

<u>Björn Spruck</u> for Mainz Group Worldwide 17.9.2015

B. Spruck, Uni Mainz, 17.9.15, p. 1

- IPMC hardware for carrier is done
 - Firmware for carrier only operation is working
 - incl FRU states (power up/down, hot swap)
 - power negotiation
 - all sensors
 - temperature alarms (which will increase the FAN speed of the shelf or shut down the board)
- IPMI \rightarrow EPICS \rightarrow CSS working
- Final PSU board where the IPMC is plugged on will be ready end of october (JZ, IHEP), until then we have to live with quirks
- MMC layout done, PCB in production, prototype expected end of september.
- MMC firmware is prepared and mostly done
- MMC ↔ IPMC ↔ ShMM gives me an headache still

- monitoring of sensors
- ullet reading limits, min/max and determine error state ullet
- sensors which are not available (yet) ×
 - \rightarrow IOC has to be started after ALL board are power-up
 - Handling of disconnects? (sensor went to major alarm and stayed there)
 - Handling of dynamic sensor changes? (AMC board unplug)
 - Change of limits?
- Anyway, only testable when we have MMC available.
- CSS \rightarrow EPICS \rightarrow IPMC has to be tested (power cycle etc)

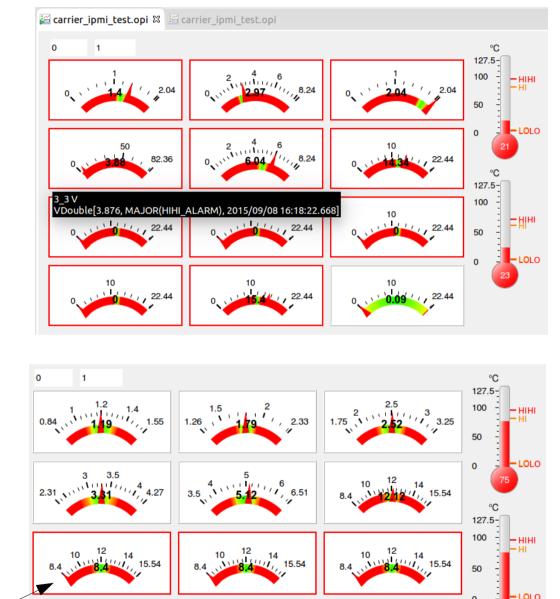
Wishlist for now:

- Startup with unavailable sensors
- Correct handling of de- and reconnect



CSS "issues"

- Scales look strange with limits
- small green band is completely hidden by arrow
- blow up relevant range
- (we anyway have to map 10 bit adc value to 8 bit IPMI value)
- tested with +-30%
- better, but still not too nice.
- (actually, I dislike the GUI items, esp the one for temperature.)



10 12 14 8.4 **8.4 8.4 15.54**

This value is 0
but shown as lowest possible value

B. Spruck, Uni Mainz, 17.9.15, p. 4

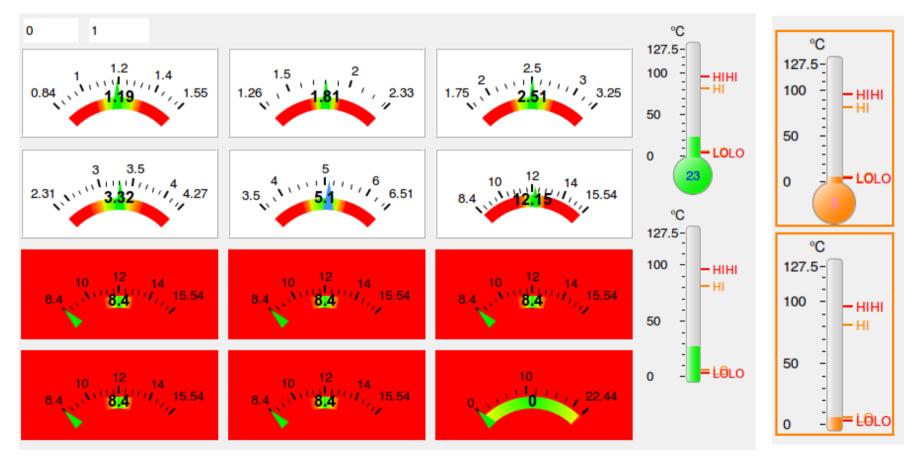
10 12 14 8.4 **8.4 8.4 15.54**





Change colors and behaviour to make it easier to detect abnormal values.

- Background or Fill Color \rightarrow alarm sensitive
- Pulsating alarm does not yield satisfactory results (IMO)
- How are you doing it?



B. Spruck, Uni Mainz, 17.9.15, p. 5



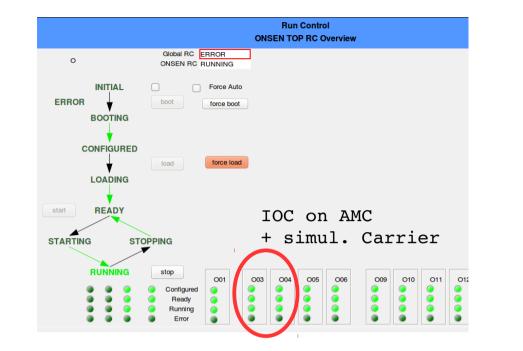
Test with a "complex" system

we know it works with single board and in software.

- Test RunControl with "full" shelf, 8 (AMC) boards and 34 simulated boards (25 AMCs, 9 Carrier).
 - Maximum I could aquire for a test
- Using bitstream and EPICS flash content from 2014 (KEK test)
 - Setup time consuming, as all boards need different MAC, IP and ID.
 - Changed a lot in the EPICS flash content
- Works! Boot, Load, Start, Stop \rightarrow cycle
- Controlled from CSS
- Real IOC runs some "init" script on AMC
 - load, start, stop possible too

Meaningless Screenshots





ONSEN Overview ATCA Shelf Overview											
O01M DI Flow H O01MC RC		03S DI low H O03SC RC	O04S DI: Flow A O04SC RC	O05S DI Flow H O05SC RC	O06S DIS Flow H O06SC RC		O09S DIE Flow H O09SC RC	O10S DI Flow H O10SC RC	O11S DI Flow H O11SC RC	O12S DIS Flow H O12SC RC	
DIG DI CO1M1 RC H	/	DI DI 003S1 RC H									
			D04S2 RC H						011S2 RC H DC DI 011S3		
	l		RC H	RC H	RC H		RC H	RC H	RC H	RC H	
		RC H	RC H		<u>RC</u> H		<u>RC</u> H	RC H	RC H	RC H	
				1							





The output of the 8 IOCs on the AMC boards (by ssh)

15025_000_0015025_000_0021.0	01.94		사망 : 2019년 2019 1919년 2019년 2019
/bin/bash 🛞	/bin/bash 😣	/bin/bash 🛛 🛞	/bin/bash 🛛 😵
₩ /bin/bash 80x24	/bin/bash 80x24	/bin/bash 80x24	
Memory 0x20000000 mapped at address 0x20000000.	Memory 0x20000000 mapped at address 0x20000000.	Memory 0x20000000 mapped at address 0x20000000.	æ /bin/bash 79x23
Transp pointers Using memory \$40000000 bytes from \$20000000 with size \$400 and \$100000 buffers	Remap pointers	Remap pointers	Remap pointers
		Using memory \$40000000 bytes from \$20000000 with size \$400 and \$100000 buffers	Using memory \$40000000 bytes from \$20000000 with size \$400 and \$100000 buffers Reset system
Reset system	Reset system	Reset system	RESET
RESET	RESET	RESET	PROVIDER 1
PROVIDER 1 >SFFEIC00-PROVIDER 1	RESET PROVIDE1 >5FFELC00 <provide1 >00000000c</provide1 	PROVIDER 1 >SFFEICOG <provider 1<br="">>DODDROG</provider>	PROVIDER 1 >SFFEIC80 <provider -00000000<="" 1="" td=""></provider>
>00000000<	>00000000<	>00000000<	
INIT	INIT	INIT	INIT
INIT SITCP	INIT SITCP	INIT SITCP	INIT SITCP ID 1 IP C0.A8.64.12
ID 1 IP C0.A8.64.12	ID 1 IP C0.A8.64.12	ID 1 IP C0.A8.64.12	INIT BLK
INIT BLK	INIT BLK	INIT BLK	INII DER
enable interrupts	enable interrupts	enable interrupts	enable interrupts
npiread2	npiread2	npiread2	npiread2
npiwrite2	npiwrite2	npiwrite2	npiwrite2
aurora	aurora	aurora	aurora
sitcp	sitcp	sitcp	sitcp Interupts enabled!
Interupts enabled!	Interupts enabled!	Interupts enabled!	SysMon: 52'C 980mV 2500mV
SysMon: 51'C 1008mV 2500mV	SysMon: 52'C 994mV 2499mV	SysMon: 51'C 987mV 2499mV	
DONE.	DONE.	DONE.	DONE.
START	START	START	START
Ci /bin/bash 🛛 😒	/bin/bash 🛛	/bin/bash 🛛 🔊	/bin/bash 🛛
₽ /bin/bash 80x24	/bin/bash 80x24	/bin/bash 80x24	₽ /bin/bash 79x23
Memory 0x20000000 mapped at address 0x20000000.	Memory 0x20000000 mapped at address 0x20000000.	Memory 0x20000000 mapped at address 0x20000000.	Remap pointers
Remap pointers	Remap pointers	Remap pointers	Using memory \$40000000 bytes from \$20000000 with size \$400 and \$100000 buffers
Using memory \$40000000 bytes from \$20000000 with size \$400 and \$100000 buffers		Using memory \$40000000 bytes from \$20000000 with size \$400 and \$100000 buffers	Reset system
Reset system	Reset system	Reset system	
PROVIDER 1	PROVIDER 1	PROVIDER 1	PROVIDER 1 >SFFE1C00 <provider 1<="" td=""></provider>
PROVIDER 1 >SFFELC00 <provider 1<br="">>D0000000</provider>	PROVIDER 1 >5FFELC00 <provider 1<br="">>00000006<</provider>	PROVIDER 1 >SFFE1C00 <provider 1<br="">>000000006<</provider>	>5FFE1C00 <provider 1<br="">>00000000<</provider>
>00000000	>0000000<	>0000000<	INIT
INIT	INIT	INIT	INIT SITCP
INIT SITCP ID 1 IP C0.A8.64.12	INIT SITCP ID 1 IP C0.A8.64.12	INIT SITCP ID 1 IP C0.A8.64.12	ID 1 IP CO.A8.64.12 INIT BLK
ID I IP CU.A8.64.12 INIT BLK	ID I IP CO.A8.64.12 INIT BLK	ID I IP CO.A8.64.12 INIT BLK	
			enable interrupts
enable interrupts	enable interrupts	enable interrupts	npiread2
npiread2	npiread2	npiread2	npiwrite2
npiwrite2	npiwrite2	npiwrite2	aurora
aurora	npiwrite2 aurora	aurora	sitcp
aurora sitcp	npiwrite2 aurora sitcp	aurora sitcp	sitcp Interupts enabled!
aurora sitcp Interupts enabled!	npiwrite2 aurora sitcp Interupts enabled!	aurora sitcp Interupts enabled!	sitcp
aurora sitcp	npiwrite2 aurora sitcp	aurora sitcp	sitcp Interupts enabled!
aurora sitep Interupts enabled! SysMon: 53°C 992mV 2500mV DONE.	npiwrite2 aurora sitcp Interupts enabled! SysMon: 53'C 992mV 2499mV DONE.	aurora sitcp Interupts enabled! SysMon: 53'C 989mV 2499mV DONE.	sitcp Interupts enabled! SysMon: 57'C 989mV 2506mV
aurora sitcp Interupts enabled! SysMon: 53°C 992mV 2500mV	npixrite2 aurora sitcp Interupts enabled! SysMon: 53'C 992mV 2499mV	aurora sitcp Interupts enabled! SysMon: 53°C 989mV 2499mV	sitcp Interupts enabled! SysMon: 57'C 989mV 2506mV DONE.



No NSM (yet)

- If one board drops out unexpectedly, the main RC will not notica as the disconnected PV will have no influence on current state
 - When it comes back, should it automatically advance to state of other boards?
 - At least one can by do it hand now w/o/ resetting all boards.
 - Important for tests.