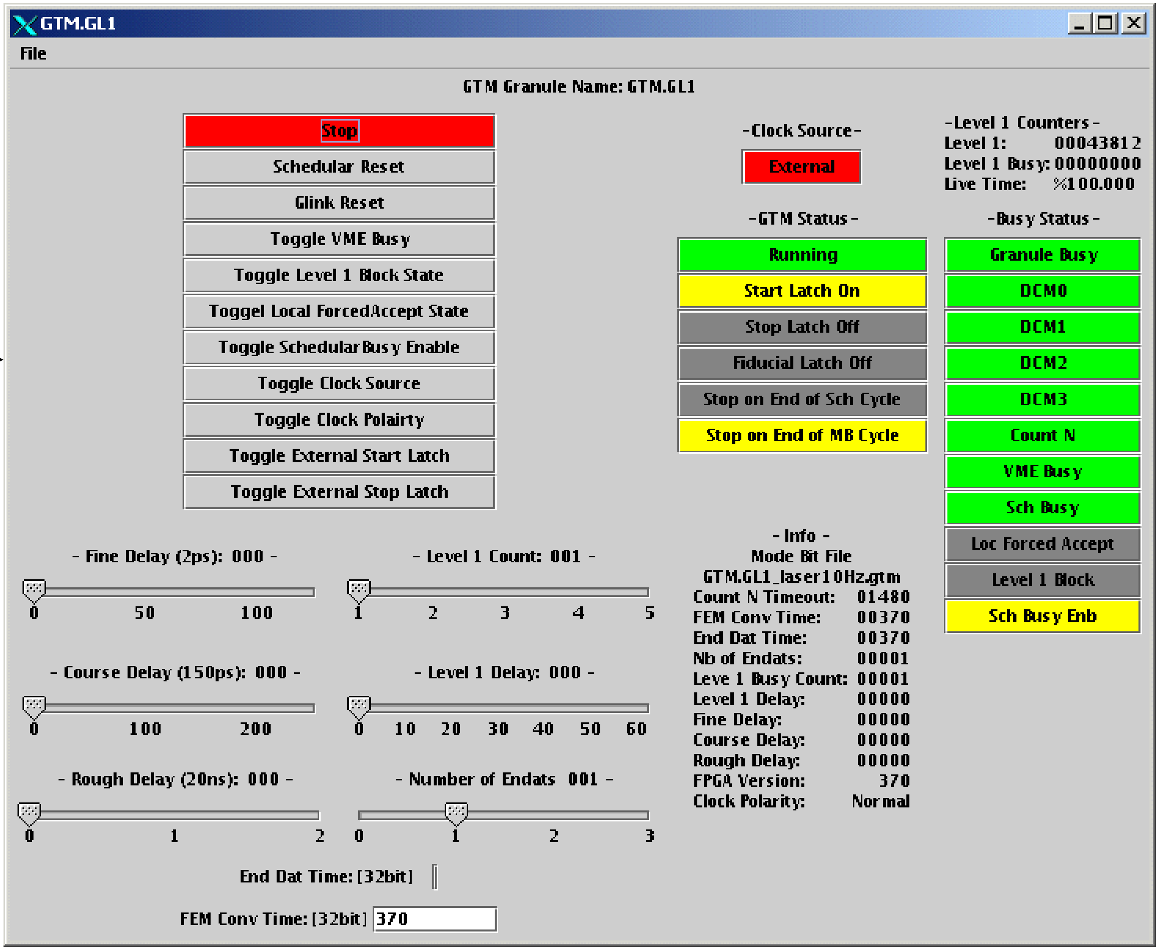
GTM operation

- **What GTM does:**

1. Transmits RHIC synchronization clock signal to FEMs. GTM receives clock signal from Maser Timing Module (MTM) or can generate it by itself. Time phase between moment of a collision and the clock signal must be adjusted for some subsystems. For example, BBC electronics make two signal conversions in clock period: first is used as pedestal, and the second as a signal. So clock arrival must be tuned so that PMT signals are delayed on 66 ns with respect to the clock signal. GTM may adjust clock delay with ~ 2 ps accuracy.
2. Forwards trigger signal received from the MTM to the FEMs. Since Trigger System need some time (of the order of few tens of beam clocks) to make a trigger decision, each FEM has a digital or analog buffer where it stores digital or analog data for at least 40 consecutive beam crossings. To adjust an arrival of trigger signal to FEM so that data from the correct buffer cell will be transferred to DCM, GTM may delay trigger signal in beam clock units.
3. GTM defines FEM conversion time and data transmitting time in units of beam clocks.
4. FEMs may need to fulfill some specific operations periodically during the data taking, for example, to make some reset of electronics, or to discharge integrating capacitors, or adjust base line. To synchronize such operations with DAQ data taking procedure, GTM sends Mode Bits to FEMs, GL1/LL1, and **P**rogrammable **P**ulse **G**enerator (the last procedure is fulfilled generally by GTM.GL1). Mode Bits are programmable 8 bits that are defined for each beam crossing. A special Mode Bit file with extension .gtm is downloaded into a GTM each time when you hit **Download** button on a Run Control gui. To know more about GTM, and how to prepare Mode Bits and .gtm file, read chapter Timing System at PHENIX web site http://www.phenix.bnl.gov/phenix/project_info/electronics/electronics_index.htm
5. DTM forwards busy signal received from DCM, to GL1. It also applies its own busy to prevent triggers arrive faster then FEMs can convert and transmit its data. Since each FEM has transmitting buffer to store converted data from up to 5 events, GTM busy may be adjusted so that it will be set only when the FEM transmitting buffer is full.
6. GTM may operate in so called "stand alone" mode, when it by itself reroutes "**F**orced **A**ccept" mode bit, generally sent to GL1, directly to FEMs. This mode permits to work with individual Granule by-passing the GL1 electronics. You may initiate this mode by typing: *rc –standalone* from **phnxrc** account on **va010** computer.

GTM control GUI

To change some GTM settings directly, you may call GTM gui by typing: *gtm <GTM name>* from **phnxrc** account on **va010** computer. Or you may click on **OK** button in the row with Granule name on a Run Control GUI panel. I'll get the GUI panel like this one →

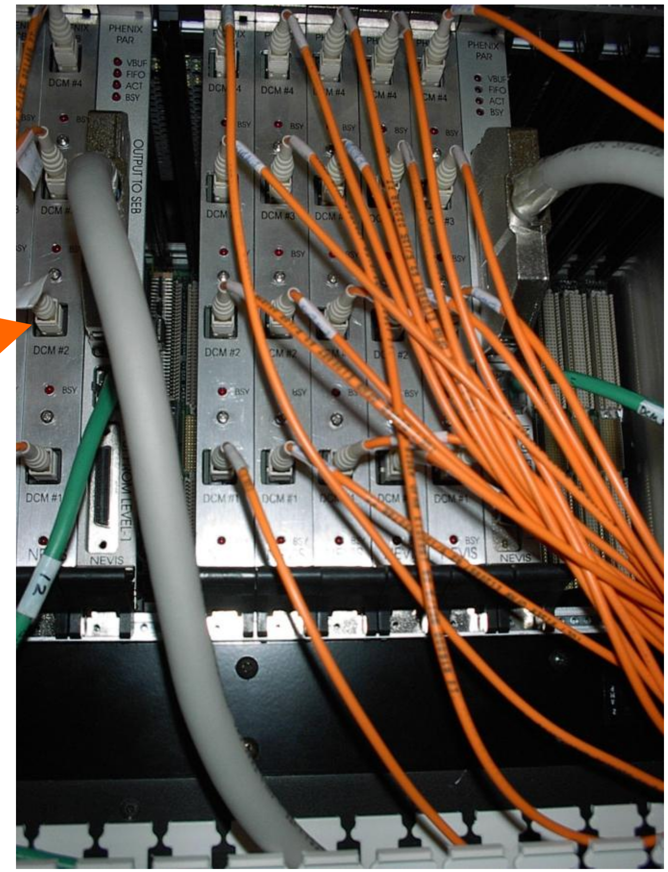


Tips:

- Keep Clock Source as an External one. This permit you to synchronize your Granule with other subsystems, with GL1 for example. The only reason to use GTM internal clock is an absence or malfunction of the MTM.
- If you have changed clock delay, click Glink Reset button: in case of significant delay change Glink may be lost

Data Collection Modules

DCMs are located in VME crates mounted in racks 4.1-4.3, 5.1-5.3. Data from FEMs to DCMs are transmitted via optical fibers. One DCM board has 4 DCMs, and up to 4 FEMs may be multiplexed to one DCM input. Since multiplexing increases data collection time, we are trying to keep it as low as possible. Right now the maximum multiplexing is two FEMs. The DCMs in a crate are grouped into partitions (do not mix them up with DAQ partitions!). Partitioner module accompanies group of DCMs in a partition. It combines DCM's busy signals and sends it to corresponding GTM module.



DCM data readout modes

- There are several modes of data readout from DCMs.

You may select DCM readout mode from Run Control GUI submenu "**Config Readout**"

1. DSP 14. **D**igital **S**ignal **P**rocessors 1-4 read directly DCMs 1-4. Readout is accomplished via VME.
2. DSP 5. DSP 5 accumulates data from DCMs 1-4. Readout from DSP 5: via VME.
3. PART[itioner]. Partitioner receives data from DSP 5 via back plane bus. By comparison of the data received from DSP 5 with the data from Partitioner we may test bus operations. Readout from PART: via VME. This mode is mainly used in ***rc -standalone*** data readout mode.
4. JSEB. In this mode data from Partitioner are sent to Sub Event Builder computer via JSEB card. This is the fastest way of data transferring, but used only with Event Builder. Main mode for physics data taking. If by some reason you have some problem with EvB, try to switch readout mode to PART.
5. DCM output data format: Zero suppression On/Off. To reduce data size DCM may fulfill zero suppression. In this mode data are compared with some thresholds, and if value is less then threshold it is discarded. To know more about zero suppressed data format for different subsystem look at DCM web page.

Trigger System

- Trigger electronics are located in rack PRR 3.1

1. GL1 is located in the top crate iocondev14. It consists of several boards (front):

Board name	Slot (fixed/range)	Installed
PPG	3 (yes)	yes
GL1-2	4 (yes)	yes
GTM	5 (yes)	yes
GL1-1	9 (no/6-13)	yes
GL1-1	(no/6-13)	no, spare
GL1-1P	11 (no/6-13)	yes
GL1-1P	(no/6-13)	no, spare
GL1-3	16 (yes)	yes

- Minimal working combination of GL1 boards: GTM, GL1-2, GL1-3, one GL1-1. GL1 may have 7 additional boards – any combination of GL1-1 and GL1-1P boards. Boards responsibility:
- **PPG** – Programmable Pulse Generator: issues signals to run Laser, EMC test pulses.
- **GL1-1** – defines 32 triggers: logic, Trig->Partition map, mask off, Scale down values.
- **GL1-2** –handles Granule's busy signals, Dead-for-N busy.
- **GL1-3** – Trigger->Partition->Granule mapping, defines event buffering counter (1-5), governor delay.
- **GL1-1P** – configures 4 luminosity scalers. Mainly used in PP runs only.

How GL1 works

- **Input signals**

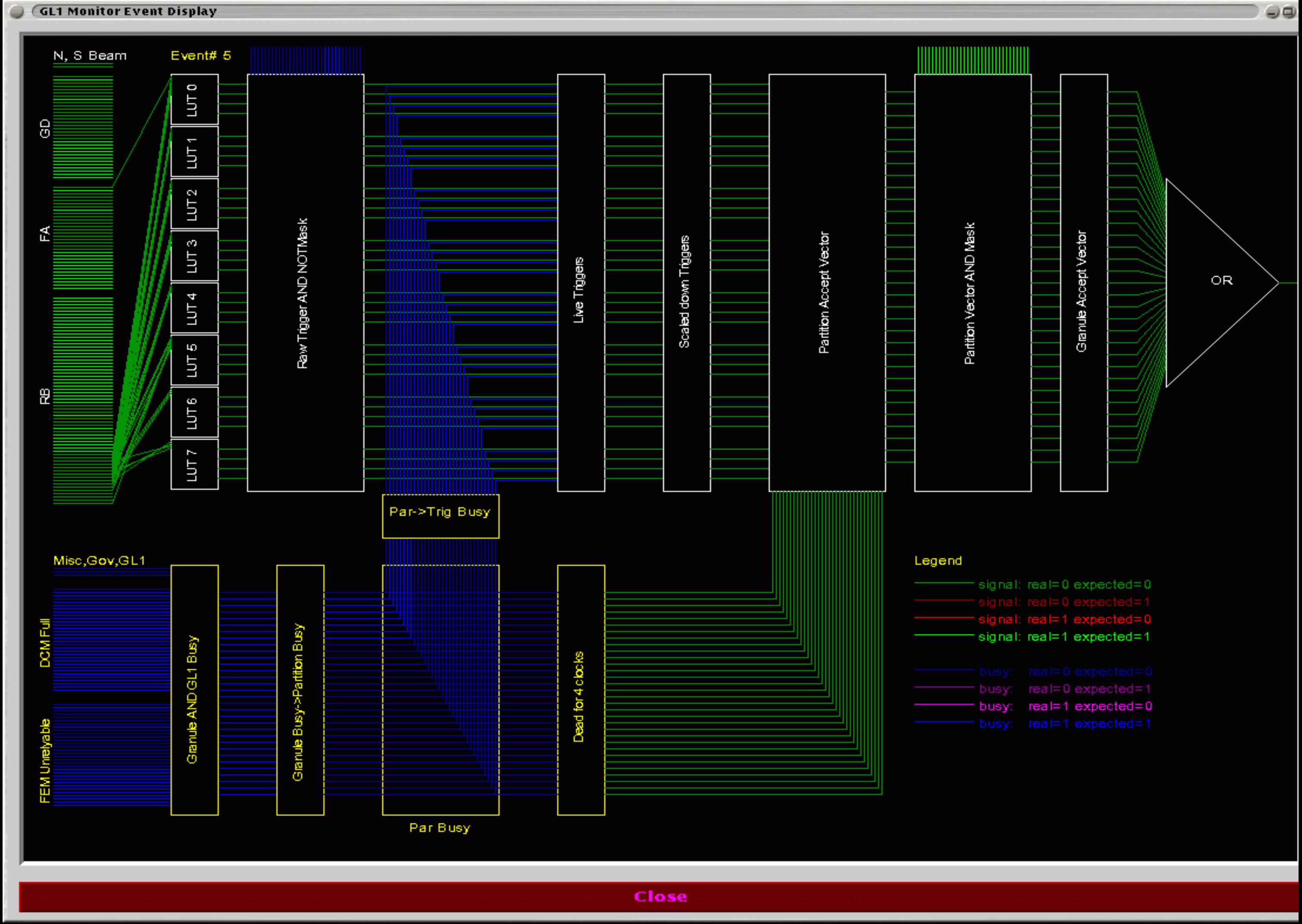
GL1 receives signals:

Signal name	Description	Number of signals
LL1 Reduced Bits (RB)	Define intermediate decisions done by subsystem's LL1 modules.	64
Forced Accept (FA)	generated by GTMs, programmable. Used as a trigger demanded by a Granule	32
Granule Disable (GD)	generated by GTMs, programmable. Used to inform GL1 that a Granule may not work properly. Currently replaced by RBIB signals	[32]
FEM Unreliable (FU)	generated by GTMs, programmable, used to prevent data taking while FEMs are in unreliable state: reset, discharge, initialize or alike stage	32
Granule Busy (GBY)	•busy signals received from GTM. These signals are logical Or of DCM busy and GTM busy signals.	32
Reduced Bit Input Board (RBIB)	Mainly different blue logic signals. Currently substitute GD signals	32
Ring Fills	Define if a bunch is empty or filled. Not implemented yet.	2
Miscellaneous Busy (MB)	Issued by GL1 DCM (iocondev1a), used as Global DAQ busy.	1

In addition two more signals participate in GL1 operation: VME crate busy – set/reset via VME command, and Governor Busy – set if the trigger rate exceeds defined maximum frequency. Any of the busy signals: VME, Gov., or MB vetoes readout of all partitions. (In standalone mode granules works independently from GL1, and GL1 busy do not affect standalone readout prucedure)

In total GL1 receives 130 signals and 32+3 busy signals.

Global Level 1 block diagram



2. How GL1 works (continued)

Previous slide shows how minimal GL1 configuration works.

1. Input signals (N, S Beam, GD, FA, RB) can be connected to 8 Look Up Table (LUT) chips. The connections are programmable and are defined in phTrigger GUI Bit->LUT pannel. Up to 20 signals may be connected to the input of one LUT. The same signal may be connected to more then one LUT, and to more then one input of the same LUT. Four outputs of each LUT are what we call “Raw Trigger” signals. Rate of the Raw Trigger shows the performance of the RHIC and corresponding trigger systems, but not the DAQ performance. So one GL1-1 board defines 8x4 = 32 triggers. Logic of each LUT is programmed in the same phTrigger program. For more information on GL1 programming see http://www.phenix.bnl.gov/phenix/WWW/publish/belikov/GL1/New_GL1_Operations_Manual.htm

2. Raw triggers can be masked off, that is disabled. Trigger Mask is defined in phTrigger or in rc GUI. Disabled triggers are still seen as Raw Triggers, that is their scalers will rolls up, and Raw Trigger bits will be set in GL1 data, but their Live and Scaled Down Triggers will never fire.

3. Granule Busy (or DCM full) signals are combined with FEM Unreliable signals and with 3 GL1 busies by the rule: BUSY(i) = GBY(i) | FU(i) | VME | MB | GovB. The resultant busies are combined into Partition Busy Vector (PBV) according to Granule->Partition map. Thus, busy signal from any of the granules that constitute a partition, raises corresponding busy bit in PBV. This vector then is OR-ed with Dead-for-N Partition Busy Vector (D44, see next slide): PBV(i)=PBV(i) | D44(i). Resultant PBV is unfolded into Trigger busies according to Partition->Triggers map. Thus, all triggers that belong to the same partition will be vetoed if partition busy bit is high.

4. Trigger signals that passed through their busy vetoing are called “Live Triggers”. Rate of these triggers takes into account DAQ dead time. The ratio of Live/Raw is used as a measure of our DAQ live time.

5. Generally we don’t want to record all events fired by some triggers. For example, we use “Clock Trigger” that fires each beam crossing, but we are unable and don’t want to record all of them, but only predefined portion of such events. To implement such “portioning” GL1 accomplishes so called “Scale Down” mechanism. This is how it works. For each trigger we write some value into corresponding Scale Down Counter (SDC). Each fired live trigger decrements its SDC. When SDC becomes equal to zero, a “Scaled Trigger” signal is fired. Thus, for example, if we set SDC =1, then only each second live trigger will evoke scaled trigger, and if SDC=2 only 1/3 of live triggers will be recorded. So Nscaled=Nlive/(SDC+1).

3. How GL1 works (continued)

6. Scaled down triggers are combined into Partition Accept Vector (PAV) according to Triggers->Partition map.

7. Some of PHENIX subsystems are unable process events if they happened too close in time, mainly if $\Delta t < 4$ clocks. To prohibit such cases GL1 issues Dead-for-N clocks busy signal, where N is programmable and generally is equal to 4 (this is why sometimes we call this busy signal Dead-for-4, or just Dead44). For each fired scaled triggers the Dead44 busy is raised for the partition to which this trigger belongs. Such formed Dead44 Busy Vector is combined then with PBV (see previous slide).

8. According to Partition->Granules mapping, PAV is converted into Granule Accept Vector (GAV). This vector is sent to MTM, where its bits are split into trigger signals for each GTM.

9. An OR over all GAV bits is used by GL1 as an input to the GL1 Transmissions Counter.

GL1-1P board works the same way as GL1-1 board, but there are significant differences:

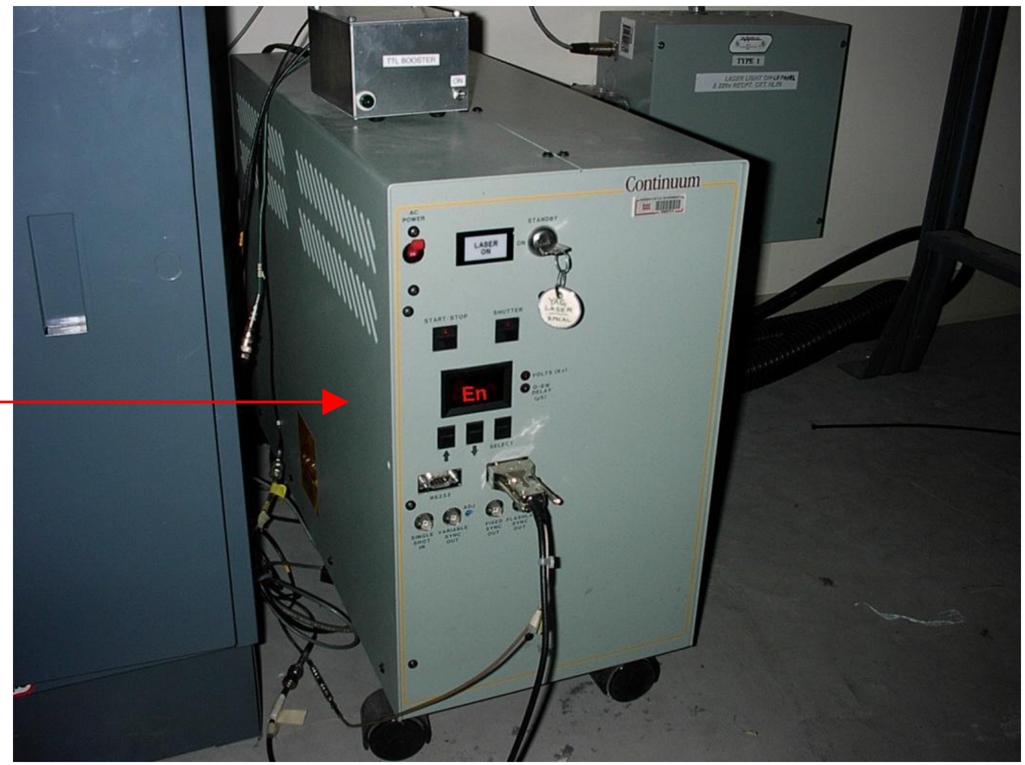
a) one GL1-1P board defines only 4 triggers;

b) GL1-1P has no scale down mechanism;

c) GL1-1P triggers are not combined into PAV;

d) Each GL1-1P trigger has 120 scalers, one for each beam crossing, so that each scaler is incremented only when the trigger fired in the corresponding beam crossing.

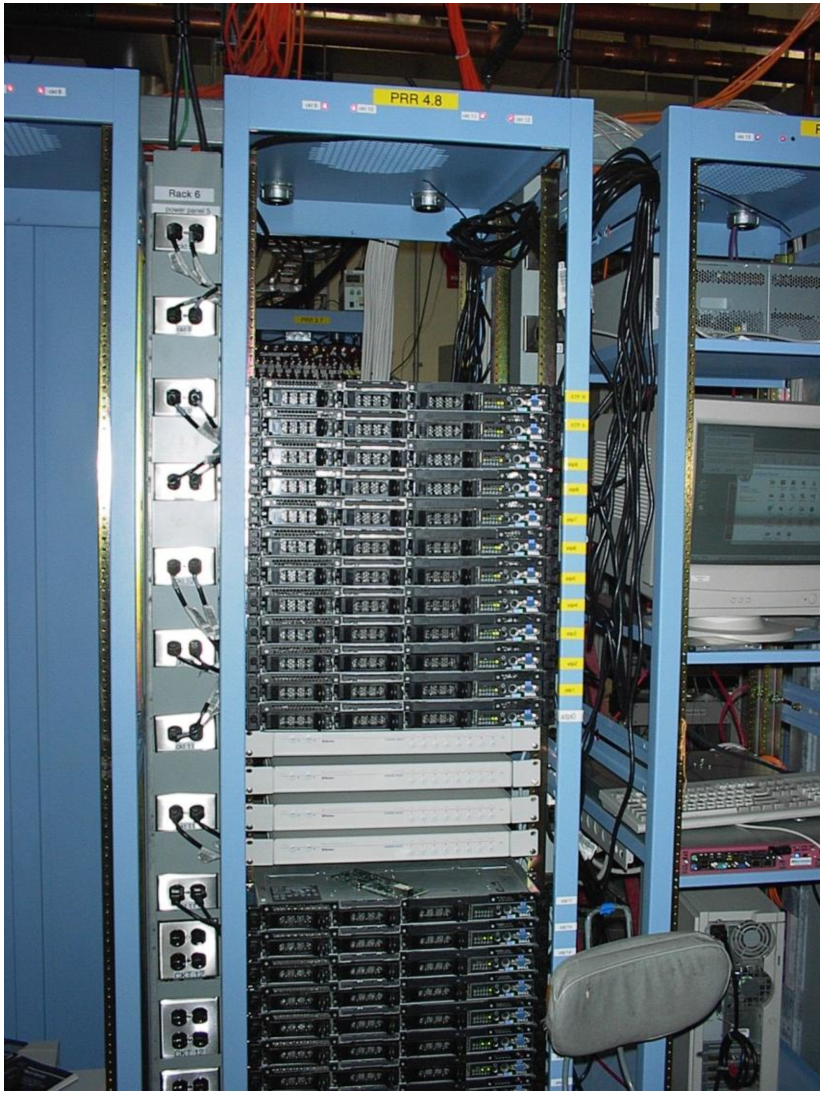
Monitoring Laser



Event Builder

Sub Event Buffers

Assembly/Trigger Processors



Data Storage

Arcnet

Run Control

Online Monitoring