COMPSYS 701

ADD-HSOC - PHASE TWO
GROUP 8

Project Update - Group 8

ReCOP and ASP adjustments since Phase One:

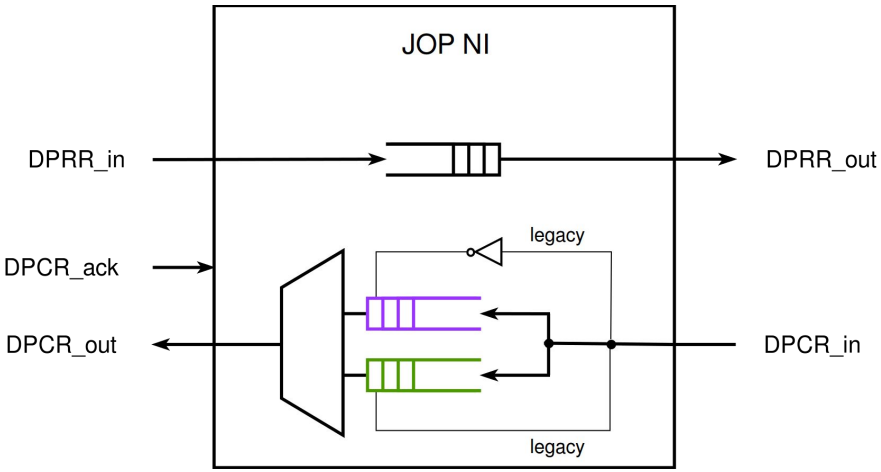
- ReCOP architecture from Von Neumann to Harvard
- ASP reset function

For Phase Two:

1. JOP Network Interface (JNI)
    a. ASP mapping function for the JNI to interface with ASP
    b. New input FIFO (separate FIFOs for ASP and ReCOP)
    c. Multiplexer for DPCR out data for ASP and ReCOP
2. ReCOP to JOP communication
    a. Java and ReCOP assembly program which communicates with each other
        i. DataCalls sent from ReCOP, and JOP responds with result
        ii. Communicates to ASP with corresponding commands
    b. Own ReCOP placed into NoC
3. ASP connection to the NoC
    a. ANI and ASP compiled into a single component
    b. Interfaces with the NoC and consequently, with the JOP
4. Multi JOP processor program
    a. Matrix multiplication program developed
        i. One core JOP constructs matrix while other JOPs supplements product calculations

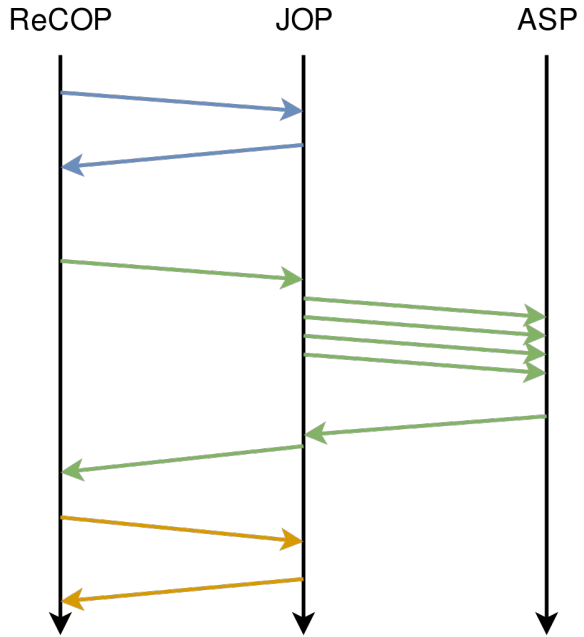
Phase Two: JNI modifications

Input FIFO for packets from ReCOP and ASP
Legacy Bit indicate ASP(1) or ReCOP(0) packet



JOP Network Interface

rd_asp_data <= valid and legacy
rd_recop_data <= valid and not legacy



Packet Communication
e.g. ASP (Store 3)

Refer to `jni_changes` for details

Phase Two: Interconnect Done

ReCOP Behaviour (**Our ReCOP**):

- `DCALLNB` → Await JOP response
- Next `DCALLNB`

JOP Behaviour:

- Receive ReCOP message
- Processes result to sent to ReCOP (Calls upon ASP if required)
  - Set DPRR to ASP if required
  - Result produced
  - Set DPRR to ReCOP
- Next ReCOP message

ASP Behaviour:

- Receive JOP message
- Produce result, send to JOP
- Next JOP message

Future Implementation - Group 8

For Phase Three (Final Phase):

1. SystemJ program
    a. Integration of all components with SystemJ, for full blown ADD-HSoC