J-PARC T59 WAGASCI実験の信号読み出しシステムの開発

Naruhiro CHIKUMA
Department of Physics, the University of Tokyo

竹馬 匠泰
東京大学 理学系研究科 物理学専攻
J-PARC T59 experiment: WAGASCI

Experiment

➢ J-PARC neutrino beam at Neutrino Monitor Hall.
➢ 1 ton target with half H2O/half CH.

Physics goal

➢ Cross section ratio measurement between H₂O/CH for charged-current interaction with different neutrino energy ranges.

Schedule

➢ Detector construction: Started now!
  Complete H₂O/CH Module by Feb/Mar 2017.
➢ NU beam data taking: will start at the autumn 2017.

Neutrino beam flux
Detector configuration

**Three-dimensional grid structure** of scintillator bars.
- $4\pi$ solid angle acceptance around target.
- 3-$mm$-thick scintillator bars.
  - Large target mass of 80\% in fiducial volume.
- 16 layers compose a H$_2$O/CH module.
  - 1m x 1m x 0.5m target region.

**Charge measurement**
- Scintillation light is collected through *WLS fibers* to 32-channel arrayed MPPCs.
- 32 fibers are gathered together by a fiber bundle.
Neutrino beam measurement

J-PARC Neutrino beam

8-bunch spill structure.
- 2.48 sec cycle.
- 8 bunches w/ 580ns time gaps.

Requirement

- Energy deposit --> Tracking, Particle ID.
  ~10 p.e. in average.
  - Threshold @ 1.5 p.e.
  - High accuracy of a few %

- Hit timing --> Hit clustering, TOF.
  - 3ns resolution.

The WAGASCI DAQ

- Open an acquisition gate for the whole period of a spill: ~5 μs.
- Conversion/readout: ~ A few ms.
- Any hits over a fixed threshold during acquisition period are automatically triggered chip by chip.
Photodetector

- 32-channel arrayed MPPC.
  - Type No. S13660(ES1)
  - Dark noise & after pulse suppressed.
  - Noise rate:
    $\sim 6\text{kHz}/\text{channel} \ (V_{th} \sim 0.5 \text{ p.e.})$
    $\sim 100\text{Hz}/\text{channel} \ (V_{th} \sim 1.5 \text{ p.e.})$
    *Over voltage $\sim 3.0\text{V}$
  - Operation voltage: $\sim 56\text{V}$
  - Gain: $\sim 10^6$
  - Flexible printed circuit cable.

<table>
<thead>
<tr>
<th>Number of channel</th>
<th>Water Module</th>
<th>CH Module</th>
<th>INGRID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1280</td>
<td>1280</td>
<td>528</td>
</tr>
</tbody>
</table>

*INGRID modules are not readout by the WAGASCI electronics, but by the T2K electronics with TFBs.

*see supplemental slides.
### WAGASCI electronics

<table>
<thead>
<tr>
<th>Electronics boards</th>
<th>Description</th>
<th>Num/Mod</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ASU</strong> (Active Sensor Unit)</td>
<td>Readouts a 32ch MPPC array with a SPIROC chip.</td>
<td>40</td>
</tr>
<tr>
<td><strong>Interface</strong></td>
<td>Transfers DAQ signals and MPPC bias voltage.</td>
<td>2</td>
</tr>
<tr>
<td><strong>DIF</strong> (Detector InterFace)</td>
<td>Send DAQ signals and SPIROC configuration.</td>
<td>2</td>
</tr>
<tr>
<td><strong>GDCC</strong> (Giga Data Concentrator Card)</td>
<td>Transfer signals between DAQ PC and DIFs.</td>
<td>1</td>
</tr>
<tr>
<td><strong>CCC</strong> (Clock &amp; Control Card)</td>
<td>Provides clock signals and fast control.</td>
<td>1</td>
</tr>
</tbody>
</table>

**Diagram:**
- DAQ PC
- Hub
- Power Supply
- CCC
- GDCC
- Interface Board
- DIF
- ASU
- Water/CH module
- Interface & DIF
- ASU

- Signal from trigger system
- Clock trigger
- Ethernet
- Config
- Readout data
- Chip power 5V
- MPPC HV ~56V
- Digital signal
- Configuration power supply / HV
- 4 ASUs lines
- 7 DIFs per GDCC
ASIC at front-end readout

**SPIROC** (Silicon PM Integrated Read Out Chip)
- Product of Omega (France).
- Dedicated very front-end ASIC for an ILC.
- Both analog signal processing and digital are contained in chip.

- **Charge measurement.**
  2 gains/ 12-bit ADC $\rightarrow$ wide dynamic range: $1pe - 2000pe$.

- **Time measurement.**
  12-bit TDC with $\sim 100ps$ step.

- **Auto-trigger.**
  Internal discriminated signal is used for *Track-and-Hold circuit*.

- **36-channel** readout.

- **16-deep analog memory.**

- **CQFP240** package.

- **5V/3.5V** operation.

- **25μW** per channel
SPIROC2D analog part

- **PreAmp**
  - Low gain: x1 - x15
  - High gain: x10 - x150

- **Slow Shaper**
  - 50 - 100ns shaping time
  - Charge is stored in analog memories with Track&Hold

- **Fast shaper & Discriminator**
  - 15ns shaping time
  - 10-bit DAC threshold
  - Auto-triggering with this discriminated signal
  - Time measurement

Charge & Time is converted by the ADC with ramp signals.
**SPIROC2 digital part**

**Acquisition phase**
- A column is filled, and moves to the next column at the same time for all the channels at timing of the next “bunch crossing”.
- “Bunch crossing” is a coarse time flag for the triggers.
- BCID is controlled by external 2.5MHz clock.

**Conversion phase**
- 36 charge/36 timing in the analog memory are sequentially converted at an ADC with using ramp signals.
- The digital data are stored in 4kbytes SRAM.
Issues on SPIROC2D

- It is only possible to set the discriminator threshold at its undershoot. Due to wrong position between signal and reference in the comparator.

- Much more sensitive to noises on ground.
- But still able to trigger on 0.5 p.e. level.

- Column 10&14 do not work. Reset of the column is not properly done.
- Still able to be used for T2K neutrino beam structure with 8 bunches.

- **Requirement**: Rate of noise and hits from cosmic rays $<< 2 \text{ per spill}$
- **OK**

*MPPC noise rate:~$10^{-2}$/32ch/5μs@1.5PE$_{th}$, Cosmic ray hits: <4x10^{-3}/32ch/5μs@ground*
Front-end boards

ASU (Active Sensor Unit)

- A SPIROC2D is embedded.
- Direct connection to 32-channel arrayed MPPC.
- 50-pin connection to an Interface board.
- Another ASU board can be put serially via the 50-pin connection.
Back-end boards

**GDCC** (Giga Data Concentrator Card)
- Designed on 6U VME format.
- 7 DIFs connections (HDMI). 50Mb/s.
- 1 CCC connection (HDMI).
- XILINX FPGA Spartan6.
  - Connection's speed auto-negotiation.
  - Preamble bits.
  - Trailer check-sums.

**CCC** (Clock & Control Card)
- The GDCC board can also be operated in CCC mode, just by programming the CCC firmware.
- Generate/distribute 50MHz clock.
- Synchronize the whole DAQ system.
- Receive spill signal from beam trigger.
Status of electronics development

- **Production**
  - ASU, Interface – Test production is done. Tested at Utokyo & Ecole Polytechnique.
  - GDCC, CCC, DIF – Final production is done. Tested at Ecole Polytechnique.

- **Test operation has been done.**
  - Periodic data taking only with MPPC dark noise.
  - Confirmed it could be operated at threshold of 1.5 p.e.

- **Threshold**
  - Good
  - trigger channel: single (0ch)
  - Overvoltage: ~2.1V
    - HV: 56.0V
    - breakdown Voltage: 51.4V
    - InputDAC value: 1(2.5V)
  - Gain value (capacitance): 60
  - Gain Select: OFF
    - (this means that the data takes by HighGain)

  - Event rate matches dark noise & crosstalk rate!
**Bunch crossing**

---

**BCID**

- Bunch crossing ID. coarse timing of triggers.
- Bunch structure is well seen.
  - Peak width : ~1 bin
  - but a bit broad...?

- LED keeps injected during the whole acquisition period.
- Some events filled into two columns, due to reflection or slow recovery.

- Trigger channel : single (0ch)
- Bunch: width=50ns, freq:250kHz (10bin)
- Threshold : 2.5p.e. level (DAC value=160)

---

*LED keeps injected during the whole acquisition period.
*Some events filled into two columns, due to reflection or slow recovery.
DAQ signals

- Reset → Acquisition → Conversion → Readout.
- Output data (Dout1b) are transmitted to back-end boards.
- Conversion starts (start_convb) after all of 16 analog memories are filled (ChipSatb).
- Auto-triggers are only valid during the validation signal (val_evt_p) from DIF.
SPIROC2B/D contains two PreAmps of different gains.

12-bit Wilkinson ADCs are embedded for each.

Correct behavior of ADC ramp signals.
- \( N_{\text{peak-ADCramp}} = 2 \times N_{\text{trigger}} - 1 \)
- in order of high, low, high, ... , high

*SPIROC2B ignore the first ADC ramp for low gain because of its fluctuation. This is solved in SPIROC2D.

For high gain:
- ADC ramp

For low gain:
- ADC ramp

Auto-triggers

Validation signal

Trigger signal

Spill
Neutrino beam synchronization

- **Beam trigger** signals are sent to CCC.
  - Data acquisition is done every spill. ➞ Every 2.48sec.
  - The whole DAQ system is **synchronized to 50MHz** clock generated on CCC.

- **Event tagging system:**
  - **SPILL# information** is merged into the readout data at DAQ PC.
  - Readout data contain **BCID** (bunch crossing ID), that gives timing of each auto-trigger as count of 2.5MHz clock signal after acquisition starts.
Synchronous beam triggers are distributed out through “TRIG OUT”
* NIM level / LEMO connection

*by Sakashita-san

✓ Pre-beam trigger
  → 100msec before beam trigger.
  * w/ 16-bit spill number

✓ Beam trigger
  → 40usec before neutrino arrives.

*SPILL# offset should also be taken into account.

SPILL# (lower 16-bit) is distributed out from 16-bit output of ECL/NIM converter module.
* ECL / 2.54-mm-pitch 34-pin flat connection (or 16 NIM out / LEMO)
**Trigger patterns**

1. **Beam**
   - 40us before neutrino beam arrives.
   - Pre-trigger stops all the other triggers’ activity.
   - Beam trigger width/delay must be adjusted on CCC.
   - Exactly 100ms before beam trigger.

2. **Periodic**
   - Fixed period: More than ~1ms for convert/readout

*Acquisition width must be calculated and fixed by using noise rate for filling many of 16 deep memories.*

*Max of DAQ frequency is 100Hz, due to handshake b/w DIF and GDCC.*

*Margin time between beam triggers can also be used for periodic acquisition.*
**Software**

- **Initialize.sh**
  - Define commands in software

- **configure.sh**
  - Send configuration file to chip

- **load_config.sh <config_file>**
  - Load configuration file

- **Start_run.sh <output_file>**
  - Start data acquisition

---

**Pyrame**

- **phygui - Calicoes dashboard**
  - Configuration manager
  - Operation board
  - Statistics
  - Errors

- **Module X**
  - State = UNDEFINED
    - Initialize
    - Deinitialize
  - State = READY
    - Configure
    - Invalidate
  - State = CONFIGURED
    - Start acquisition
    - Stop acquisition
  - State = ACQUIRING

---

**Pyrame**

- **XML config file with:**
  - general parameters
  - functions' names and types

- **Python interpreter**
  - `def function_Y(param1,...):`
  - `retcode,res = submod.execcmd(function_Y,param1,...)`
  - `submod.setres(retcode,res)`
  - `return`

- **TCP server**
  - Pyrame protocol

- **TCP client**
  - Pyrame protocol

---

F. Magniette, M. Rubio-Roy
Summary

- The WAGASCI electronics has been designed with SPIROC2D.
- Test operation is being performed at LLR and UTokyo.
- Synchronous readout system for neutrino beam is being designed.

Schedule

- The whole DAQ system construction by beginning of 2017.
- Will be ready at spring 2017, after test operation and modification.
Supplemental slides
The WAGASCI detector

- Water tank
- Module

5 Hexagon head screws on each side M8x40 with washers and M8 nuts.

- Clamp
- Tighten screws
**WAGASCI DAQ system**

- **ASU** (Active Sensor Unit)
  
  Readout a 32ch MPPC array with a SPIROC chip.

- **Interface**
  
  Transfer DAQ signals and MPPC bias voltage.

- **DIF** (Detector InterFace)

  Send DAQ signals and SPIROC configurations.

- **GDCC** (Giga Data Concentrator Card)

  Transfer signals between DAQ PC and DIFs.

- **CCC** (Clock & Control Card)

  Provide clock signals and fast control.

<table>
<thead>
<tr>
<th>Modules</th>
<th># of channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Water Module</td>
<td>1280</td>
</tr>
<tr>
<td>CH Module</td>
<td>1280</td>
</tr>
<tr>
<td>SideMRD (right)</td>
<td>88</td>
</tr>
<tr>
<td>SideMRD (left)</td>
<td>88</td>
</tr>
<tr>
<td>Vetos</td>
<td>?</td>
</tr>
</tbody>
</table>
SPIROC DAQ signals

DAQ signals

Main Signals between DAQ and SPIROC
Interface boards

- Interface board
  - 4 ASU chains connection.
  - HV supply connection for all MPPCs via connected ASUs.
  - LV supply connection for DIF and ASUs.

- DIF
  - Send digital signals to all ASUs.
  - Receive raw data from ASUs, and send it to GDCC with header/trailer.
SPIROC2D: Main improvements

- Many modifications
  - Most of them tested in Spiroc2c
    - Individual Tunable Gain LG, HG
    - Crosstalk between HG and LG
    - “Zero event” suppression: CHECKED, OK!
    - Rate dependency: CHECKED, OK!
    - New TDC
    - New Delay cell: CHECKED, 1 ch fired shows same delay as 36 ch fired
    - AutoGain fixed: CHECKED, OK!
  - New External Trigger scheme: CHECKED, OK!
  - Digital part: Timestamp counter 12 → 16 bits: CHECKED, OK!
  - Improved Input DACs (with probe system)
    - Protection added (PAD Diodes + internal 100ohm)
  - Channel to channel uniformity: CHECKED, OK!
  - 4-bit DAC adjustment ch. by ch.: no influence on global threshold: CHECKED, OK
  - Temperature sensor added
  - LVDS receiver boosted for NoTrig/RazChn
SPIROC 2c: TDC improvements

- Modifications on the TDC
  - To decrease dead time during transition => alternation of a rising and a falling ramp implemented
  - Conservative modification but not completely satisfying solution
  - Anyway, a new TDC has to be re-designed in SPIROC 3
SPIROC2D : Linearity of Charge Measurement

LG Within 1% up to 700 pe-

HG Within 1% up to 70 pe-

Qinj Ch 0
Column 3

Cf = 200fF
Ssh = 50ns
Test operation at LLR

Modules

- New: prototype for the WAGASCI electronics.
  - new ASU (with SPIROC2B/2D) ... connection with 36-pin FFC.
  - new Interface board ... transfer of power supply, configuration from DIF, and data from ASU.
  - new DIF ... the firmware is updated to include SPIROC2D control.
**CCC firmware: updated**

**Input**
- SPILL_IN
- RESET_BUTTON
- LOCAL_CLK_50MHZ

**Output**
- CNTL_BUF_DIF_P[8:1]
- CNTL_BUF_DIF_N[8:1]
  *Data to GDCC*

---

**Edge Detect**
- Start: falling
- Stop: rising
- (Busy)

**Trigger Mode Select**
- Beam
- Periodic
- (Both?)

**Synchronizing**

**Encoding**

**Periodic Trigger Generator**
- Synchronous signal to 50MHz CLK

**CCC_TX**

**Buf**
- OBUDFS
  *Serial → 8-bit DS*

**RJ45**
- (to DAQ PC)

**SPILL**
- from LEMO

**50MHz CLK**

**RESET**

**OUTPUT [8:1]**
- to GDCC

*Ethernet connection for trigger mode selection:*
- RBCP?
- or the same as GDCC?

*Pre-beam trigger, arriving 100ms before beam trigger*
- “Pre-beam trigger” is not used as SPILL
- but changes the trigger mode into beam,
  and makes it ready for “beam trigger” for ~100+α ms after this.
Beam trigger timing

**Timing Chart at O**

- Trigger
- Spill # Bit 0
- Spill # Bit 1
- Spill # Bit 2
- Spill # 001
- Spill # 010

Note: there is an offset between spill# in BSD/QSD (beam data) and spill# from LTC

spill# (data) - spill# (LTC output) = 1
### GDCC Packet Format

<table>
<thead>
<tr>
<th>Dst MAC</th>
<th>Src MAC</th>
<th>Ethernet Type</th>
<th>GDCC Type</th>
<th>GDCC_Modifier</th>
<th>GDCC_PktID</th>
<th>GDCC_DataLength</th>
<th>GDCC_Data</th>
<th>PAD</th>
<th>CRC32</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 Bytes</td>
<td>6 Bytes</td>
<td>2 Bytes</td>
<td>2 Bytes</td>
<td>Variable</td>
<td>Variable</td>
<td>Variable</td>
<td>Variable</td>
<td>Pad</td>
<td>4 Bytes</td>
</tr>
</tbody>
</table>

- **Variable** is used for SPILL#.

### DIF data format

<table>
<thead>
<tr>
<th>Section</th>
<th>subsection</th>
<th>field</th>
<th>hex</th>
<th>ascii</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPILL header</td>
<td></td>
<td>Marker</td>
<td>0xFFFFC</td>
<td>&quot;SP&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;ACQid&gt; msb</td>
<td></td>
<td>&quot; &quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;ACQid&gt; lsb</td>
<td></td>
<td>&quot; &quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ascii tag</td>
<td>0x5053</td>
<td>&quot;IP&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ascii tag</td>
<td>0x4C49</td>
<td>&quot; &quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Blank space</td>
<td>0x2020</td>
<td>&quot; &quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot; &quot;</td>
<td></td>
<td>&quot; &quot;</td>
</tr>
<tr>
<td>CHIP header</td>
<td></td>
<td>Marker</td>
<td>0xFFFFD</td>
<td>&quot;CH&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;ID&gt;</td>
<td>0xFF..</td>
<td>&quot; &quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ascii tag</td>
<td>0x4843</td>
<td>&quot; &quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ascii tag</td>
<td>0x5049</td>
<td>&quot; &quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Blank space</td>
<td>0x2020</td>
<td>&quot; &quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot; &quot;</td>
<td></td>
<td>&quot; &quot;</td>
</tr>
<tr>
<td>Raw DATA</td>
<td></td>
<td></td>
<td>binary</td>
<td></td>
</tr>
<tr>
<td>CHIP trailer</td>
<td></td>
<td>Marker</td>
<td>0xFFFFE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;ID&gt;</td>
<td>0xFF..</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Blank space</td>
<td>0x2020</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot; &quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPILL trailer</td>
<td></td>
<td>Marker</td>
<td>0xFFFF</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;ACQid&gt; msb</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;ACQid&gt; lsb</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;nb chip&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;ACQid&gt; msb</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;ACQid&gt; lsb</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Blank space</td>
<td>0x2020</td>
<td></td>
</tr>
</tbody>
</table>

### SPIROC data format

- **0 0 G H** Charge measure Chn 35 (12 bit)
- **0 0 G H** Charge measure Chn 0 (12 bit)
- **0 0 G H** Time measure Chn 35 (12 bit)
- **0 0 G H** Time measure Chn 0 (12 bit)
- **0 0 G H** Gain (1 bit)
- **0 0 G H** Hit (1 bit)
- **0 0 G H** Bunch Crossing ID (12 bit)
- **0 0 G H** Bunch Crossing ID (12 bit)
- **0 0 G H** Bunch Crossing ID (12 bit)
- **0 0 G H** Chip ID (8 bit)
*Off-axis method*
- narrow-band flux
- peak shifted to lower energy

T2K uses $2.5^\circ$ off-axis $\Rightarrow$ peak: $\sim 600\text{MeV}$
- large $\nu_e$ appearance probability
- suppress other interactions than CCQE
First INGRID neutrino event candidate
Nov. 22, 2009
20:25:48 JST

Side view

Top view

Iron (6.5cm thick)
Plastic scintillator (5cm wide, 1cm thick)
Hit in plastic scintillator

MR Run #27, Shot #19655
T2K Spill# 241792
Trip-t Front end Board (TFB)

- 12 layer board (6 signal routing, 6 power/ground)
- 16 cm x 9 cm.
- Each TFB takes 4 Trip-t chips, up to 64 MPPC channels.
- TFB operation is controlled by an FPGA.

Fig. 4. Schematic of one Trip-t front end channel.