SPS LEP LHC

Note: SPS and LEP are very different machines,

for LHC the experience with the LEP timing is the most relevant

Information on SPS and LEP (slow) timing systems.

- –What functionality it provides
- –What functionality it does not provide
- –What we can learn
- –Observations & Conclusions

Presented at the LHC timing workgroup 12-may-1999, M.Jonker

SPS timing (what it is)

- Distribution of short synchronized timing messages throughout the SPS machine (and a small part of LEP).
- Most equipment is timing event oriented
  - 4 byte event format: (header, event, cycle, cycle#).
    With TG3 constraint 2.4 byte effectively available:
    - fixed header
    - cycle = 1,2,3,4,5,6
    - event and cycle# are free
- There are from 40 to 100 events generated per cycle:
  - Equipment oriented events
  - Beam / cycle oriented events
- Only a few asynchronous events.

SPS timing, how it works

- Timing requirements are defined in templates per cycle type and per equipment.
- Event times are defined relative to beam-processes (I.e. injection, acceleration, extraction, dump)

```
/* KIK_proton common KIK events for proton cycles              */
/*-------------------------------------------------------------- */
kiksps:mkp.inj.wn.1:proton:01:-1000:9401  (all,   injection)
kiksps:Dump.Reset..:proton:01: -100:9401  (all,   injection)
kiksps:Dump.Trigger:proton:01:   -6:8001  (end,   flat_top)
kiksps:Dump.Check..:proton:01:   -1:9401  (end,   flat_top)
kiksps:Dump.Trigger:proton:01:  +30:1001  (last,  injection)
kiksps:Dump.Trigger:proton:01: +224:0401  (first, coast)
```

```
/* ARF_leptons
/*--------------
*/
rfqlep:filling..
arfsps:lep.resync
arfsps:wn.first.inj:e*****:00:-0650:8601  (first, injection)
arfsps:wn.all.inj..:e*****:00:-0600:8601  (first, injection)
arfsps:wn.2nd.inj..:e*****:00:-0030:8601  (last,  injection)
arfsps:prep.extr...:e*****:00:-0100:8601  (first, extraction)
arfsps:wn.all.extr.:e*****:00:-0006:8601  (first, extraction)
arfsps:st.flat.top.:e*****:00:-0015:8601  (first, extraction)
arfsps:en.flat.top.:e*****:00:00120:8601  (last,  extraction)
```

SPS timing, how it works

- Timing programs are assembled from these templates per cycle, using the “beam-process” times.

```
CY_proton_md120 =
('-key=injection(0,1200) -key=economy_full(999) -key=flat_top(2460,04800) -key=coast(4570)
  -cycle_no=01' *_proton *_proton_md )

CY_lead =
('-key=injection(0,1200,2400,3600) -key=economy_full(999) -key=economy_lepton(999)
  -key=flat_top(7650,12450) -key=ss_e(7650,12450) -cycle_no=01' *_lead   *_lead_ft )

CY_eplus_20 =
('-key=injection(0,30) -key=extraction(385) -cycle_id=eplus.' *_lepton)

CY_eplus_22 =
('-key=injection(0,30) -key=extraction(405) -cycle_id=eplus. *_lepton)
```

SPS timing, how it works

- Cycle tables are combined in SuperCycle timing programs using cycle start times.

```
# cycle 924:  fixed target cycle, two lepton cycles 22 GeV, MD Proton cycle 26 GeV.
CYCLE                              $mtgctl_14.4
CYCLE -offset=0     -ssc_len=14400 $CY_proton_noeco
CYCLE -offset=09730 -ssc_len=14400 $CY_eplus_22  -cycle_no=01
CYCLE -offset=10930 -ssc_len=14400 $CY_eminus_22 -cycle_no=01
CYCLE -offset=12482 -ssc_len=14400 $CY_protmd_26
CYCLE -offset=12482 -ssc_len=14400 $CY_prot26STORE
```

- A single SuperCycle timing table is loaded in the MTG hardware card at the time. (They are not stored in the MTG host computer).

SPS timing, how it works

- Once the timing table is loaded in the MTG hardware card, the timing runs autonomous. You could shut down the MTG-host.
- A monitor tasks in the MTG host computer surveys the timing and may take the following action:
  - De-synchronization with PS ⇒ resynchronize by selecting economy mode
  - No beam in the SPS (PS error or SPS error) ⇒ economy mode
  - MPS switch request ⇒ Special switch-on/switch-off cycle
  - Operator coast request ⇒ coast cycle
- Real time mode selection of events by timing mode masks. This was implemented for the PPbar mode.
- The MTG has no influence on the selection of executed cycles in the PS. The MTG is the executing agent only.

SPS timing, how it works

- Beam requests to the PS
  - Request from operator. Interlocks status from the Sps Software Interlock Surveillance (SSIS). (Beam interlocks not fully cycle oriented, but they will be with SPS2001).
- Change of super cycle
  - Only one SuperCycle active at the same time due to hardware limitations in the past.
  - Changes are manual and slow (due to the reload time of the front ends and not due to timing)
- Use of asynchronous events
  - Required to confirm a previous function update request.
  - VERY IMPORTANT: the only trusted way to commit a parameter update request send to all equipment.

SPS timing, how it works

The future with SPS2001 / CBCM

- SPS local timing (classical event generation) will become cycle oriented
- SPS local timing driven by the cycles in a sequence selected by the CBCM
- The CBCM combines local cycle information from the SPS (I.e. cycle-priorities, injection-vetos, and cycle-vetos) with similar information from the PS complex to provide the best possible sequence of cycles. (automatic/semi-automatic)

and now for something completely different...

LEP timing

- LEP: a non cycling machine for most of the time:
  - Idle 5%
    - No events
  - Cycling (I.e. filling) 15 % (Used to be even lower during Z_0 running)
    - 5 events per cycle (10 per super cycle) + 1 asynchronous event / minute
  - Energy ramp 5%
    - 3 events to start, (1 or 3 events to end), no cyclic events
  - **Coast 75%**
    - no cyclic events
    - up to 30 unscheduled/asynchronous event per minute (commit trim requests)
    - measurement coordination event sequences. K-mod: 160 events / second
- Usage of timing is most important during a coast (non cycling related usage).

LEP timing

- Filling

  No sophisticated rendez-vous:

  – When LEP needs beam, the LEP/SPS operator sets the lepton request. Usually this is done before to make sure that leptons are healthy in the SPS.

  – LEP MTG timing synchronizes to SPS/PS timing

  Cyclic events are used to:

  - trigger acquisition of data (e.g. first turn)
  - ~~Arm the injection kickers~~
  - Trigger the RF synchronization (Note: the roles are reversed for fast timing, LEP-RF is master, PS/SPS-RF are slave, however the PS/SPS request synchronization by a slow timing message.

  Asynchronous events to commit trim request in the hardware.

LEP timing

- Ramp
  - Under control of a ramp timing table:
    - 3 Events to start a ramp. (PC, RF, ZL)
    - 3 Events to stop a ramp (only one needed, the RF and ZL allow programmed stopped, the PC have not implemented this)
    - Optional acquisition events.
    - Ramp tables are constructed ‘on demand’
  - Only the start is synchronized with the PS/SPS super cycle
  - 1 Emergency stop event (Not used)

LEP timing

- Asynchronous or unscheduled events:
  - Used during filling, used a lot during coasts
  - The only trusted way to commit and synchronize a hardware update request (trim request)
- Event sequences for measurement coordination. The timing provides a way to synchronize control and measurement in various equipment:

  K - modulation

  - programmed up and down events that control up to 8 different groups of power converters.

  Synchronization of measurements with parameter trims

LEP timing

- What is missing
  - **Real real time knobs with predefined functions to step through.**

    K-modulation implementation in the power converters was deliberately (and for unknown reasons) restricted to one step up or one step down.

    Multiple steps, useful for real time knobs in LEP, was made impossible.
  - **Timing controlled measurement procedures**

    Some beam parameters are measured by controlling an equipment and measuring the effect seen by a beam observable. Example:
    - Chromaticity: control RF frequency and measure tune
    - central frequency: control sextupoles and frequency and measure tune
    - dispersion: control frequency and measure orbit
    - Vernier setting, control vernier generators and measure orbits

    Currently these measurements are executed by operator or special software.
  - **Coherent functionality of the function generators for RF, PC’s ZL’s**

    Why did they all implement their own controller, and why couldn't they all look like a Power Converter. This also holds for feedback control loops: Qloop

LHC timing, observations

- PC control fulfills already a lot of the timing requirements (local synchronization)
- PC control will implement a powerful coordinated trim commit system. (independent and concurrent control channels?)
- Are they already doing part of the timing job?
  If so, can we forget about the slow timing all together?
  - Can the power converter controls also be used to control other equipment

    All equipment that needs a parameter control during the ramp or coast, including measurement equipment and feedback control, could use the power converter function generator equipment.
  - Can the power converter control also implement measurement programs like preprogrammed stepping through a function and triggering measurements?

LHC timing, conclusion

The LHC slow timing system should:

- provide a mechanism to apply synchronized commits to parameter modification requests. This mechanism should be available for all equipment. (Power converters, RF, feedback controls, measurement, injection kickers, SPS, PS, etc.
- provide mechanisms for synchronized and sequenced measurement and control procedures.
- provide a central hart beat for equipment synchronization (e.g. 1-ms clock)
- distribute calendar events.
- handle emergency dump requests ?
- distribute machine state ? (exists in LEP but not used)

Note: Whatever mechanism is used by the power converters to synchronize **and control** their equipment should be available to all LHC equipment.

LHC timing, conclusion

- The cycling related requirements of LHC will be very minor:
  - Filling can be easily implemented with the LHC local timing as being driven from the SPS/PS complex.
  - If needed, it could be driven directly by the SPS local timing.
- Simple rendez-vous mechanism:
  - The LHC request beam, and then waits for n pilot-pulses and m filling-pulses from the SPS. When it requests, there need to be some kind of guaranteed response from PS/SPS.

Not only is the LHC machine loosely coupled to the PS/SPS, also the LHC local timing system should be loosely coupled to the PS/SPS complex.

Local synchronization and coordination aspects of LHC equipment are much more important.