

BAND 1:

An example of ALMA modularity

Martin Tourneboeuf

August 24, 2021

1/ Band1 and its friends

2/ Execution

3/ Configuration

4/ Compilation

References

1/ Band1 and its friends

2/ Execution

3/ Configuration

4/ Compilation

References

Disclaimer: software presentation.

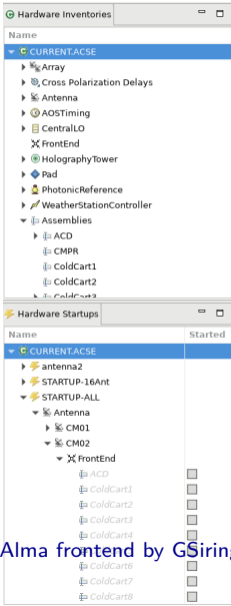
For hardware, see [frontend presentation by GSiringo](#)

Band1

Geo manufacture	Taiwan (ASIAA)
Frecuency	40 GHz = 4 mm = 0.7 K
Field of view	100" (12m antenna)
Resolution (spatial)	0.1" (16km)

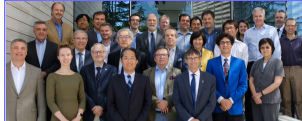


Frontend



Ref: Presentation on Alma frontend by G.Siringo

Friends



OMC

SCIENCE



AHales



MRadiszcz



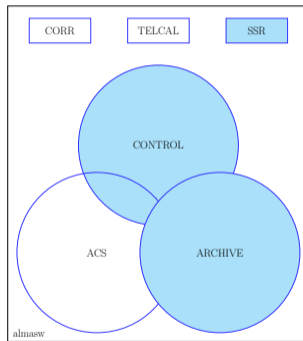
AHirota



HYatagai

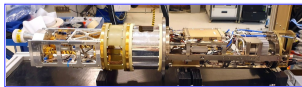


Mtourneb



ENGINEERING

Ref: ICT-18986: Band1 IRM master ticket



GSiringo

Friends



AHales



MRadiszcz



AHirota



HYatai



Mtourneb

- **Subsidiarity** principle
- **Fail fast**: faster with hardware and software operational
- Alma software turns on CONTROL and CORR.

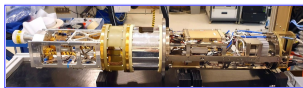
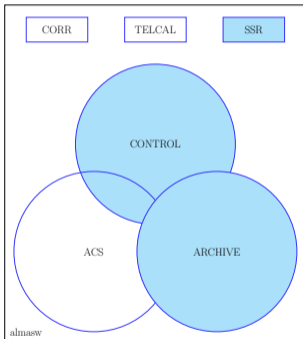
But is **not** in the data channel between antenna receiver and correlator.

<= Not real time, not possible to build IP stack at 40Ghz.

- Vertical modularity

OMC

SCIENCE



GSiringo

Ref: [ICT-18986: Band1 IRM master ticket](#)

1/ Band1 and its friends

2/ Execution

3/ Configuration

4/ Compilation

References

Note: Install `ripgrep`

Turn on B1: Do it your way!

1. Band checkout (runOMC)

Operations Monitoring and Control - almaop@alma

Session View Debug | Devices **Alma** **OpScreen** | Memory RequestQueue

Alma Array1-BLC

Antenna Status Scheduler

Interactive Array1-BLC

Projects P. I. Filter Reset

CSV	UID	Code	Name	Version	Status	State	Executive	P. I.
<input checked="" type="checkbox"/>	uid://A002/xc07e550c15	0000.d.00355.CSV	Band Checkout	0	uid://A002/...	CSVReady	EU	Niel Philips
<input checked="" type="checkbox"/>	uid://A001/x116d465	0000.d.00378.CSV	Band Checkout	0	uid://A001/...	CSVReady	EU	Niel Philips
<input checked="" type="checkbox"/>	uid://A001/xc07e550c15	0000.d.00355.CSV	Band Checkout	0	uid://A001/...	CSVReady	EU	Niel Philips

Showing 54 projects of 2381

SchedBlocks

CSV	UID	Name	State	Status
<input checked="" type="checkbox"/>	uid://A001/x12a162d8	Band 5, B. 0	CSVReady	uid://A001/x12a162d8
<input checked="" type="checkbox"/>	uid://A002/xc07e550c15	Band 1	CSVReady	uid://A002/xc07e550c15

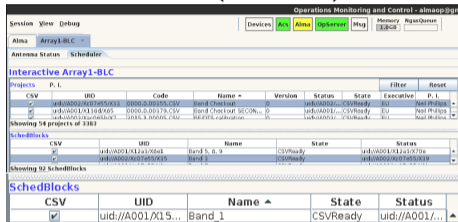
Showing 12 SchedBlocks

SchedBlocks

CSV	UID	Name	State	Status
<input checked="" type="checkbox"/>	uid://A001/x15...	Band_1	CSVReady	uid://A001/...

Turn on B1: Do it your way!

1. Band checkout (runOMC)



Operations Monitoring and Control - almaop@gn

Session View Debug | Devices **Alma** **OpScreen** | Memory RequestQueue

Alma Array1-BLC

Antenna Status Scheduler

Interactive Array1-BLC

Projects	P. I.	Filter	Reset					
CSV	UID	Code	Name	Version	Status	State	Executive	P. I.
<input checked="" type="checkbox"/>	uid://A001/X12121648	0000.A.00355.CSV	Band Checkout	0	uid://A001/...	CSVReady	EU	Ned Phillips
<input checked="" type="checkbox"/>	uid://A001/X11818465	0000.A.00378.CSV	Band Checkout	SEC0...	uid://A001/...	CSVReady	EU	Ned Phillips
<input checked="" type="checkbox"/>	uid://A001/X12040143	0000.A.00690.CSV	RECON Calibration	0	uid://A001/...	CSVReady	EU	Ned Phillips

Showing 54 projects of 2381

SchedBlocks	UID	Name	State	Status
<input checked="" type="checkbox"/>	uid://A001/X12121648	Band 5, B. 0	CSVReady	uid://A001/X12121648
<input checked="" type="checkbox"/>	uid://A002/X1207455035	Band 3	CSVReady	uid://A002/X1207455035

Showing 92 SchedBlocks

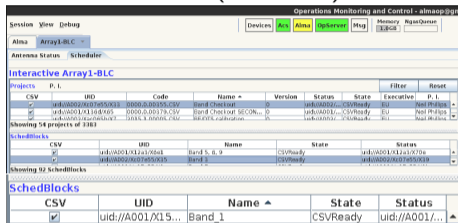
SchedBlocks	CSV	UID	Name	State	Status
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	uid://A001/X15...	Band_1	CSVReady	uid://A001/...

2. Radio setup (bash)

```
radioSetup.py -b 1,3 -y Array1-BLC
```

Turn on B1: Do it your way!

1. Band checkout (runOMC)



Operations Monitoring and Control - almaop@alma

Session View Debug | Devices **Alma** **OpScreen** **Msg** | Memory **Queue**

Alma **Array1-BLC**

Antenna Status Scheduler

Interactive Array1-BLC

Projects	P. I.	Filter	Reset					
CSV	UID	Code	Name	Version	Status	State	Executive	P. I.
<input checked="" type="checkbox"/>	uid://A001/X15/015	0000.0.00355_CSV	Band Checkout	0	uid://A001/...	CSVReady	EU	Neil Pringle
<input checked="" type="checkbox"/>	uid://A001/X15/015	0000.0.00378_CSV	Band Checkout	SEC0...	uid://A001/...	CSVReady	EU	Neil Pringle
<input checked="" type="checkbox"/>	uid://A001/X15/015	0000.0.00379_CSV	Band Checkout	SEC0...	uid://A001/...	CSVReady	EU	Neil Pringle

Showing 54 projects of 2381

SchedBlocks

CSV	UID	Name	State	Status
<input checked="" type="checkbox"/>	uid://A001/X15/015	Band 5, B. 0	CSVReady	uid://A001/X15/015
<input checked="" type="checkbox"/>	uid://A001/X15/015	Band 1	CSVReady	uid://A001/X15/015

Showing 92 SchedBlocks

SchedBlocks

CSV	UID	Name	State	Status
<input checked="" type="checkbox"/>	uid://A001/X15/015	Band 1	CSVReady	uid://A001/X15/015

2. Radio setup (bash)

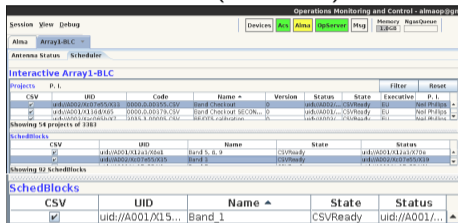
```
radioSetup.py -b 1,3 -y Array1-BLC
```

3. Power up band (startCCL)

```
from CCL.FrontEnd import FrontEnd
FrontEnd('DV25').powerUpBand( \
    FrontEnd.Band.ALMA_RB_03)
```

Turn on B1: Do it your way!

1. Band checkout (runOMC)



Operations Monitoring and Control - almaop@alma

Session View Debug | Devices **Alma** **OpServer** | Memory | RequestQueue

Alma Array1-BLC

Antenna Status Scheduler

Interactive Array1-BLC

Projects	P. I.	Filter	Reset						
CSV	UID	Code	Name	Version	Status	State	Executive	P. I.	
<input checked="" type="checkbox"/>	uid://A001/X15.../0000	0000_0_00355_CSV	Band Checkout	0	uid://A001/...	CSVReady	EU	Neil Pringle	
<input type="checkbox"/>	uid://A001/X15.../0000	0000_0_00378_CSV	Band Checkout	SEC...	0	uid://A001/...	CSVReady	EU	Neil Pringle
<input type="checkbox"/>	uid://A001/X15.../0000	0000_0_00390_CSV	Band Checkout	SEC...	0	uid://A001/...	CSVReady	EU	Neil Pringle

Showing 54 projects of 2381

SchedBlocks

CSV	UID	Name	State	Status
<input checked="" type="checkbox"/>	uid://A001/X15.../0000	Band 5, B. 0	CSVReady	uid://A001/X15.../0000
<input checked="" type="checkbox"/>	uid://A001/X15.../0000	Band 3	CSVReady	uid://A001/X15.../0000

Showing 92 SchedBlocks

SchedBlocks

CSV	UID	Name	State	Status
<input checked="" type="checkbox"/>	uid://A001/X15.../0000	Band 1	CSVReady	uid://A001/X15.../0000

2. Radio setup (bash)

```
radioSetup.py -b 1,3 -y Array1-BLC
```

3. Power up band (startCCL)

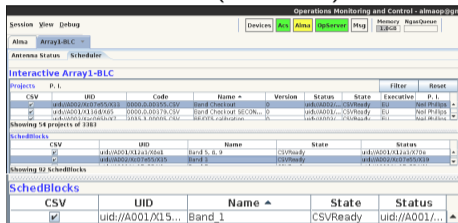
```
from CCL.FrontEnd import FrontEnd
FrontEnd('DV25').powerUpBand( \
    FrontEnd.Band.ALMA_RB_03)
```

4. Turn on device (startCCL)

```
turn_on(ColdCart1('DV25'))
```

Turn on B1: Do it your way!

1. Band checkout (runOMC)



The screenshot shows the 'Interactive Array1-BLC' interface. It displays a table of projects and a table of scheduled blocks. The 'Band Checkout' project is highlighted in blue.

Projects	P. I.	Filter	Reset					
CSV	UID	Code	Name	Version	Status	State	Executive	P. I.
<input checked="" type="checkbox"/>	uid://A002/xc07e5035	0000.d.00355_CSV	Band Checkout	0	uid://A002/...	CSVReady	EU	Ned Phillips
<input type="checkbox"/>	uid://A001/X116485	0000.d.00378_CSV	Band Checkout	SEC0...	uid://A001/...	CSVReady	EU	Ned Phillips
<input type="checkbox"/>	uid://A001/xc07e5035	0000.d.00355_CSV	Band Checkout	0	uid://A001/...	CSVReady	EU	Ned Phillips

SchedBlocks	UID	Name	State	Status
<input checked="" type="checkbox"/>	uid://A001/X12416248	Band 5, B. 9	CSVReady	uid://A001/X12416248
<input checked="" type="checkbox"/>	uid://A002/xc07e5035	Band 3	CSVReady	uid://A002/xc07e5035

SchedBlocks	CSV	UID	Name	State	Status
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	uid://A001/X15...	Band 1	CSVReady	uid://A001/...

2. Radio setup (bash)

```
radioSetup.py -b 1,3 -y Array1-BLC
```

3. Power up band (startCCL)

```
from CCL.FrontEnd import FrontEnd
FrontEnd('DV25').powerUpBand( \
    FrontEnd.Band.ALMA_RB_03)
```

4. Turn on device (startCCL)

```
turn_on(ColdCart1('DV25'))
```

5. Start hardware (startCCL)

```
cc = ColdCart1('DV25')
cc.getHwState()
cc.hwConfigure()
cc.hwInitialize()
cc.hwOperational()
```

Ref: `vim ~/AlmaSw/CONTROL/Common/ControlDevice/src/CCL/HardwareDevice.py` # or
from `CCL.HardwareDevice` import `HardwareDevice`;

Communicate with B1: Do it your way!

1. Run Observation (runOMC)

The screenshot shows the Scheduler interface for a Dynamic Array18-BLC. It is divided into three main sections: Projects, Pending Executions, and Schedules.

Projects: A table listing various projects with columns for CSV, UID, Code, Name, Version, Status, Start, and End. The table shows 52 projects out of 3870.

CSV	UID	Code	Name	Version	Status	Start	End	
✓	uid.0	0000-0.0011-E-CV	Obs mode test 0	uid.0	C.S.	00	00	
✓	uid.0	0000-0.0018-E-CV	24 AmpRate 0.25	uid.0	C.S.	00	00	
✓	uid.0	0000-0.00171-CV	24 Amplitude 0.3	uid.0	C.S.	00	00	
✓	uid.0	0000-0.0018-E-CV	Focus test	uid.0	C.S.	00	00	
✓	uid.0	0000-0.0020E-CV	Focus 981 A... 0	uid.0	C.S.	00	00	
✓	uid.0	0000-0.00201-CV	Focus 981 BL... 0	uid.0	C.S.	00	00	
✓	uid.0	0000-0.0021-E-CV	Grid Survey	uid.0	C.S.	00	00	
✓	uid.0	0000-0.00222-CV	Jetstream plot	uid.0	C.S.	00	00	
✓	uid.0	0000-0.00229-CV	Weak sources	uid.0	C.S.	00	00	
✓	uid.0	0000-0.00238-CV	Ge-Noble	0.3	uid.0	C.S.	00	00
✓	uid.0	0000-0.00245-CV	Focus 981 BL... 0	uid.0	C.S.	00	00	
✓	uid.0	0000-0.00271-CV	Focus 981	uid.0	C.S.	00	00	

Schedules: A table listing schedulables with columns for CSV, UID, Name, State, and Status. It shows 20 schedulables out of 3870.

CSV	UID	Name	State	Status
✓	uid.0001805	Focus_Band_1_2	C.VReady	uid.0001805
✓	uid.0001805	Focus_Band_1_2_V_X	C.VReady	uid.0001805
✓	uid.0002209a	Focus_Band_10_2_V_X	C.VReady	uid.0002209a
✓	uid.0001805	Focus_Band_2_2	C.VReady	uid.0001805
✓	uid.0001805	Focus_Band_2_2_V_X	C.VReady	uid.0001805
✓	uid.0001805	Focus_Band_2_2_V_Y	C.VReady	uid.0001805
✓	uid.0002209b	Focus_Band_2_2	C.VReady	uid.0002209b
✓	uid.0002209a	Focus_Band_2_V_Y	C.VReady	uid.0002209a
✓	uid.0002209a	Focus_Band_2_V_X	C.VReady	uid.0002209a
✓	uid.0002209a	Focus_Band_3_2	C.VReady	uid.0002209a
✓	uid.0002209a	Focus_Band_3_V_X	C.VRunning	uid.0002209a

Pending Executions: A section for managing pending executions, including a table for UID, P, L, Time, Exec, Name, and State. It includes buttons for Process, Destroy, and Release.

Current Executions: A section for managing current executions, including a table for UID, P, L, NA, R, S, I, J. It includes buttons for Step SB, Full Auto, Start Queue, Stop Queue, and Destroy Array.

Completed Executions: A section for managing completed executions, including a table for UID, P, L, Time, Exec, Name, and State. It includes buttons for Start Queue, Stop Queue, and Destroy Array.

Communicate with B1: Do it your way!

1. Run Observation (runOMC)

The Scheduler interface displays two tables: 'Projects' and 'Schedules'.

Projects Table:

Proj	UID	Code	Name	Version	Status	Start	Exp.
✓	uid.0	00000.0.0011.E	CEC mode test.0	uid.0	CS	DU	PL
✓	uid.0	00000.0.0018.F	24 Amp/Rate.0	uid.0	CS	DU	PL
✓	uid.0	00000.0.0017.I	24 Amplitude.0.3	uid.0	CS	DU	PL
✓	uid.0	00000.0.0020.H	Focus test.0	uid.0	CS	DU	PL
✓	uid.0	00000.0.0022.C	Focus 981 A. 0	uid.0	CS	DU	PL
✓	uid.0	00000.0.0021.B	Focus 981 BL. 0	uid.0	CS	DU	PL
✓	uid.0	00000.0.0021.E	Grid Setup. 0	uid.0	CS	DU	PL
✓	uid.0	00000.0.0022.Z	Antenna pos. 0	uid.0	CS	DU	PL
✓	uid.0	00000.0.0023.V	Weak source. 0	uid.0	CS	DU	PL
✓	uid.0	00000.0.0023.H	Ge-Noise. 0.3	uid.0	CS	DU	PL
✓	uid.0	00000.0.0023.S	Focus 981 BL. 0	uid.0	CS	DU	PL
✓	uid.0	00000.0.0021.F	Focus 981. 0	uid.0	CS	DU	PL

Showing 52 projects of 3870

Schedules Table:

Proj	UID	Name	State	Status
✓	uid.W001R05	Focus_Band_1_2	CSReady	uid.W001R05
✓	uid.W001R05	Focus_Band_1_2_V_X	CSReady	uid.W001R05
✓	uid.W002R06	Focus_Band_10_2_V_X	CSReady	uid.W002R06
✓	uid.W001R05	Focus_Band_2_2	CSReady	uid.W001R05
✓	uid.W001R05	Focus_Band_2_V_X	CSReady	uid.W001R05
✓	uid.W001R05	Focus_Band_2_2_V_X	CSReady	uid.W001R05
✓	uid.W002R06	Focus_Band_2_2	CSReady	uid.W002R06
✓	uid.W002R06	Focus_Band_2_V_X	CSReady	uid.W002R06
✓	uid.W002R06	Focus_Band_2_2_V_X	CSReady	uid.W002R06
✓	uid.W002R06	Focus_Band_5_2	CSReady	uid.W002R06
✓	uid.W002R06	Focus_Band_5_V_X	CSRunning	uid.W002R06

Showing 28 Schedules

The Scheduler interface includes execution control buttons and logs:

- Pending Executions:** Project: uid.0, UID: 0018.F, Name: 24 Amp/Rate.0, State: Pending. Buttons: Proceed, Denote, Release.
- Current Executions:** Project: uid.0, UID: 0023.H, Name: Ge-Noise.0.3, State: Running. Buttons: Stop SB, Full Auto, Start Queue, Stop Queue, Destroy Array.
- Completed Executions:** Project: uid.0, UID: 0018.F, Name: 24 Amp/Rate.0, State: Completed. Buttons: OK, Cancel.

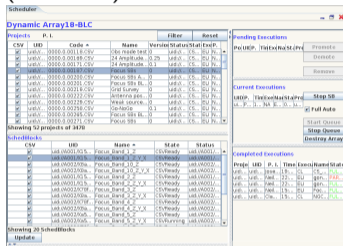
2. Click M&C point (objexp)

The M&C point interface shows the following details:

- By type / By device:** A tree view showing the hierarchy of objects, with 'CONTROL_0303FrontEndColdCart1' selected.
- Object:** CONTROL_0303FrontEndColdCart1
- Operations:** A list of operations for the selected object, including GET_MAGNETIUM_CURR_SIGN, GET_MAJOR_REV_LEVEL, GET_MINOR_REV_LEVEL, GET_MGR_TEMP1, GET_MGR_TEMP2, GET_MGR_TEMP3, GET_MGR_TEMP4, GET_MGR_TEMP5, GET_MGR_TEMP6, GET_MGR_TEMP7, GET_MGR_TEMP8, GET_MGR_TEMP9, GET_MGR_TEMP10, GET_MGR_TEMP11, GET_MGR_TEMP12, GET_MGR_TEMP13, GET_MGR_TEMP14, GET_MGR_TEMP15, GET_MGR_TEMP16, GET_MGR_TEMP17, GET_MGR_TEMP18, GET_MGR_TEMP19, GET_MGR_TEMP20, GET_MGR_TEMP21, GET_MGR_TEMP22, GET_MGR_TEMP23, GET_MGR_TEMP24, GET_MGR_TEMP25, GET_MGR_TEMP26, GET_MGR_TEMP27, GET_MGR_TEMP28, GET_MGR_TEMP29, GET_MGR_TEMP30, GET_MGR_TEMP31, GET_MGR_TEMP32, GET_MGR_TEMP33, GET_MGR_TEMP34, GET_MGR_TEMP35, GET_MGR_TEMP36, GET_MGR_TEMP37, GET_MGR_TEMP38, GET_MGR_TEMP39, GET_MGR_TEMP40, GET_MGR_TEMP41, GET_MGR_TEMP42, GET_MGR_TEMP43, GET_MGR_TEMP44, GET_MGR_TEMP45, GET_MGR_TEMP46, GET_MGR_TEMP47, GET_MGR_TEMP48, GET_MGR_TEMP49, GET_MGR_TEMP50, GET_MGR_TEMP51, GET_MGR_TEMP52, GET_MGR_TEMP53, GET_MGR_TEMP54, GET_MGR_TEMP55, GET_MGR_TEMP56, GET_MGR_TEMP57, GET_MGR_TEMP58, GET_MGR_TEMP59, GET_MGR_TEMP60, GET_MGR_TEMP61, GET_MGR_TEMP62, GET_MGR_TEMP63, GET_MGR_TEMP64, GET_MGR_TEMP65, GET_MGR_TEMP66, GET_MGR_TEMP67, GET_MGR_TEMP68, GET_MGR_TEMP69, GET_MGR_TEMP70, GET_MGR_TEMP71, GET_MGR_TEMP72, GET_MGR_TEMP73, GET_MGR_TEMP74, GET_MGR_TEMP75, GET_MGR_TEMP76, GET_MGR_TEMP77, GET_MGR_TEMP78, GET_MGR_TEMP79, GET_MGR_TEMP80, GET_MGR_TEMP81, GET_MGR_TEMP82, GET_MGR_TEMP83, GET_MGR_TEMP84, GET_MGR_TEMP85, GET_MGR_TEMP86, GET_MGR_TEMP87, GET_MGR_TEMP88, GET_MGR_TEMP89, GET_MGR_TEMP90, GET_MGR_TEMP91, GET_MGR_TEMP92, GET_MGR_TEMP93, GET_MGR_TEMP94, GET_MGR_TEMP95, GET_MGR_TEMP96, GET_MGR_TEMP97, GET_MGR_TEMP98, GET_MGR_TEMP99, GET_MGR_TEMP100.
- Message Log:** A log of messages, including 'Message: Starting engine initialization...', 'Message: Obtained reference to 'Temperature'', 'Message: Obtained reference to 'Magnetar'', 'Message: Obtaining root nodes.', and 'Message: Infoterm DAC engine. Please wait...'. The log also shows a timestamp: [01 : 2023-05-24T13:20:30.002].

Communicate with B1: Do it your way!

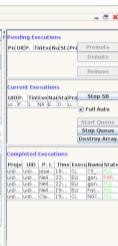
1. Run Observation (runOMC)



The Scheduler interface displays a table of projects for 'Dynamic Array18-BLC'. The table has columns for Project ID, UID, Code, Name, Version, Status, Start, and End. The 'Status' column shows various states like 'Ready', 'Running', and 'Error'. Below the table, there are sections for 'SchedBlocks' and 'Update'.

Project	UID	Code	Name	Version	Status	Start	End
proj_0000000001	00000001	1E	06 module test	0	Ready	05.05.2023	05.05.2023
proj_0000000002	00000002	1E	24 Amp/Rate	0	Ready	05.05.2023	05.05.2023
proj_0000000003	00000003	1E	24 Amplitude	0	Ready	05.05.2023	05.05.2023
proj_0000000004	00000004	1E	Focus 181 A	0	Ready	05.05.2023	05.05.2023
proj_0000000005	00000005	1E	Focus 181 B	0	Ready	05.05.2023	05.05.2023
proj_0000000006	00000006	1E	Grid Setup	0	Ready	05.05.2023	05.05.2023
proj_0000000007	00000007	1E	Antenna pos	0	Ready	05.05.2023	05.05.2023
proj_0000000008	00000008	1E	Weak source	0	Ready	05.05.2023	05.05.2023
proj_0000000009	00000009	1E	Ge-Noble	0	Ready	05.05.2023	05.05.2023
proj_0000000010	00000010	1E	Focus 181 BL	0	Ready	05.05.2023	05.05.2023
proj_0000000011	00000011	1E	Focus 181	0	Ready	05.05.2023	05.05.2023

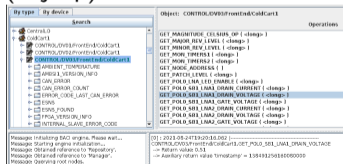
3. Call device control method (startCCL) ColdCart1('DV25').GET_POLO_SB1_LNA1_DRAIN_VOLTAGE()



The Pending Executions panel shows a table with columns for Project ID, UID, P. L., Time, Exec, Name, and State. The Completed Executions panel shows a table with columns for Project ID, P. L., Time, Exec, Name, and State.

Project	UID	P. L.	Time	Exec	Name	State
proj_0000000001	00000001	1E	05.05.2023	05.05.2023	06 module test	Ready
proj_0000000002	00000002	1E	05.05.2023	05.05.2023	24 Amp/Rate	Ready
proj_0000000003	00000003	1E	05.05.2023	05.05.2023	24 Amplitude	Ready
proj_0000000004	00000004	1E	05.05.2023	05.05.2023	Focus 181 A	Ready
proj_0000000005	00000005	1E	05.05.2023	05.05.2023	Focus 181 B	Ready
proj_0000000006	00000006	1E	05.05.2023	05.05.2023	Grid Setup	Ready
proj_0000000007	00000007	1E	05.05.2023	05.05.2023	Antenna pos	Ready
proj_0000000008	00000008	1E	05.05.2023	05.05.2023	Weak source	Ready
proj_0000000009	00000009	1E	05.05.2023	05.05.2023	Ge-Noble	Ready
proj_0000000010	00000010	1E	05.05.2023	05.05.2023	Focus 181 BL	Ready
proj_0000000011	00000011	1E	05.05.2023	05.05.2023	Focus 181	Ready

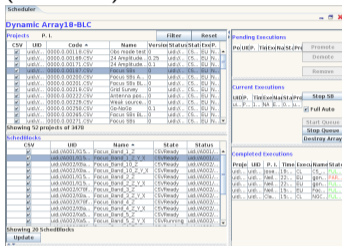
2. Click M&C point (objexp)



The Object Explorer interface shows a tree view of objects. The 'CONTROL_0000000001' object is selected, and its operations are listed in the right pane. The operations include GET_MAGNITUDE, GET_MAJOR_REV_LEVEL, GET_MINOR_REV_LEVEL, GET_MOR_TEMP1, GET_MOR_TEMP2, GET_MOR_TEMP3, GET_MOR_TEMP4, GET_MOR_TEMP5, GET_MOR_TEMP6, GET_MOR_TEMP7, GET_MOR_TEMP8, GET_MOR_TEMP9, GET_MOR_TEMP10, GET_MOR_TEMP11, GET_MOR_TEMP12, GET_MOR_TEMP13, GET_MOR_TEMP14, GET_MOR_TEMP15, GET_MOR_TEMP16, GET_MOR_TEMP17, GET_MOR_TEMP18, GET_MOR_TEMP19, GET_MOR_TEMP20, GET_MOR_TEMP21, GET_MOR_TEMP22, GET_MOR_TEMP23, GET_MOR_TEMP24, GET_MOR_TEMP25, GET_MOR_TEMP26, GET_MOR_TEMP27, GET_MOR_TEMP28, GET_MOR_TEMP29, GET_MOR_TEMP30, GET_MOR_TEMP31, GET_MOR_TEMP32, GET_MOR_TEMP33, GET_MOR_TEMP34, GET_MOR_TEMP35, GET_MOR_TEMP36, GET_MOR_TEMP37, GET_MOR_TEMP38, GET_MOR_TEMP39, GET_MOR_TEMP40, GET_MOR_TEMP41, GET_MOR_TEMP42, GET_MOR_TEMP43, GET_MOR_TEMP44, GET_MOR_TEMP45, GET_MOR_TEMP46, GET_MOR_TEMP47, GET_MOR_TEMP48, GET_MOR_TEMP49, GET_MOR_TEMP50, GET_MOR_TEMP51, GET_MOR_TEMP52, GET_MOR_TEMP53, GET_MOR_TEMP54, GET_MOR_TEMP55, GET_MOR_TEMP56, GET_MOR_TEMP57, GET_MOR_TEMP58, GET_MOR_TEMP59, GET_MOR_TEMP60, GET_MOR_TEMP61, GET_MOR_TEMP62, GET_MOR_TEMP63, GET_MOR_TEMP64, GET_MOR_TEMP65, GET_MOR_TEMP66, GET_MOR_TEMP67, GET_MOR_TEMP68, GET_MOR_TEMP69, GET_MOR_TEMP70, GET_MOR_TEMP71, GET_MOR_TEMP72, GET_MOR_TEMP73, GET_MOR_TEMP74, GET_MOR_TEMP75, GET_MOR_TEMP76, GET_MOR_TEMP77, GET_MOR_TEMP78, GET_MOR_TEMP79, GET_MOR_TEMP80, GET_MOR_TEMP81, GET_MOR_TEMP82, GET_MOR_TEMP83, GET_MOR_TEMP84, GET_MOR_TEMP85, GET_MOR_TEMP86, GET_MOR_TEMP87, GET_MOR_TEMP88, GET_MOR_TEMP89, GET_MOR_TEMP90, GET_MOR_TEMP91, GET_MOR_TEMP92, GET_MOR_TEMP93, GET_MOR_TEMP94, GET_MOR_TEMP95, GET_MOR_TEMP96, GET_MOR_TEMP97, GET_MOR_TEMP98, GET_MOR_TEMP99, GET_MOR_TEMP100.

Communicate with B1: Do it your way!

1. Run Observation (runOMC)



The 'Scheduler' window displays two tables: 'Dynamic Array18-BLC' and 'SchedBlocks'. The 'Dynamic Array18-BLC' table lists projects with columns for Project, P. I., Code, Name, Version, Status, Start, End, and Exp. The 'SchedBlocks' table lists tasks with columns for CSW, UID, Name, State, and Status.

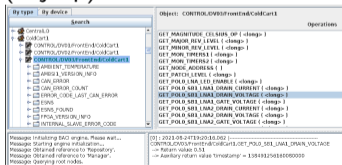
3. Call device control method (startCCL)

```
ColdCart1('DV25').GET_POLO_SB1_LNA1_DRAIN_VOLTAGE()
```

4. Send CAN command from ABM (startCCL)

```
from CCL.AmbManager import *; import struct  
# Monitor: channel, device, RCA -> (data, time)  
dt = AmbManager("DV25").monitor(1, 0x13, 0x40)  
print(struct.unpack('8B', dt[0])) # len <= 8
```

2. Click M&C point (objexp)



The 'By type' window shows a list of objects under 'By device' with a search bar. The selected object is 'CONTROL_OV933FrontEndColdCart1'. Below the list, a log window displays messages and a list of operations.

Operations:

- GET_MAGNITUDE_CSL_SAMP_OF (-cloop-)
- GET_MAJOR_REV_LEVEL (-cloop-)
- GET_MINOR_REV_LEVEL (-cloop-)
- GET_MON_FREQHz (-cloop-)
- GET_MON_TEMPHz (-cloop-)
- GET_ADDR6_ADDRESS (-cloop-)
- GET_PATH_LEVEL (-cloop-)
- GET_PDR_1_LED_ENABLE (-cloop-)
- GET_POLO_SB1_LNA1_DRAIN_CURRENT (-cloop-)
- GET_POLO_SB1_LNA1_DRAIN_VOLTAGE (-cloop-)
- GET_POLO_SB1_LNA1_GATE_VOLTAGE (-cloop-)
- GET_POLO_SB1_LNA2_DRAIN_CURRENT (-cloop-)
- GET_POLO_SB1_LNA2_DRAIN_VOLTAGE (-cloop-)
- GET_POLO_SB1_LNA2_GATE_VOLTAGE (-cloop-)

Log messages:

```
[0]: 2023-05-24T12:20:30.002 |  
Message: Starting engine initialization...  
Message: Obtained reference to 'Topology'.  
Message: Obtained reference to 'Manager'.  
Message: Defining root nodes.
```

1/ Band1 and its friends

2/ Execution

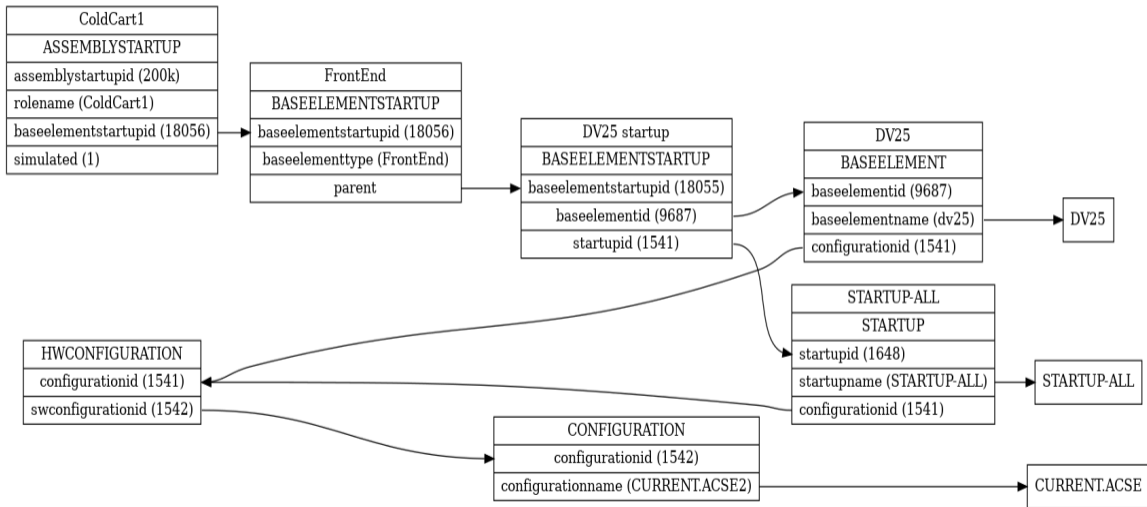
3/ Configuration

4/ Compilation

References

Alias: alias fr='find . | rg '
alias fr='find . | grep -r '

Device: TMCDB database model



Ref: [How to add Band1 to database \(TMCDB\)](#)

Device: Failure: CAN configuration

Creating node 0x13, s/n 0x65b0e5a10002b5dd, ColdCart1 device

[CONTROL/DV03/cppContainer-GL -] Switched state of component CONTROL/DV03/FrontEnd/ColdCart1: NEW -> INITIALIZING

[CONTROL/DV03/cppContainer-GL - virtual void AmbDeviceImpl::initialize()] Life cycle error (type=28, code=0)

UserErrorMessage="Cannot get, from the TMCDB, the CAN channel and node number needed by the CONTROL/DV03/FrontEnd/ColdCart1"

[TMCDBAccess - getDefaultCanAddress] TMCDB error (type=100000, code=0) javaex.class="alma.TmcdErrType.wrappers.AcsJTmcdbAccess"

UserErrorMessage="No default CAN address found for CONTROL/DV03/FrontEnd/ColdCart1"

[maci::LibraryManager - maci::LibraryManager::unload]

Unloaded '/alma/ACS-2021AUG/ACSSW/lib/libColdCart1CompSimImpl.so'.

Device: Failure: CAN configuration

Creating node 0x13, s/n 0x65b0e5a10002b5dd, ColdCart1 device

[CONTROL/DV03/cppContainer-GL -] Switched state of component CONTROL/DV03/FrontEnd/ColdCart1: NEW -> INITIALIZING

[CONTROL/DV03/cppContainer-GL - virtual void AmbDeviceImpl::initialize()] Life cycle error (type=28, code=0)

UserErrorMessage="Cannot get, from the TMCDB, the CAN channel and node number needed by the CONTROL/DV03/FrontEnd/Co

[TMCDBAccess - getDefaultCanAddress] TMCDB error (type=100000, code=0) javaex.class="alma.TmcdErrType.wrappers.AcsJTMc

UserErrorMessage="No default CAN address found for CONTROL/DV03/FrontEnd/ColdCart1"

[maci::LibraryManager - maci::LibraryManager::unload]

Unloaded '/alma/ACS-2021AUG/ACSSW/lib/libColdCart1CompSimImpl.so'.

The screenshot shows the ALMA TMCDB Hardware Explorer interface. The main window displays a table of deployed components for the CAN/Ethernet Editor. The table has four columns: Path, Name, Node Dec., and Node Hex. The component 'ColdCart3' is highlighted in blue, corresponding to the error message above, which mentions node 0x13 and s/n 0x65b0e5a10002b5dd.

Path	Name	Node Dec.	Node Hex
CONTROL/DV03/FrontEnd	ACD	40	0x28
CONTROL/DV03/FrontEnd	ColdCart3	19	0x13
CONTROL/DV03/FrontEnd	ColdCart4	19	0x13
CONTROL/DV03/FrontEnd	ColdCart5	19	0x13
CONTROL/DV03/FrontEnd	ColdCart6	19	0x13
CONTROL/DV03/FrontEnd	ColdCart7	19	0x13
CONTROL/DV03/FrontEnd	ColdCart8	19	0x13
CONTROL/DV03/FrontEnd	ColdCart9	19	0x13

Device: Failure: CAN configuration

Creating node 0x13, s/n 0x65b0e5a10002b5dd, ColdCart1 device

[CONTROL/DV03/cppContainer-GL -] Switched state of component CONTROL/DV03/FrontEnd/ColdCart1: NEW -> INITIALIZING

[CONTROL/DV03/cppContainer-GL - virtual void AmbDeviceImpl::initialize()] Life cycle error (type=28, code=0)

UserErrorMessage="Cannot get, from the TMCDB, the CAN channel and node number needed by the CONTROL/DV03/FrontEnd/Co

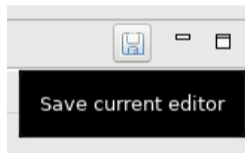
[TMCDBAccess - getDefaultCanAddress] TMCDB error (type=100000, code=0) javaex.class="alma.TmcdErrType.wrappers.AcsJTMc

UserErrorMessage="No default CAN address found for CONTROL/DV03/FrontEnd/ColdCart1"

[maci::LibraryManager - maci::LibraryManager::unload]

Unloaded '/alma/ACS-2021AUG/ACSSW/lib/libColdCart1CompSimImpl.so'.

Path	Name	Node Dec.	Node Hex
CONTROL/DV03/FrontEnd	ACD	40	0x28
CONTROL/DV03/FrontEnd	ColdCart3	19	0x13
CONTROL/DV03/FrontEnd	ColdCart4	19	0x13
CONTROL/DV03/FrontEnd	ColdCart5	19	0x13
CONTROL/DV03/FrontEnd	ColdCart6	19	0x13
CONTROL/DV03/FrontEnd	ColdCart7	19	0x13
CONTROL/DV03/FrontEnd	ColdCart8	19	0x13
CONTROL/DV03/FrontEnd	ColdCart9	19	0x13



	CONTROL/DV03/FrontEnd	WLA8	19	0x13	1	True	not set	-1	not set	-1	-1.0	-1
	CONTROL/DV03/FrontEnd	WCA9	19	0x13	1	false	not set	-1	not set	-1	-1.0	-1
*	CONTROL/DV03/FrontEnd	ColdCart1	19	0x13	1	false	not set	-1	not set	-1	-1.0	-1
*	CONTROL/DV03/FrontEnd	WCA1	19	0x13	-1	false	not set	-1	not set	-1	-1.0	-1
+	CONTROL/DV03/FrontEnd	PowerDist1	0	0x0	-1	false	not set	-1	not set	-1	-1.0	-1

Configuration point (assembly): Files

file: /alma/ste/config/TMCDB_DATA/{101,113186802193399896}.xml

```
<?xml version='1.0' encoding='UTF-8'?>
```

```
<ConfigData>
```

```
  <ASSEMBLY value="ColdCart1"/>
```

```
  <!-- <CCAConfig value="0000" timestamp="2020-01-01T00:00:00"/> -->
```

```
  <!-- <ESN value="000000000000000000"/> -->
```

```
  <!-- <SN value="CCA1-01"/> -->
```

```
  <!-- <TempSensorOffsets Te0="0.00" Te1="0.00" Te2="0.00" Te3="0.00" -->
```

```
  <PreampParamsPol0Sb1 FreqL0="31.00E9"
```

```
    VD1="0.70" VD2="0.70" VD3="1.65" <!-- VD4="1.40" VD5="1.50" -->
```

```
    ID1="3.04" ID2="4.04" ID3="7.02" <!-- ID4="7.40" ID5="7.50" -->
```

```
    VG1="0.09" VG2="0.10" VG3="-0.20" <!-- VG4="0.40" VG5="0.50" -->/>
```

```
  <PreampParamsPol1Sb1 FreqL0="31.00E9"
```

```
    VD1="0.67" VD2="0.89" VD3="1.48" <!-- VD4="1.40" VD5="1.50" -->
```

```
    ID1="3.07" ID2="3.58" ID3="7.07" <!-- ID4="7.40" ID5="7.50" -->
```

```
    VG1="0.14" VG2="0.12" VG3="-0.17" <!-- VG4="0.40" VG5="0.50" -->/>
```

```
</ConfigData>
```

Ref: C9: ICT-17809: Track assembly changes Ref: B1:
ICT-18566: Update FrontEnd software <- ICD" commit
1e2863e31ed

Configuration point (assembly): Failure

- **Game over**

```
"ColdCart1" is not appearing in the logs  
The requested Component had not been already activated at request time  
The FrontEnd is in Shutdown. No operations allowed.
```

- **Bad assembly**

```
SerialNumber for device CONTROL/DA65/DTXBBpr0 has been  
set to 0x734745ba0b87df55. # This comes at turn on CONTROL  
Cannot get, from the TMCDB, the assembly data for  
S/N: 734745ba0b87df55 # Warning -> Error
```

- **Bad baci**

```
Failed to read static data for 'CONTROL/DV03/FrontEnd/ColdCart7:POL1_SB2_LNA_ENA
```

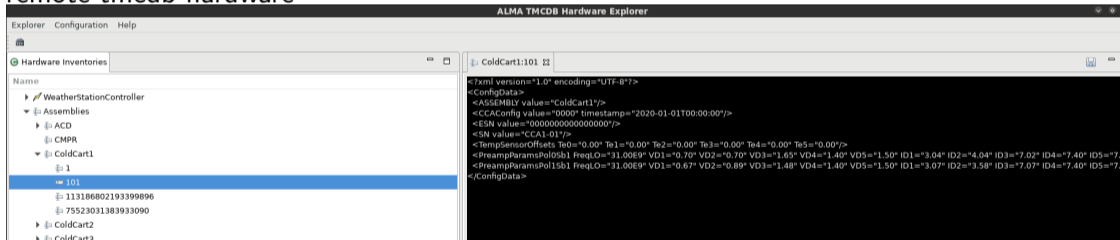
- **Harmless baci**

```
Field alma/CONTROL/DV03/FrontEnd/Cryostat/TCPIP_ADDRESS/  
{archive_delta_percent,alarm_on,description} does not exist
```


Configuration point (assembly): Locations

Those xml configuration files are in:

1. remote git repository “configurations”
2. local directory /alma/ste/config/TMCDB_DATA
3. remote tmcdb-hardware



From git (1) to sql (3), run:

```
updateAssemblies -b C9 -f -v
```

Ref: [How to add/update an xml assembly file](#)
Ref: [Script to convert assembly files: C7 -> C9](#)

Monitor or control point (BACI)

```
# Push BACI: local build files -> remote database sql  
# Or: -component CONTROL/DV25/FrontEnd/ColdCart1  
MonitoringSyncTool -vv -c \  
  -component_type IDL:alma/Control/ColdCart1:1.0 \  
  -configuration CURRENT.ACSE2 \  
  -logfile "$HOME/Test/log_ColdCart1_$(date).log"
```

1/ Band1 and its friends

2/ Execution

3/ Configuration

4/ Compilation

References

Cmd: `rsync -av ~mtourneb/.local/ ~/.local`

Cmd: `echo 'PATH=$HOME/.local/bin:$PATH' >> ~/.bashrc`

Note: No Double Side Band en band1

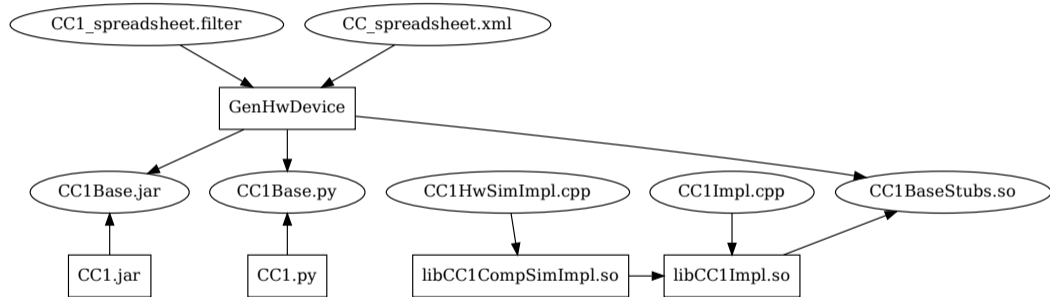
Source code: list of BACI

	A	B	C	D	H	Q	U
21	Assembly	Name	RCA	Raw Data Type	Data Units	Mode	Description
22	ColdCart	POL1_SB1_SIS_MAGNET_CURRENT	0x00430	float	ampere	startup	Get the SIS magnet current for Pol1 SB1
23	ColdCart	POL1_SB2_SIS_MAGNET_CURRENT	0x004B0	float	ampere	startup	Get the SIS magnet current for Pol1 SB2
24	ColdCart	POL0_SB1_LNA1_DRAIN_VOLTAGE	0x00040	float	volt	operational	Get the LNA St1 drain voltage for Pol0 SB1
25	ColdCart	POL0_SB2_LNA1_DRAIN_VOLTAGE	0x000C0	float	volt	operational	Get the LNA St1 drain voltage for Pol0 SB2
26	ColdCart	POL1_SB1_LNA1_DRAIN_VOLTAGE	0x00440	float	volt	operational	Get the LNA St1 drain voltage for Pol1 SB1

libreoffice ~/AlmaSw/CONTROL/Device/HardwareDevice/FrontEnd/FrontEnd/idl/ColdCart_spreadsheet.xml

Ref: [ICT-18566: Update FrontEnd software <- ICD" commit 1e2863e31ed](#)

From Source to library



1/ Band1 and its friends

2/ Execution

3/ Configuration

4/ Compilation

References

References

- [Jira ICT-18986](#): C9: Band1 first use in production: IRM master ticket to track related projects (database, science ..)
- [Review: First Science 2009](#)
- [Image: News from Japan](#)
- [Report: Alma 2030](#)
- [Images: Alma band1](#)