DMON and the AGGA-4

Michael Ryan

CTO

O.C.E. Technology

Once upon a time…

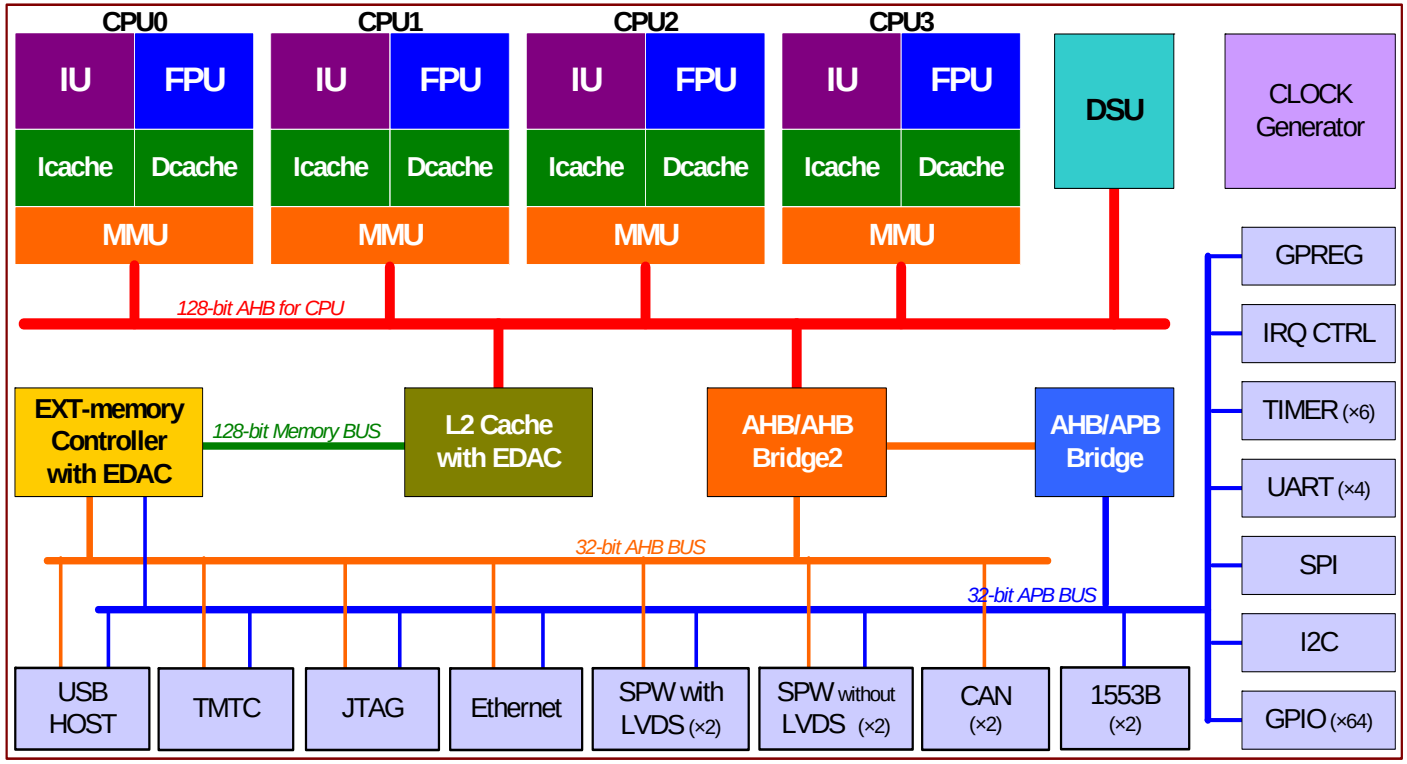
Embedded System:

Microprocessor	*chip*
Interface	*chips*
RAM	*chips*
EPROM	*chip*

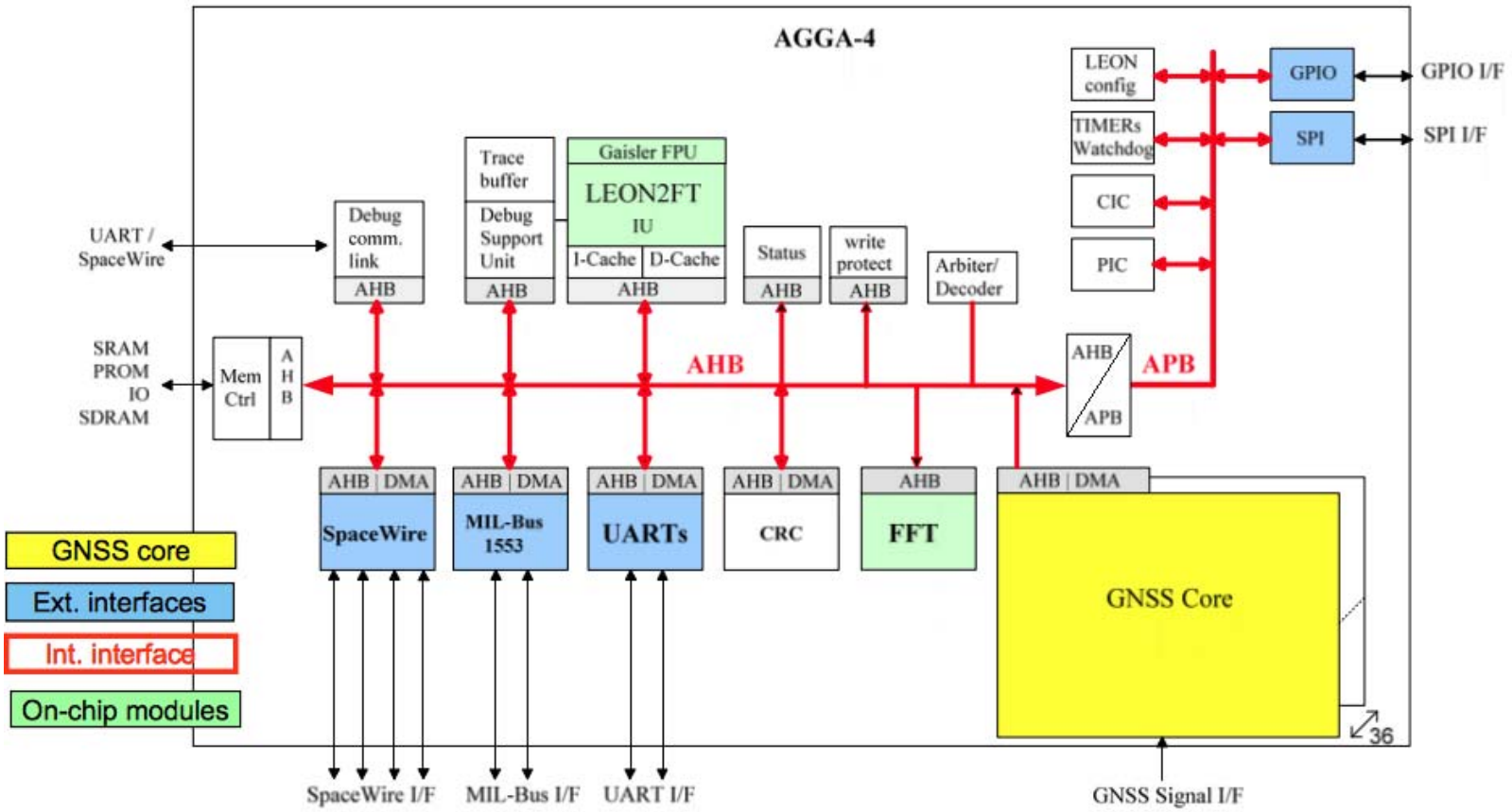
- *All separate chips -> you could get at them*
  - In-circuit emulator (remember Tektronix 8002A?)
  - Logic analyser
  - Oscilloscope

But today …

e.g. OCE's E698PM: 4 CPUs, 34 other units, 1 chip



But today…


AGGA-4
LEON config
GPIO
GPIO I/F
TIMERs Watchdog
SPI
SPI I/F
CIC
PIC
Trace buffer
Debug Support Unit
Gaisler FPU
LEON2FT
IU
I-Cache
D-Cache
AHB
Debug comm. link
Status
write protect
Arbiter/ Decoder
UART / SpaceWire
SRAM
PROM
IO
SDRAM
Mem Ctrl
AHB
APB
SpaceWire
MIL-Bus 1553
UARTs
CRC
FFT
GNSS Core
36
GNSS core
Ext. interfaces
Int. interface
On-chip modules
SpaceWire I/F
MIL-Bus I/F
UART I/F
GNSS Signal I/F

But today …

e.g. FPGA:

Selected from device libraries

Designer rolls own device

-> Multiple devices on SoC, some unique

-> Maybe very heterogeneous

All internal to 1 chip!

And also …

Often not COTS, may be expensive (> Euro 10K) , limited supply

Chip may be a prototype (FPGA), limited supply

Chip may need to be actually embedded

perhaps in hazardous area

perhaps in area not suitable for debugging

May need flexible access to a scarce resource

And tomorrow …

…. will be worse:

Greater hardware complexity
Greater hardware heterogeneity

Number of device registers -> infinity

Increasingly complex tasks

Even greater software complexity

So the challenges:

How to know what's happening down there.

1. States of the different devices
2. Meanings of register bits
3. What about caches, busses, MMU?
4. What is the software doing???
5. What is happening to the data?

-> intuitive user interface - > GUI

Provide flexible/shared access

User extendable for user defined IP

Test scripting

SoC hardware needed

SoC must have built-in hardware debug support

At least the following:

Debug link to outside, bus master with own command language and protocol

Supporting standard PC connection, at least one of UART, USB, Ethernet, JTAG (via e.g. USB converter)

CPUs must have DSU to allow CPU be controlled and registers be examined

Buffers to capture bus transactions

Hardware breakpoints/watchpoints

Workstation software

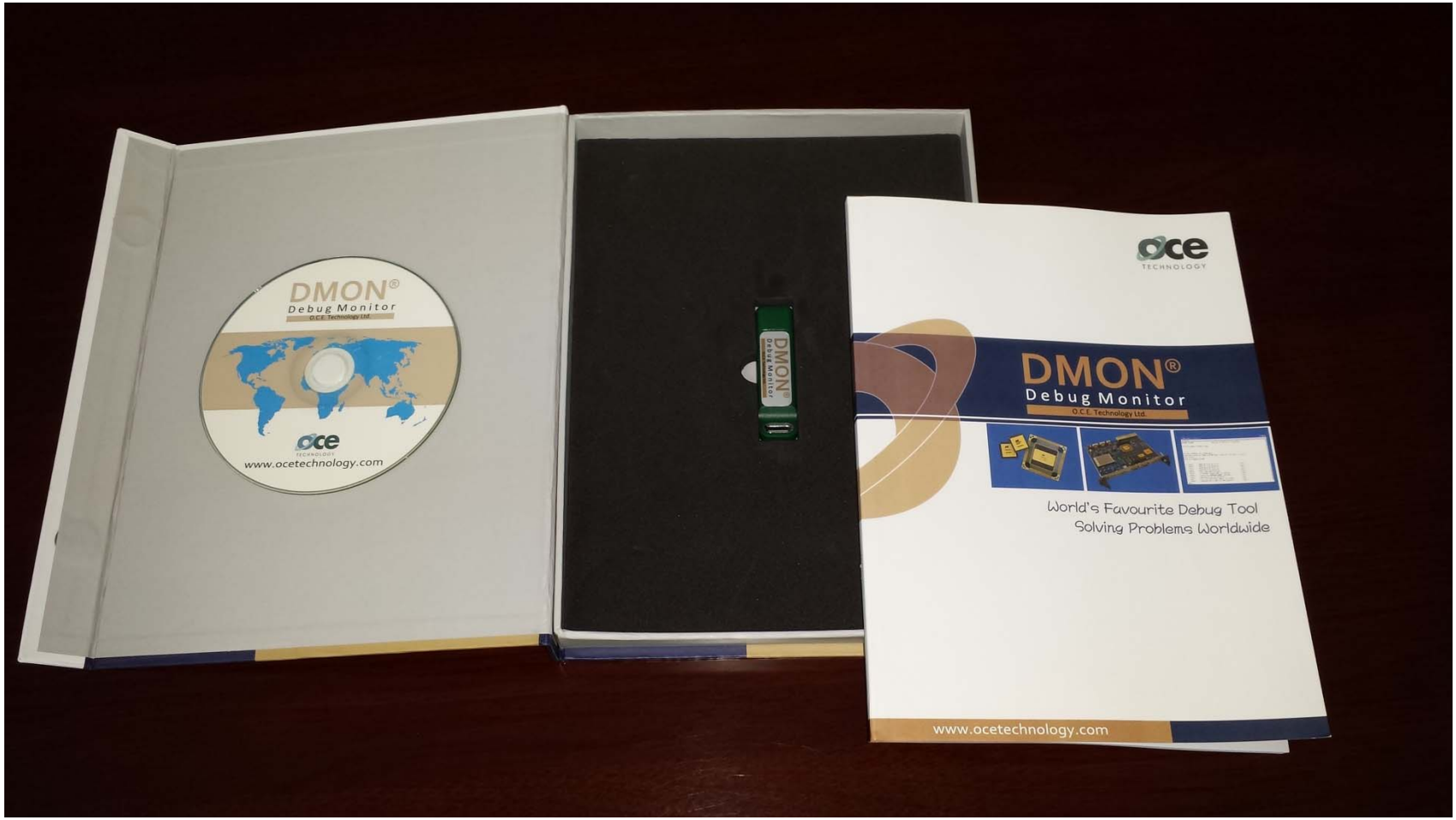
Workstation/PC software able to

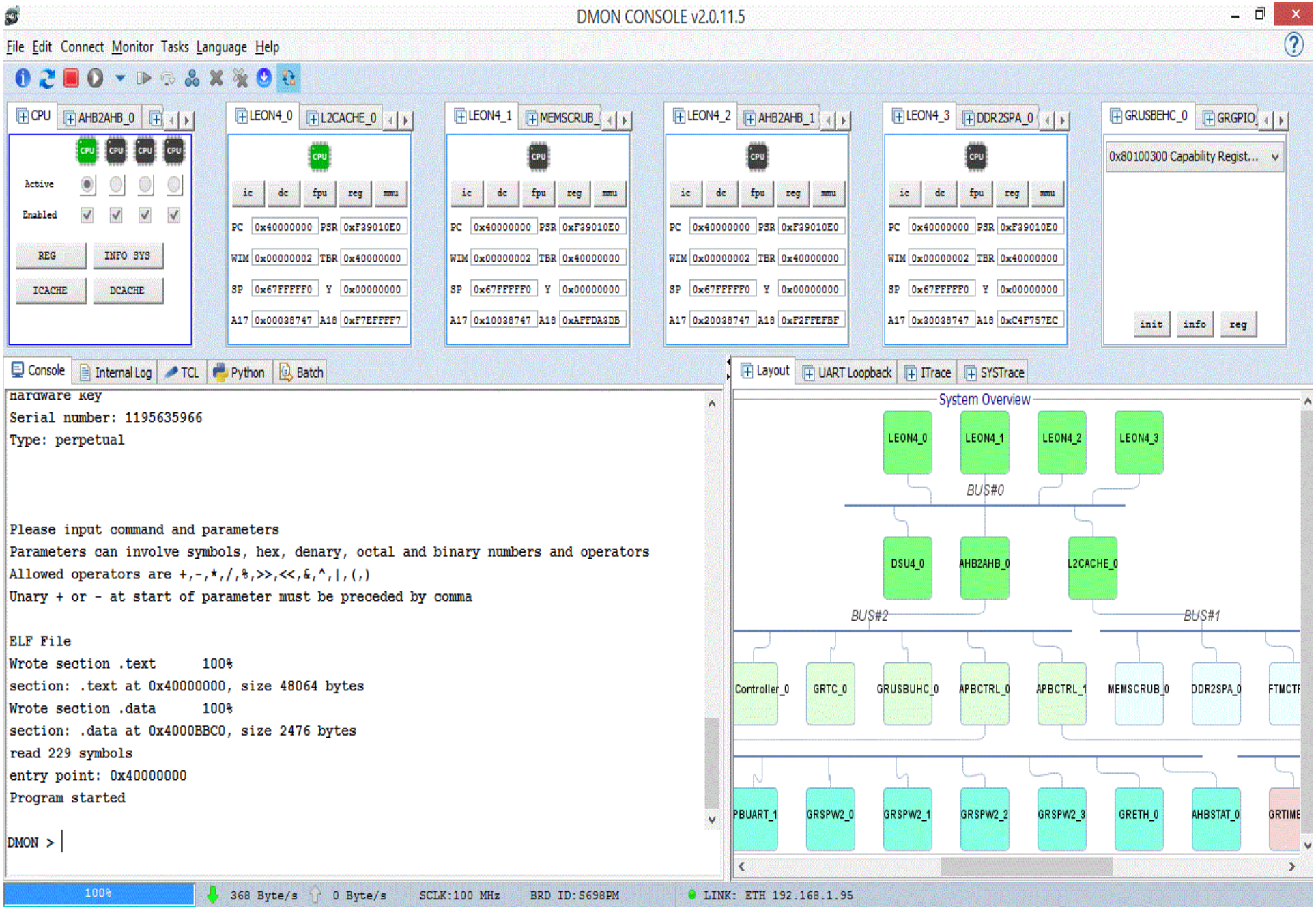
- Use SoC debug link (UART, Ethernet, USB, others via converters)
- Recognise devices present on SoC

- Load executables
- Provide usual software debug, breakpoints etc

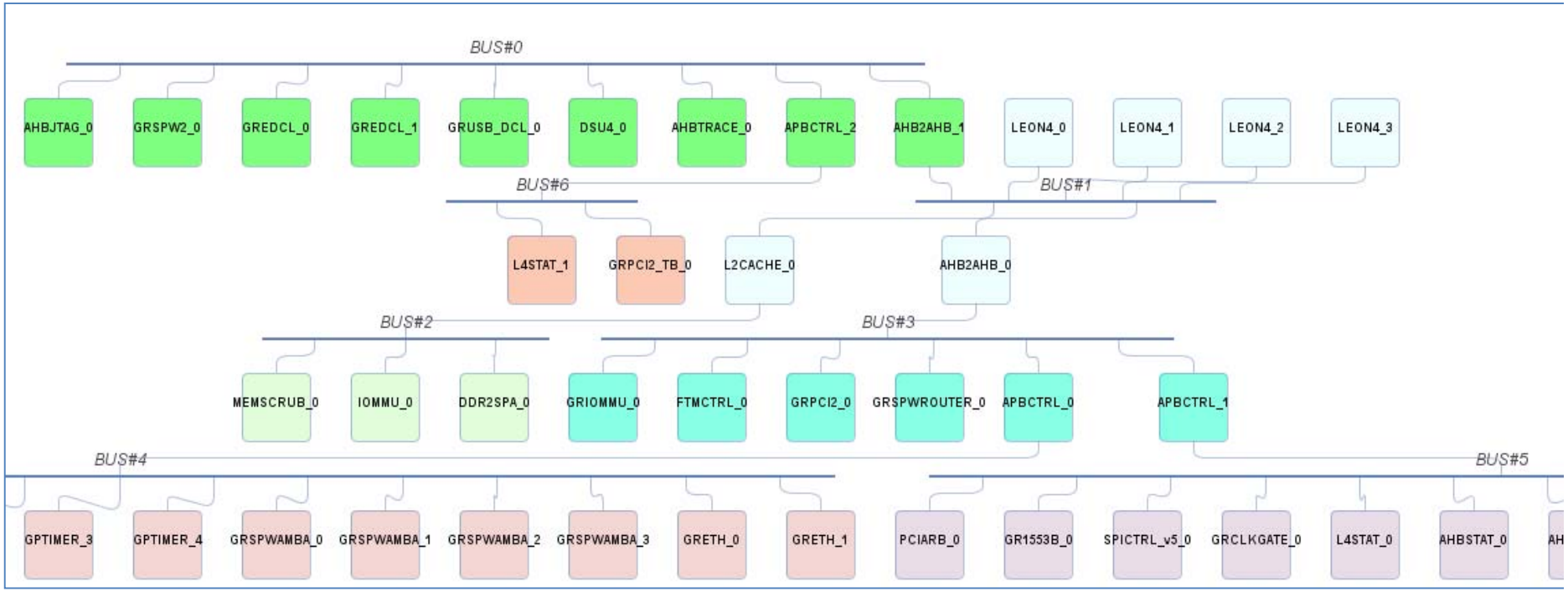
- Provide secure remote access
- Be extended by user
- Monitor data
- Run test scripts (TCL. Python)
- Be intuitive/easy to use

One approach – DMON




DMON CONSOLE v2.0.11.5
File Edit Connect Monitor Tasks Language Help
CPU
AHB2AHB_0
LEON4_0
L2CACHE_0
LEON4_1
MEMSCRUB_
LEON4_2
AHB2AHB_1
LEON4_3
DDR2SPA_0
GRUSBEHC_0
GRGPIO
Active
Enabled
REG
INFO SYS
ICACHE
DCACHE
ic
dc
fpu
reg
mmu
0x80100300 Capability Regist...
init
info
reg
Console
Internal Log
TCL
Python
Batch
Hardware key
Serial number: 1195635966
Type: perpetual
Please input command and parameters
Parameters can involve symbols, hex, denary, octal and binary numbers and operators
Allowed operators are +,-,*,/,%,>>,<<,&,^,|,(,)
Unary + or - at start of parameter must be preceded by comma
ELF File
Wrote section .text 100%
section: .text at 0x40000000, size 48064 bytes
Wrote section .data 100%
section: .data at 0x4000BBC0, size 2476 bytes
read 229 symbols
entry point: 0x40000000
Program started
DMON >
Layout
UART Loopback
ITrace
SYSTrace
System Overview
LEON4_0
LEON4_1
LEON4_2
LEON4_3
BUS#0
DSU4_0
AHB2AHB_0
L2CACHE_0
BUS#2
BUS#1
Controller_0
GRTC_0
GRUSBUHC_0
APBCTRL_0
APBCTRL_1
MEMSCRUB_0
DDR2SPA_0
PBUART_1
GRSPW2_0
GRSPW2_1
GRSPW2_2
GRSPW2_3
GRETH_0
AHBSTAT_0
100%
368 Byte/s
0 Byte/s
SCLK:100 MHz
BRD ID:S698PM
LINK: ETH 192.168.1.95

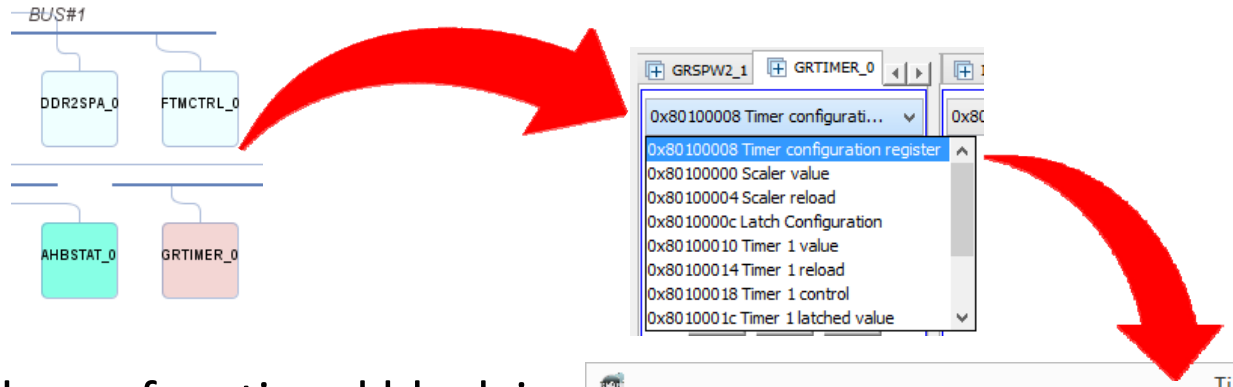
DMON SoC block diagram



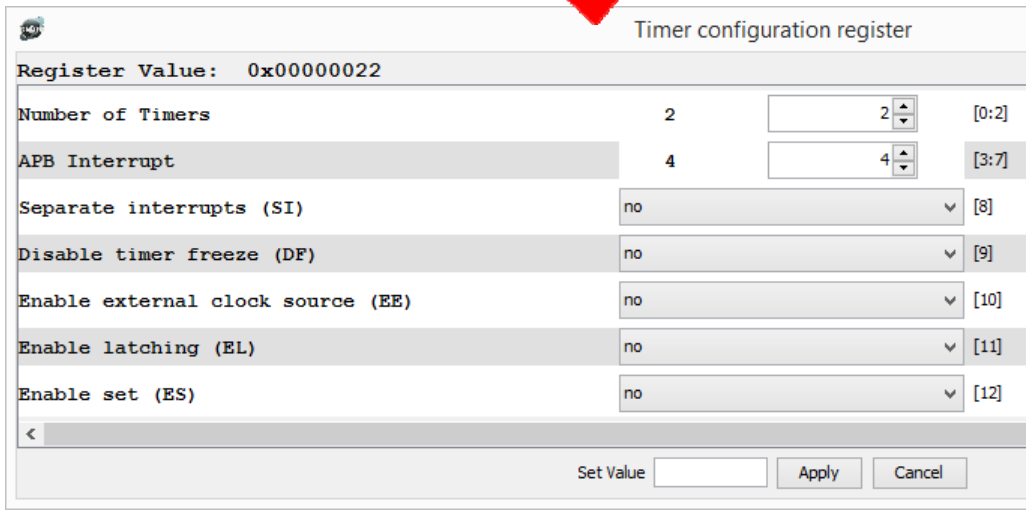
Created from plug-and-play or configuration file

Colour indicates connectivity or state, click on block for details

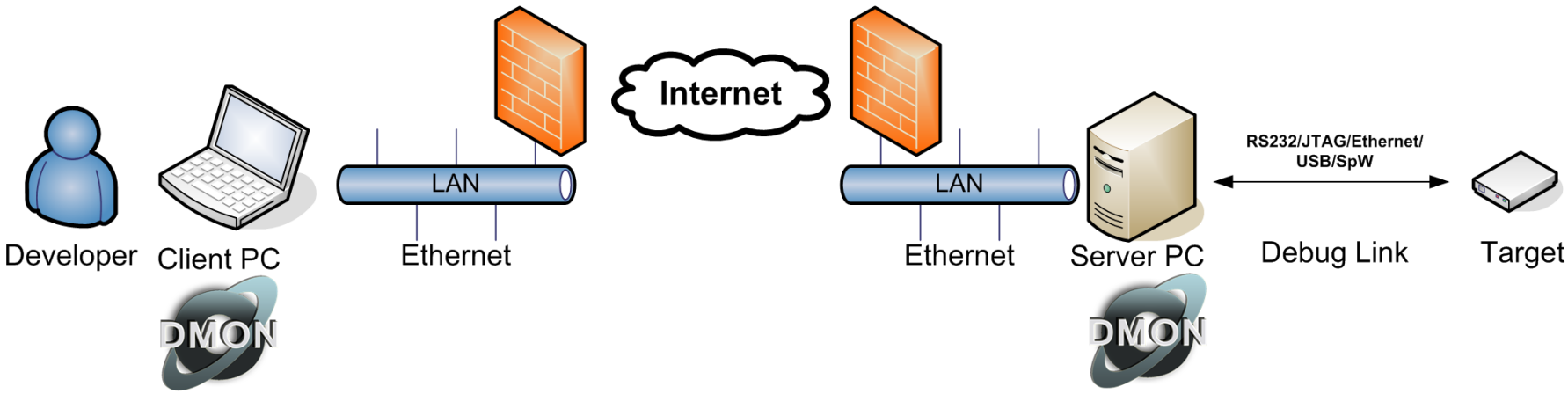
GUI - Drill down to register bits



When a functional block is selected, any of its associated registers can be selected and a new window opened showing the bit definitions of the register.



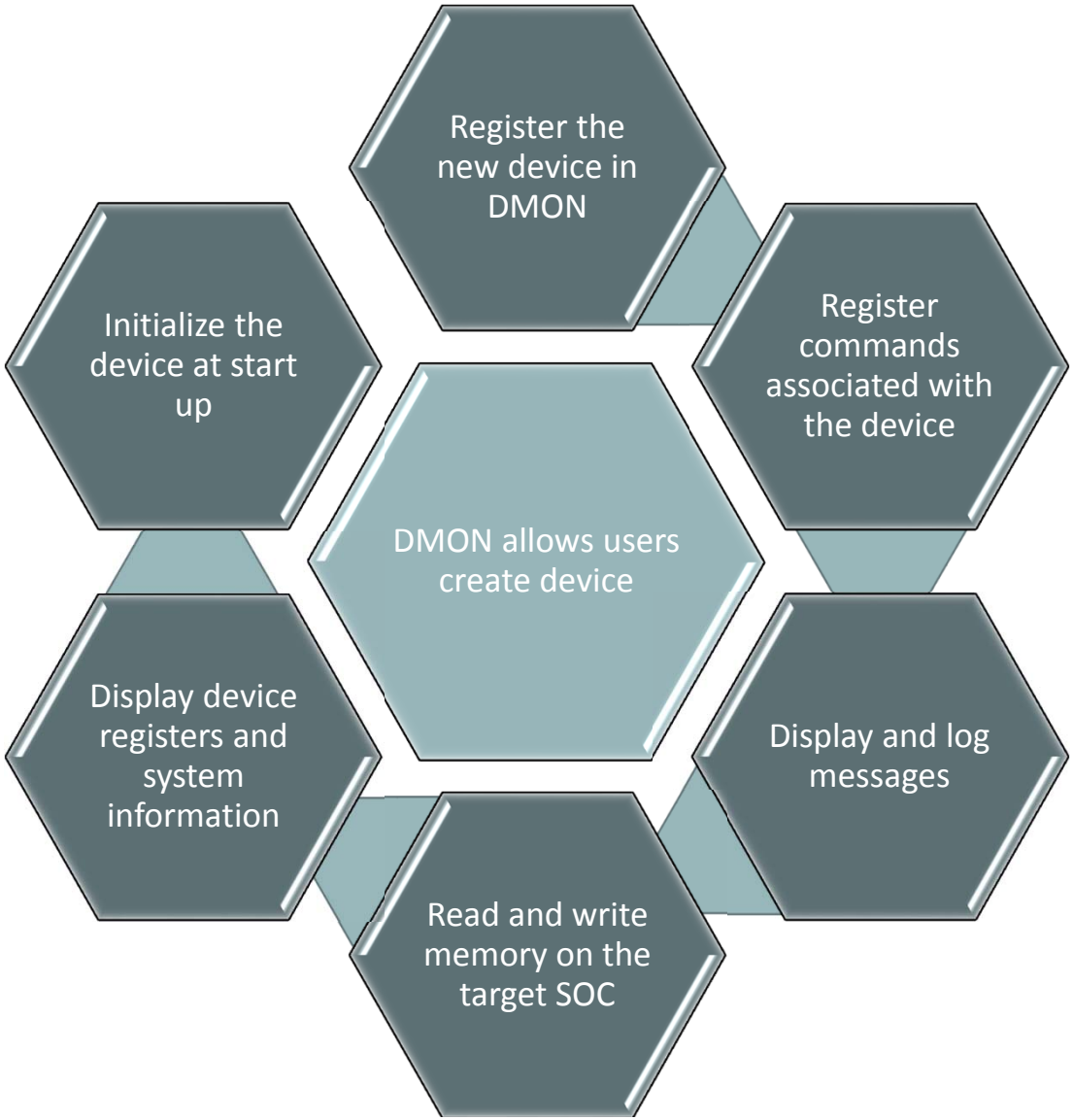
Remote Access (wide-area)



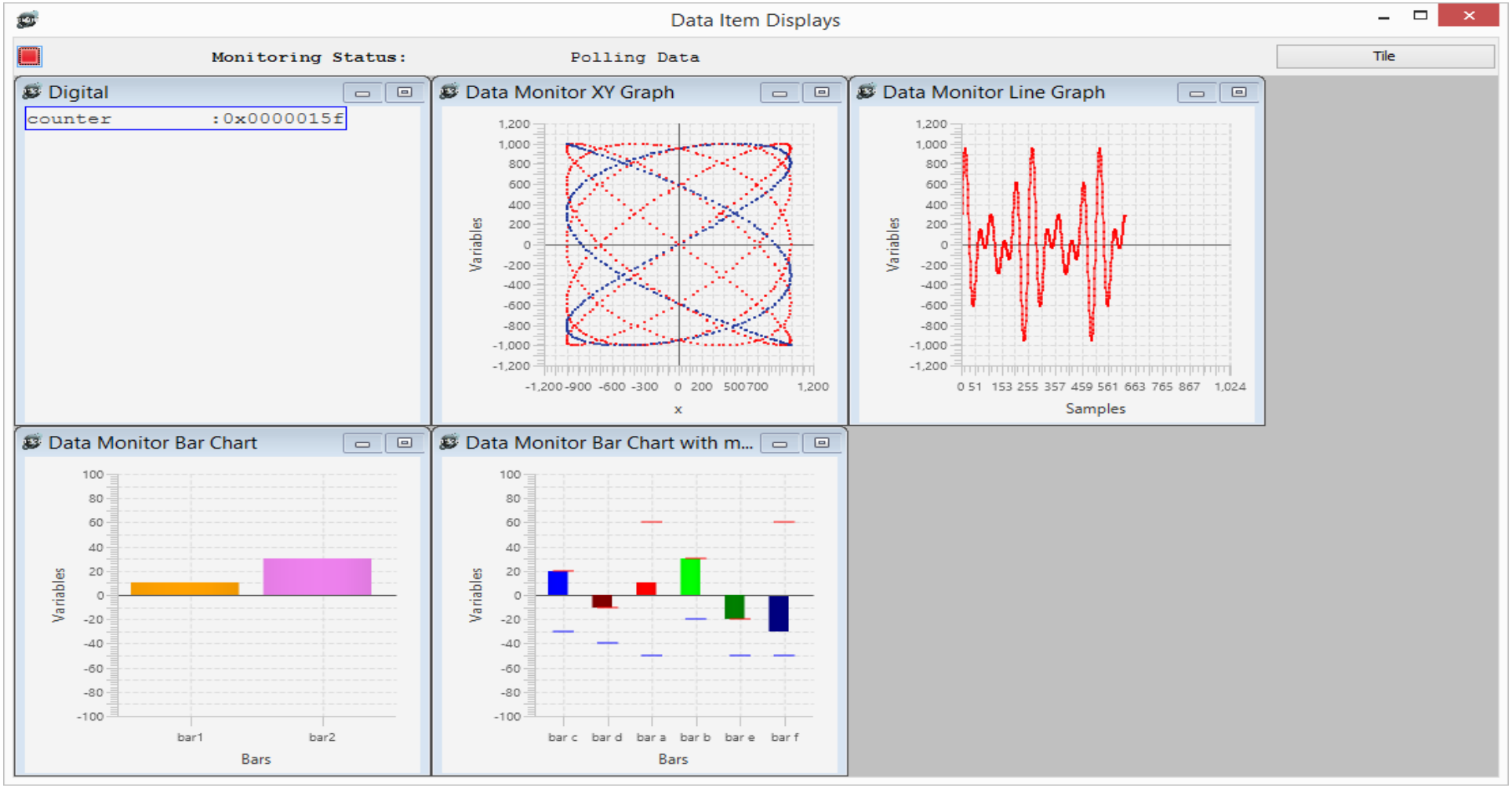
Remote debugging over the Internet secured with SSL

For distributed development or when hardware scarce/inaccessible

DMON support for user-defined IP

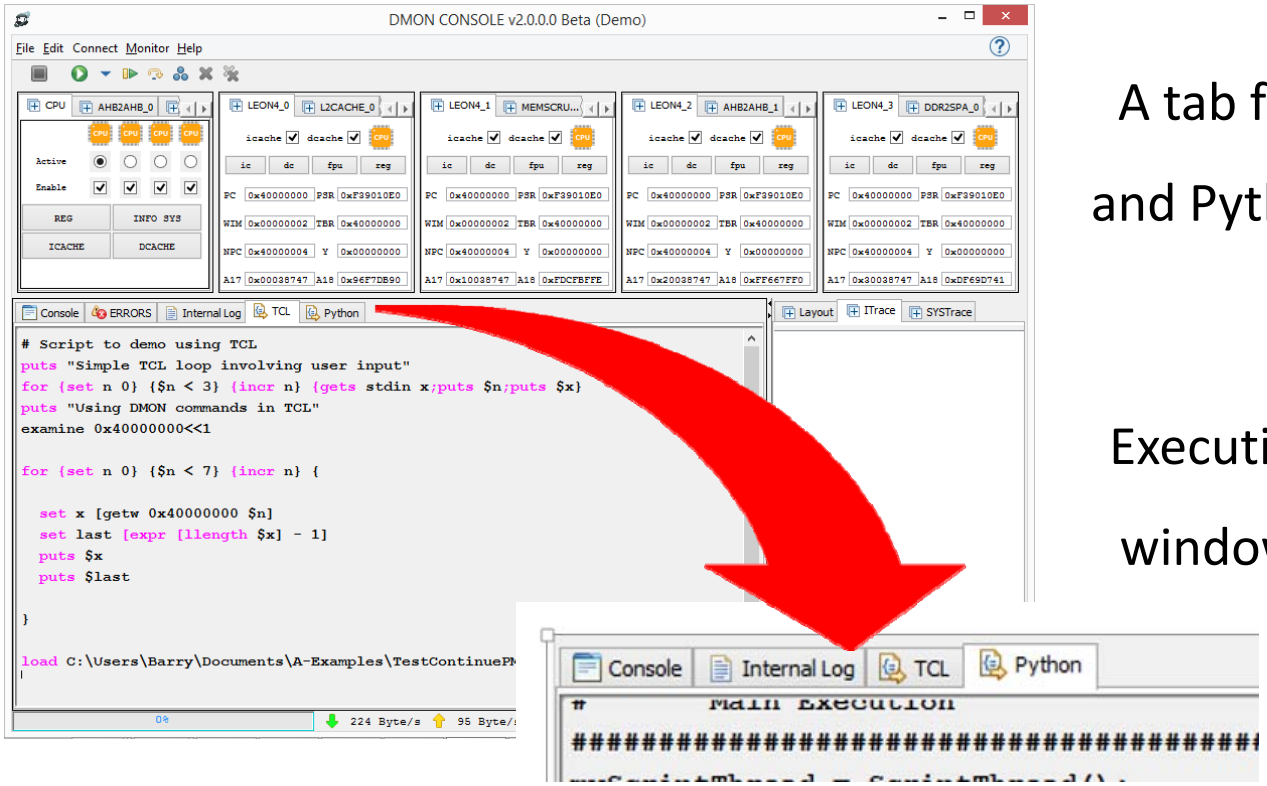


Real-time target data display & store



Data to be shown visually using inbuilt graphs or stored to disk.

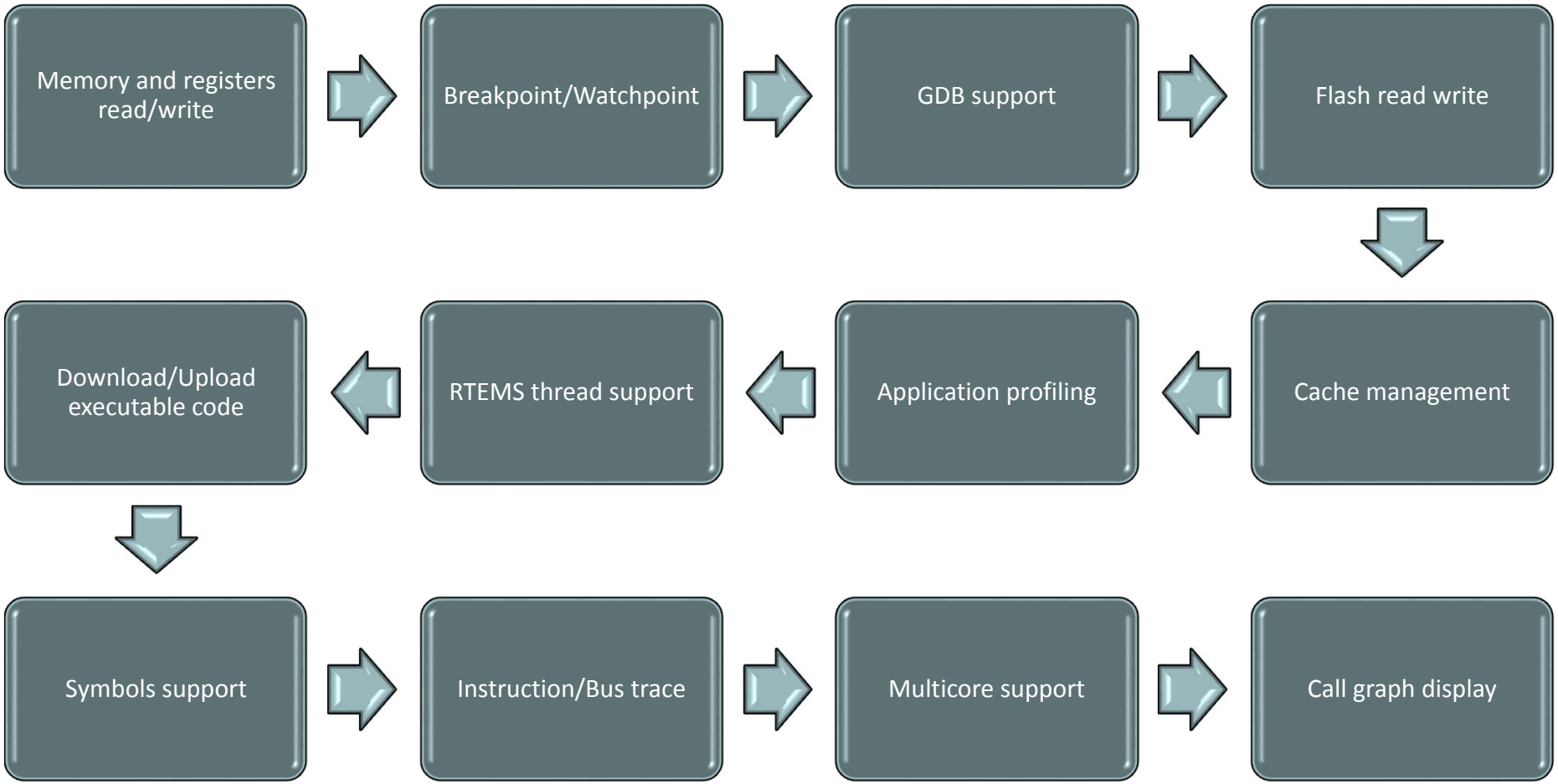
Tcl and Python scripting



A tab for editing and running TCL and Python script files is included.

Executing scripts use the console window for output which can be logged for later analysis.

DMON – Standard Functions



Platforms, architectures & debug interfaces

Client Support

- Windows (XP, 7, 8, 10)
- Linux
- Cygwin

Processor Architectures

- SPARC V8 (multicore)
- ARM

Debug interfaces

- RS232 serial
- Ethernet
- USB
- JTAG (Digilent, FTDI, Xilinx)
- SpaceWire

AGGA-4

Leon2FT CPU
Various interfaces
FFT unit
CRC unit

GNSS Signal processor
36 Channels, settings, correlators etc.

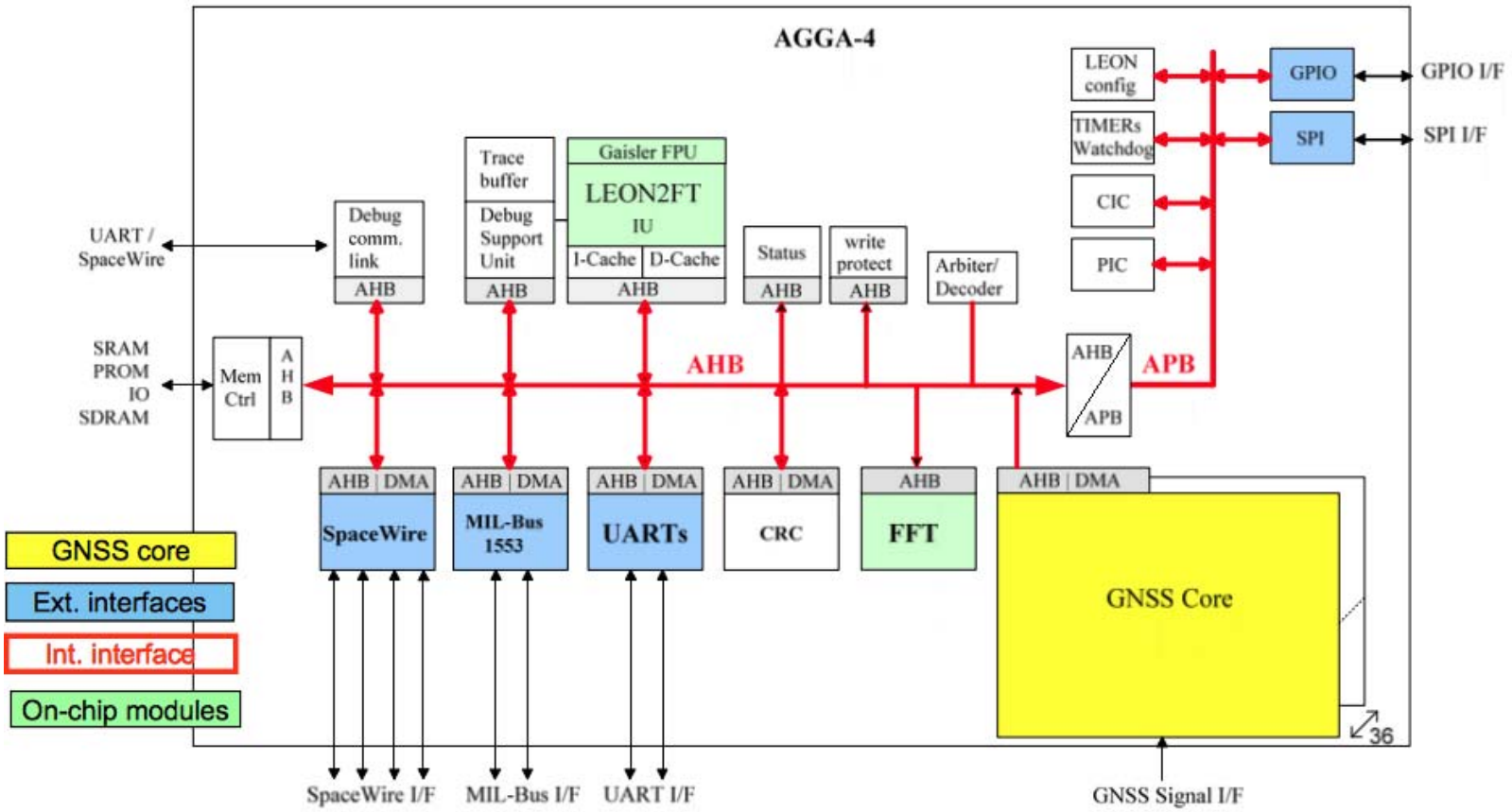
Datasheet(!) 269 pages

Heterogeneous (to say the least)

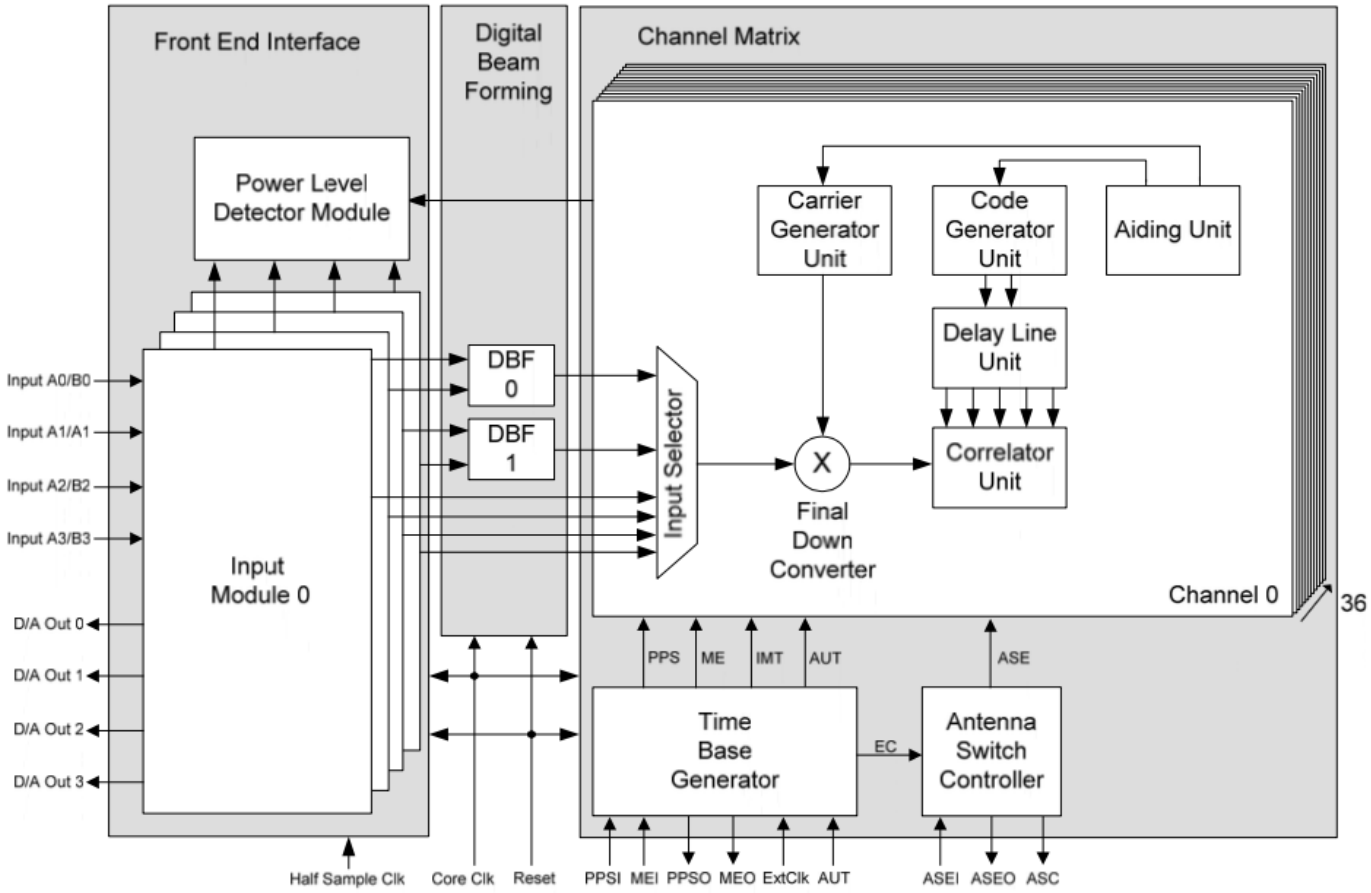
AGGA-4 GNSS

- Can handle different standards (GPS, Galileo, GLONASS,..)
  - Interfaces with up to 4 RF Front-ends
  - Digital Beam-forming unit
  - 36 single frequency double code GNSS channels with aiding functionality (code and carrier aiding for autonomous NCO update)
  - Flexible primary code generators (LFSR and memory based)
  - Support of Binary Offset Carrier and secondary codes
- Used for e.g.
  - Precise orbit determination
  - Meteorology
    - Upper atmosphere, reflections from sea surface

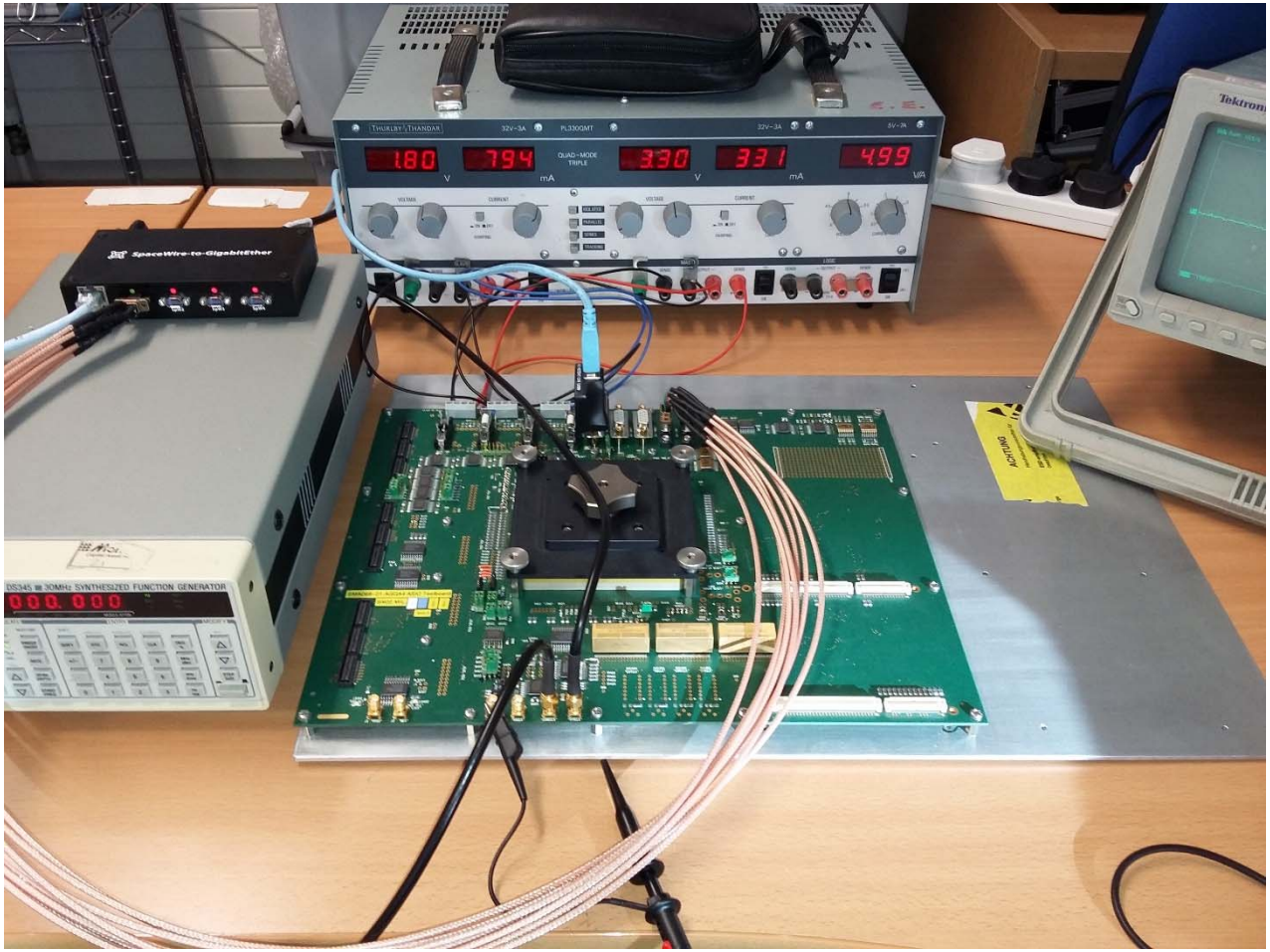
AGGA-4


AGGA-4
LEON config
GPIO
GPIO I/F
TIMERs Watchdog
SPI
SPI I/F
CIC
PIC
Trace buffer
Debug Support Unit
Gaisler FPU
LEON2FT
IU
I-Cache
D-Cache
AHB
Debug comm. link
Status
write protect
Arbiter/ Decoder
UART / SpaceWire
SRAM PROM IO SDRAM
Mem Ctrl
AHB
APB
AHB DMA
SpaceWire
MIL-Bus 1553
UARTs
CRC
FFT
GNSS Core
36
GNSS core
Ext. interfaces
Int. interface
On-chip modules
SpaceWire I/F
MIL-Bus I/F
UART I/F
GNSS Signal I/F

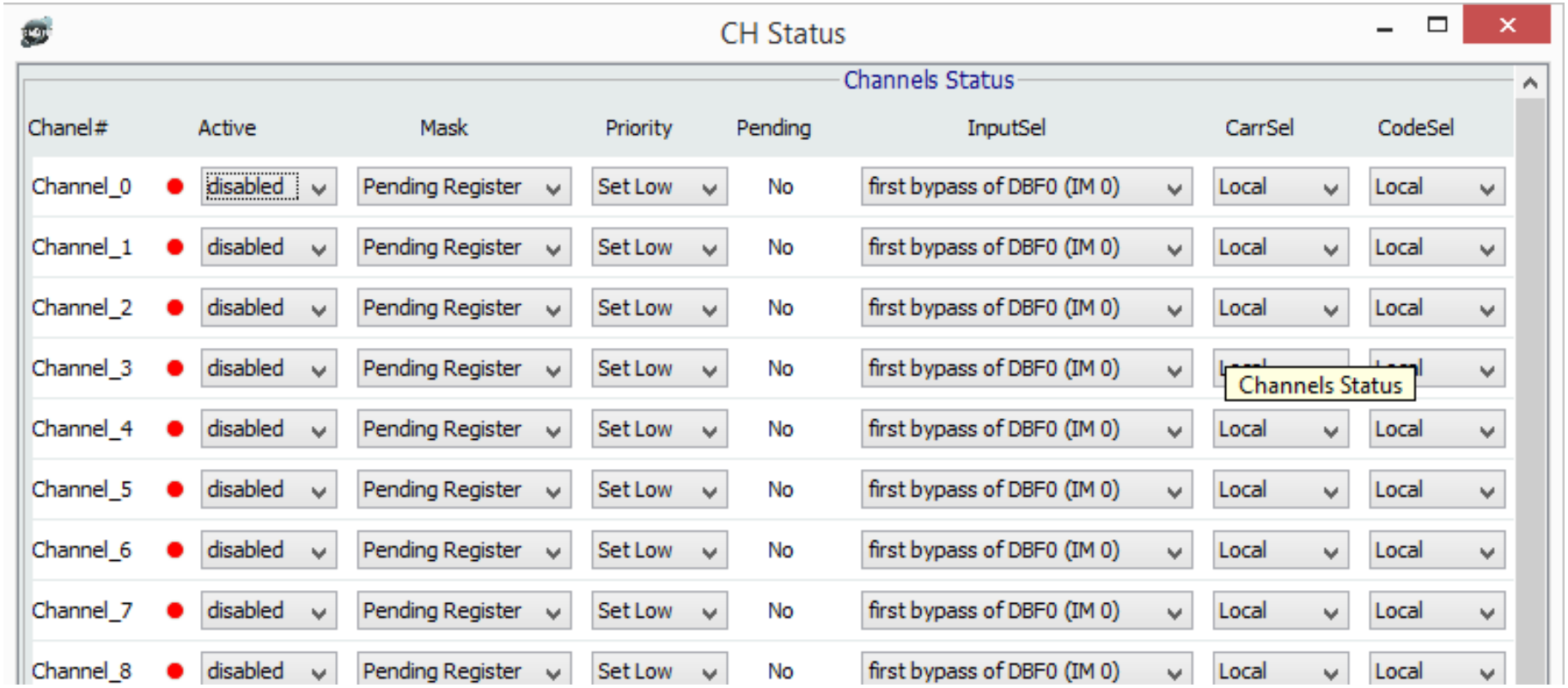
AGGA-4 GNSS


Front End Interface
Digital Beam Forming
Channel Matrix
Power Level Detector Module
Carrier Generator Unit
Code Generator Unit
Aiding Unit
Delay Line Unit
DBF 0
DBF 1
Input Selector
X
Final Down Converter
Correlator Unit
Channel 0
36
Input A0/B0
Input A1/A1
Input A2/B2
Input A3/B3
Input Module 0
D/A Out 0
D/A Out 1
D/A Out 2
D/A Out 3
PPS
ME
IMT
AUT
ASE
Time Base Generator
EC
Antenna Switch Controller
Half Sample Clk
Core Clk
Reset
PPSI
MEI
PPSO
MEO
ExtClk
AUT
ASEI
ASEO
ASC

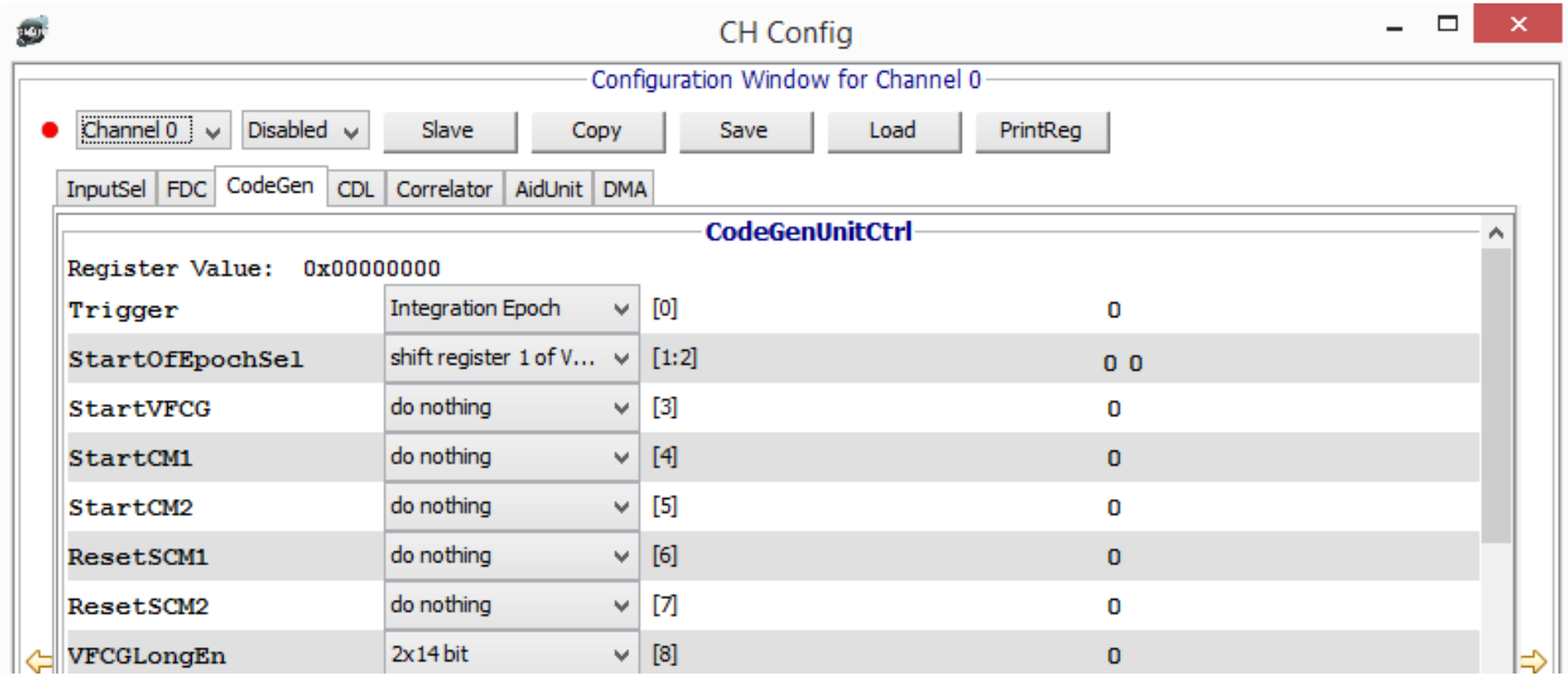
AGGA-4 (Ruag board)



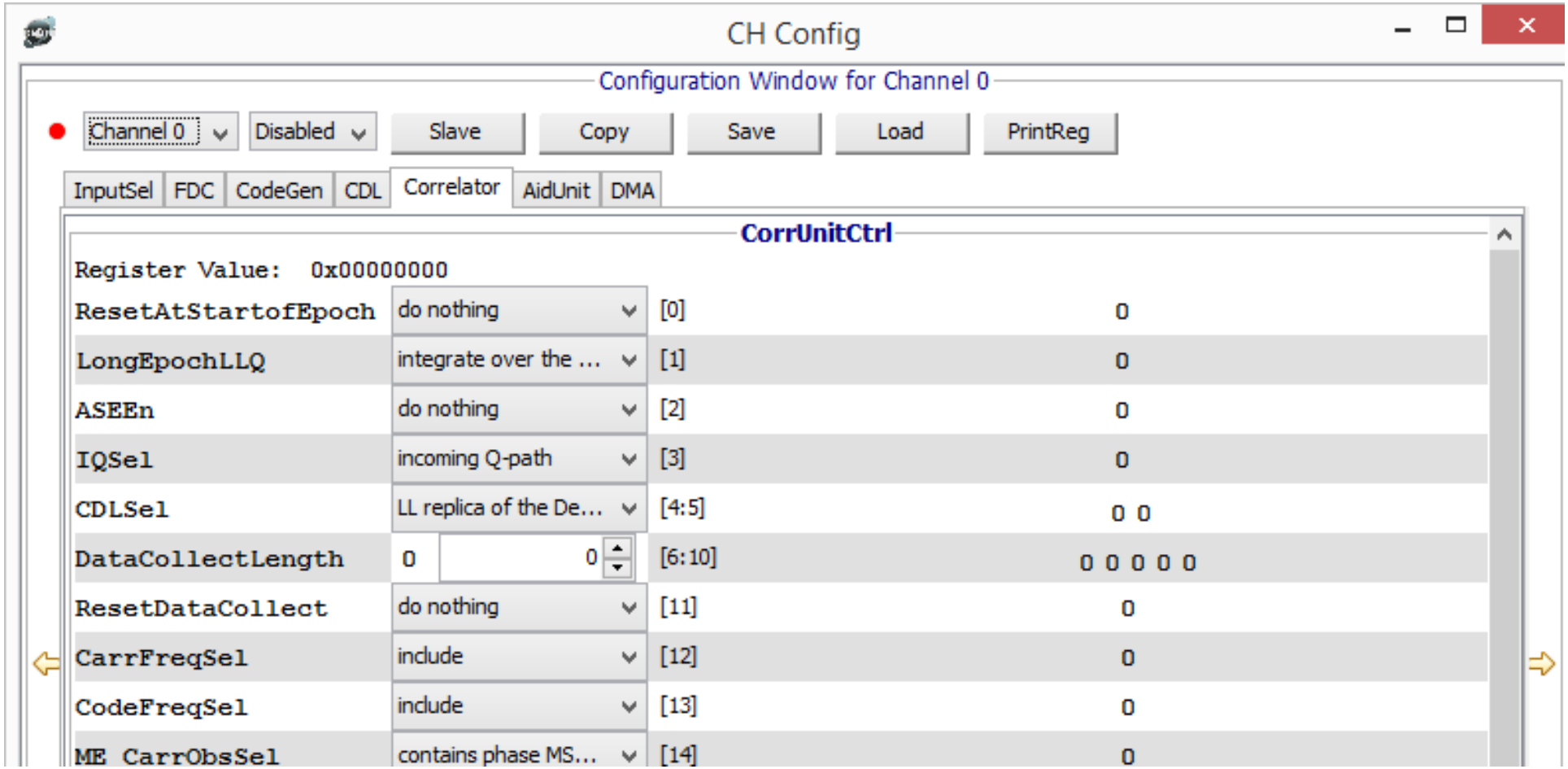
DMON and AGGA-4



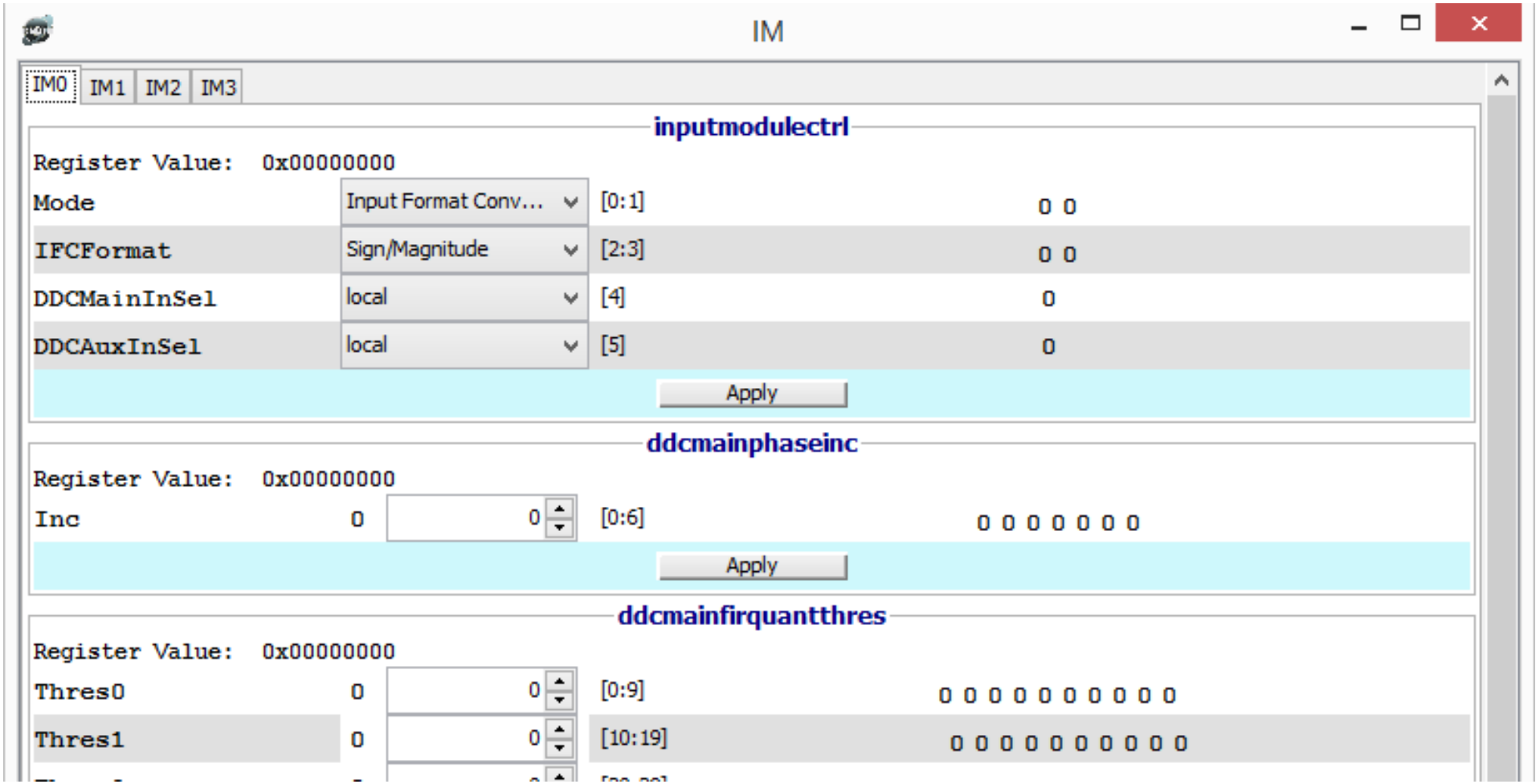
DMON and AGGA-4



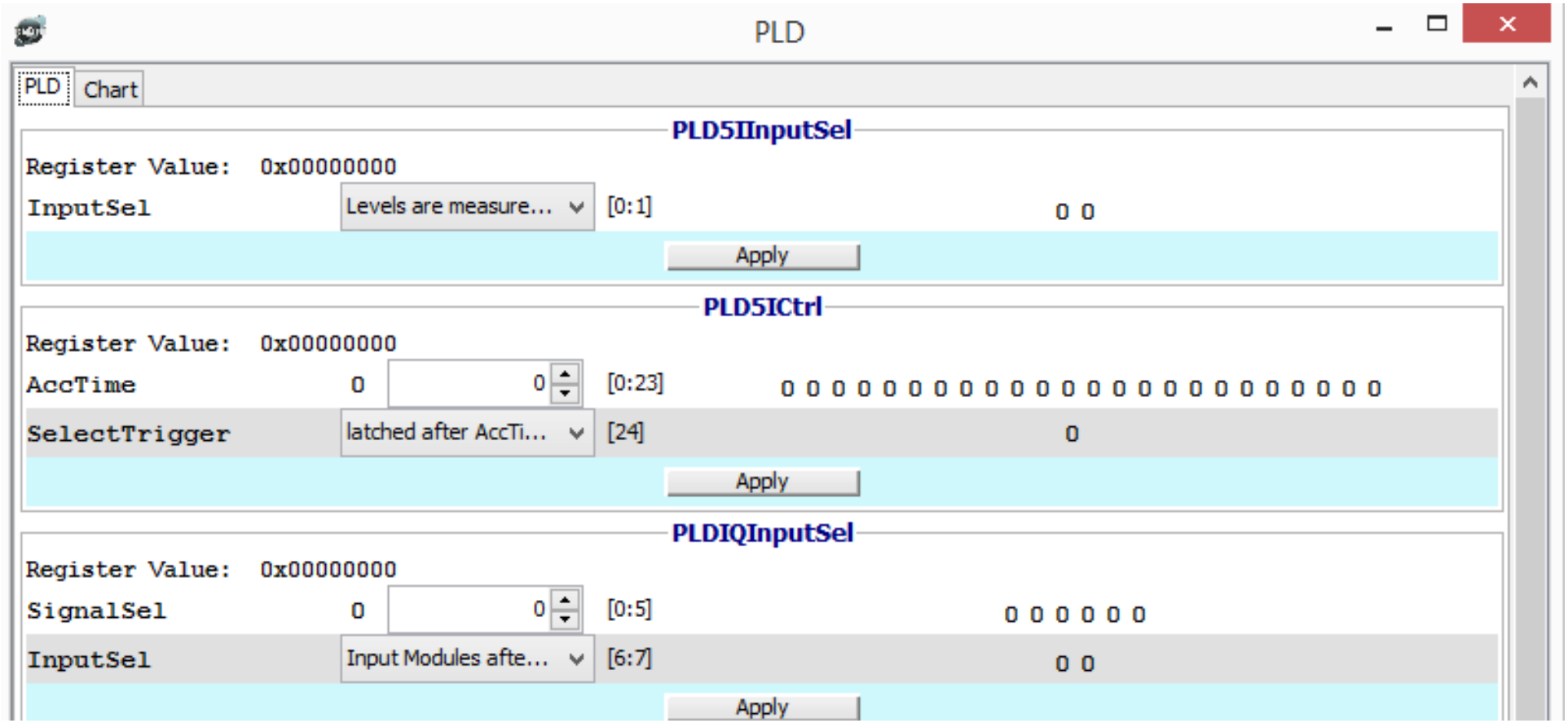
DMON and AGGA-4



DMON and AGGA-4



DMON and AGGA-4



Conclusion

DMON and AGGA-4:

Probably still necessary to read the manual

But a lot easier/more efficient using DMON

And similarly:

Other SoCs much easier/more efficient to work with using DMON

The future:

DMON library continually growing, over 100 devices at present

DMON: The debug System of Choice for Systems on Chip

DMON

Wiki: http://dmon.ocetechnology.com

Evaluation versions available : www.ocetechnology.com

Thanks for support to :

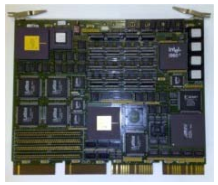
European Space Agency

(Contract 4000112849/14/NL/CBi/fk)

Irish Government (Enterprise Ireland)

OCE Technology

- **Launched 2013, Space & Aerospace Market focusing on Software Tools, SoCs, SIPs, Satellite Subsystems**
- **Distributors: Orbita (China), Dimac Red (Europe), Syraton (India)**
- **Experienced Team: Have delivered products to International OEMs (Siemens, Reuters, Vtel, NEC, Intel, Honeywell) (General purpose, Crypto & XML processors)**



- **Funding & Shareholders: Private Irish Company. Shareholders are private investors and Enterprise Ireland.**

michael.ryan@ocetechnology.com

www.ocetechnology.com

http://dmon2.shoutwiki.com