PCIe400: Statut développement software

- Modélisation des composants software
  - Statut
  - Question Mailbox client
- Advanced JTAG Interface (AJI)
  - Addressing an AVMM register
  - Bug encountered

29 Août 2024 –
Julien Langouët

Modélisation des composants software

Modèle → Instances → Quality Assurance (QA): fonctionnel/collect

- Framework complet et testé
- Tests fonctionnel : pass/fail avec boucle au travers de chaque interface disponible (PCIe, AJI, FTDI), génération de rapports au format CSV - Renaud
- Tests collect : mesure de l'ensemble des données pendant une durée et à une fréquence donnée au moyen de l'interface la plus rapide disponible, génération d'un rapport CSV – Renaud

À tester

- p400-qa : instantiation d'un composant via PCIe et AJI

Statut sur les développements

- Composants « Power DC/DC» : LTM4677, LTM4681, LTC2975 Terminés – Renaud, Christophe, Paul
- Objectif : Terminer l'implémentation des modèles de composants externes d'ici à mi-novembre ?
- Cf : component_list_followup.xlsx

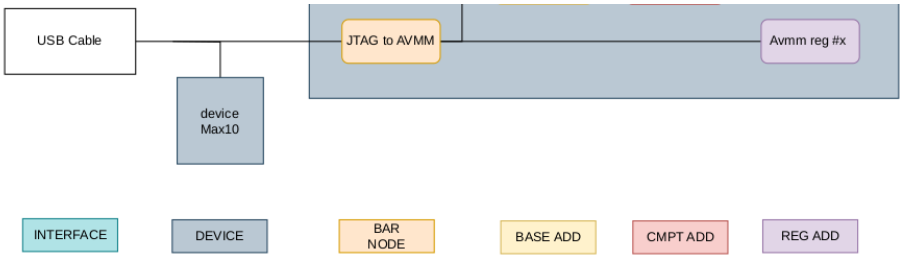
Order	Component	Manufacturer	Description	Protocol	Test setup	Statut	CSV	Model	Test suite functional collect
1	LTM4677	Analog Devices	DCDC	I2C	**Devkit** DC2066A	Ok (configuration, voltage, current, temperature)	X	X	X
2	LTM4681	Analog Devices	DCDC	I2C	**Devkit** DC2524-A-A DC3082A-B	Ok (configuration, voltage, current, temperature)	X	X	X
3	LTC2975	Analog Devices	LDO monitoring	I2C	**Devkit** DC2428A (with LTM4644)	Ok (configuration, voltage, current, temperature) Devkit can only test LTC2975 + LTM4644 case (use LTC2975 + LTM4668A	X	X	X
4	LTC2497	Analog Devices	ADC current and voltage	I2C	DC1012A-B	Model tested (voltage)	X	X	on going
5	Mailbox Client	Altera	FPGA general monitoring	AVMM	**Agilex I/F series** DK-DEV-AGI027R1BES DK-DEV-AGF014E2ES	Model tested (general information about FPGA, voltage, internal temperature)	X	X	on going
6	MAX31730	Analog Devices	Temperature Sensor	I2C	**Agilex SI bit** (from MAX10 with AJI)	Test setup : MAX10 (AJI + JTAG master + I2C opencore)] tested Need to push some small modification to ensure good intantiation of a component through "AJI path"	X	X untested	
7	MAX10 Sequencer	Altera	Sequencer general monitoring Sequencer VMonitor	I2C	**Devkit** FK-10M08E144	Test setup is connected, under debug : 6 rails	X	X untested	
7	MAX10 Sequencer	Altera	Board power control / temperature alert	AVMM	**Devkit** EK-10M08E144	Custom implementation, PIO address mapping to be defined			
8	SI5397	Skyworks	external PLL	I2C	**Devkit** SI5397I-A-EB	Skywork PLL are very similar : SI5397 : 4 independant inputs (crossbar) and 8 outputs	X		
9	SI5391	Skyworks	external PLL	I2C	**Agilex I series** DK-DEV-AGI027R1BES	SI5395 : 4 inputs and 10 outputs SI5394 : 4 inputs and 12 outputs SI5391 : 3 inputs and 11 outputs			
9	SI5394	Skyworks	external PLL	I2C	**Devkit** SI5394I-A-EB				
9	SI5395	Skyworks	external PLL	I2C	None	**They should be implemented with a BaseSi class and inheriting specific class**	X		
10	LMK04821	Texas Instrument	PLL	SPI	Idrogen		X	X	
11	Core White Rabbit	CERN	white rabbit monitoring	UART	Idrogen	Banc de test accessible pour implémenter la suite de test ?	X	X	
12	OBT	Amphenol	Opto-electronic XCVR	I2C	None				
13	QDFP112	TBD	Opto-electronic XCVR	I2C	None				
13	SFP+ 10GPON	TBD	Opto-electronic XCVR	I2C	None				
14	XCVR PHY	Agilex	FPGA XCVR F-tile reconfig, monitoring	AVMM	Agilex I series	Waiting to determine best implementation of PHY IP (bonded, bidir, ...) because it changes the AVMM register mapping			
15	Frequency measure	Agilex	Monitor core clock frequency	AVMM	Agilex I series	Custom implementation exist, need to formalize a general AVMM register map			
16	DDMTD	Agilex	Measure delay between clock	AVMM	Agilex I series	Custom implementation exist, need to formalize a general AVMM register map			

Mailbox client IP conflict with Mailbox AVST Client

Mailbox AVST client has been implemented to retrieve chip_id (FPGA Serial Number) and expose BAR2 @ 0x8000_0010 0x8000_0020

- Timeout from Mailbox Client IP when adding Mailbox Client IP (with AVMM) to peripheral QSYS
- Can we get rid of the conflict ? Or should we retrieve FPGA serial number in another way ?
  - In fact Mailbox Client is accessing a single hardware ressource : SDM (boot microcontroller for the FPGA)

Advanced JTAG Interface (AJI)



Addressing an AVMM register through JTAG

- ***Interface*** : None of the ways to identify the interface are unique → specific to the board you are addressing to
  - **cable id :** start from 1, depend on number of JTAG cable available to the server
  - **cable** persistent ID : (start from 1, when through JTAG server typically > 0x02000000
  - USB **port :** eg 1-2.3
  - **cable name** : Name of the board appearing on `jtagconfig` output (eg : AGI FPGA Development Kit, PCIe400)

    Cable persistent ID should be persistant from power cycle, but also the cas for cable id until now, persistent ID does not always match cable id => should not be used...

- ***Device*** :
  - device_id : index in the JTAG chain (start from 0)
  - device : hashed version of device opcode (exact reference), appears on `jtagconfig` output eg : AGIB027R29A1E2VR3 gives 0x034BB0DD
  - device_name : string appearing on jtagconfig output for eg : AGIB027R29A(.|B|C|R3)/.., partial name is acceptable, error if ambiguous

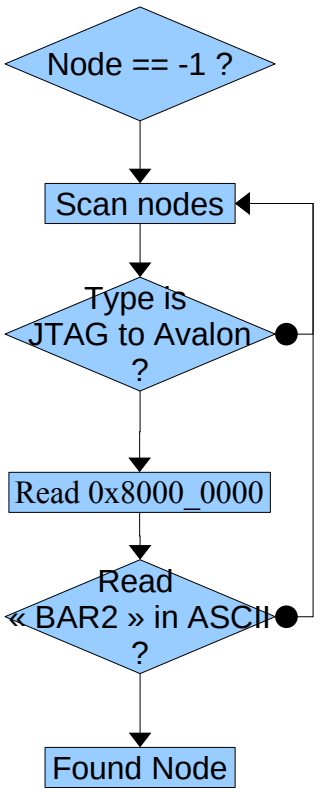
- ***Node*** :
  - Rank of the JTAG virtual node on device, it can change if a signal tap is added,
  - Driver libaji accepts a specific node, if -1 is provided → triggers an auto node identification

- **Register address** : Base address AVMM address found on Qsys + Offset in IP documentation

Advanced JTAG Interface (AJI)

Auto node identification

- You need to read « BAR2 » in ASCII at address 0x8000_0000



Advanced JTAG Interface (AJI)

Bug encountered

- Every time a frame contained byte with value 0x7C, 0x7A or0x7B exit with AJI_FAILURE
  Reason : Bad byte stuffing in AVMM /AVST encapsulation and decoding
  - 0x4D is used to escape IDLE [0x4A] char and 0x7D is used to escape SOP, EOP and CHAN [0x7A, 0x7B, 0x7C] control characters

- Conflict between libaji and system_console
  As soon as system console has claimed path once, libaji fails to read or write an AVMM register.
  - New bug since initial release 1.4.0 of p400-rw-aji ? In fact, previous bug has been fixed using systemconsole to write, libaji to read the same register.
  - Using different release of system_console (from Q23.4 or Q24.2) does not seem to change behavior.
  - Requires some more investigation with regression happened in AJI driver or new bug...