Additional Details Regarding Data Calls

1 The DPRR register

Your ReCOP has to handle two types of data calls, blocking and non-blocking. The blocking data call should cause your processor to essentially suspend operation until a result appears. The non-blocking call allows your processor to continue operation as normal.

DPRR should be 32 bits wide. The following fields are used:

Field	Use
DPRR[0] - RES	Result of operation presented by external source (JOP)
DPRR[1] - NRR	New result ready, received from external source (JOP)
DPRR[2..17]	Data Lock Address, the address in ReCOP Data Memory where the new result RES has to be stored when ready (NRR=1)
DPRR[17..31]	Currently unused

2 Behaviour

DCALLBL is a blocking instruction that blocks ReCOP control flow until the result from JOP is returned.

DCALLNB is the non-blocking instruction. This launches the call through DPCR, but does not block ReCOP and continues execution of a program on ReCOP as long as there is anything to do.

As new data call processing cannot be started before the previous call has been processed, all non-blocking data calls are stored in a FIFO that can be read by JOP in order of their arrival.

However, in order to get information on the completion of the non-blocking data call, ReCOP's control unit has to be modified by introducing an additional check for whether a new result is ready (NRR=1) before each new instruction fetch (fetch becomes conditional).

If a new result is received, both least significant bits of DPRR are automatically stored in the data lock address specified by DPRR content (effectively, DM[DPRR[18..2]] ←DPRR[1..0]) and you should pad this result with zeroes. After this, normal instruction cycles on ReCOP can resume.