Tile Link Protocol :

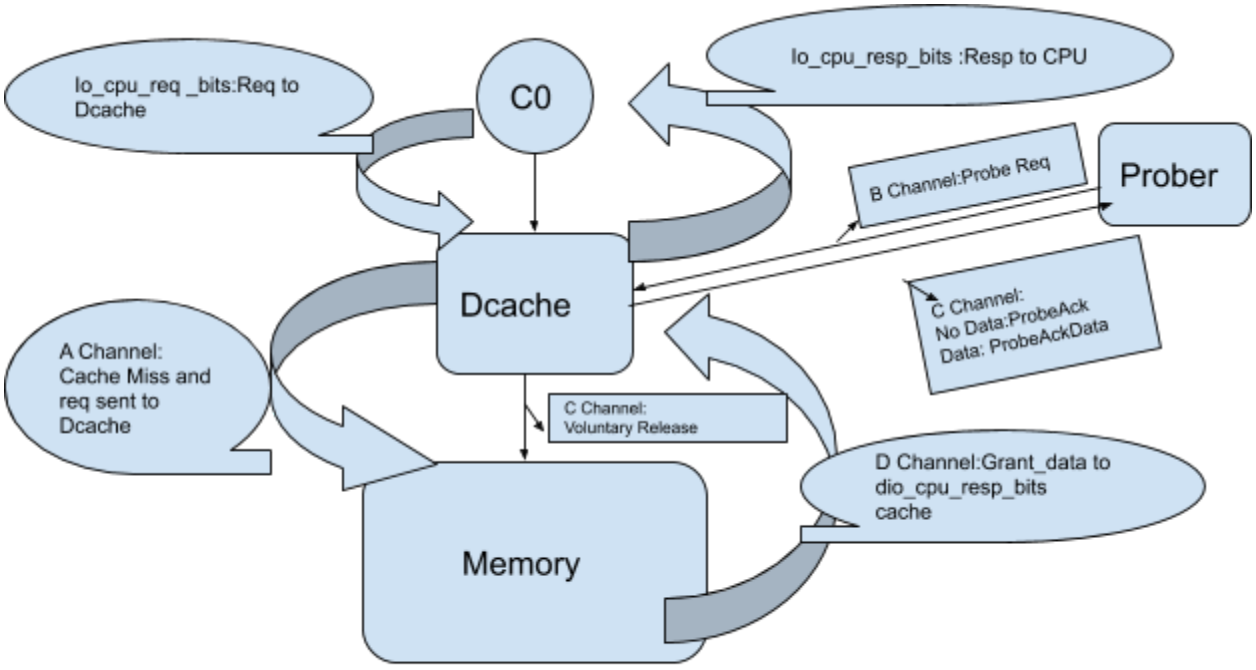
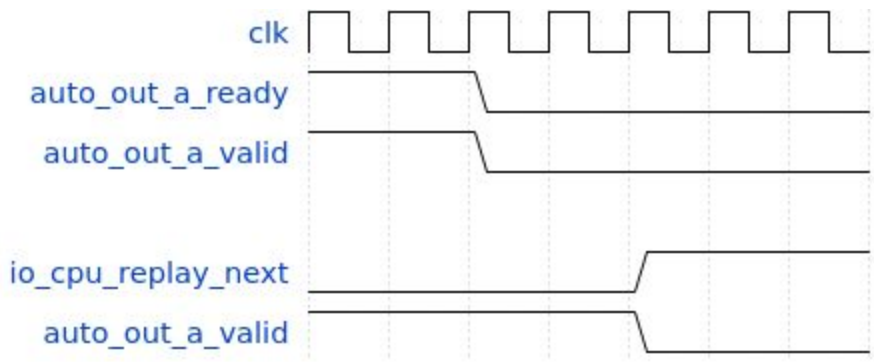


Figure 1: Tile Link Protocol

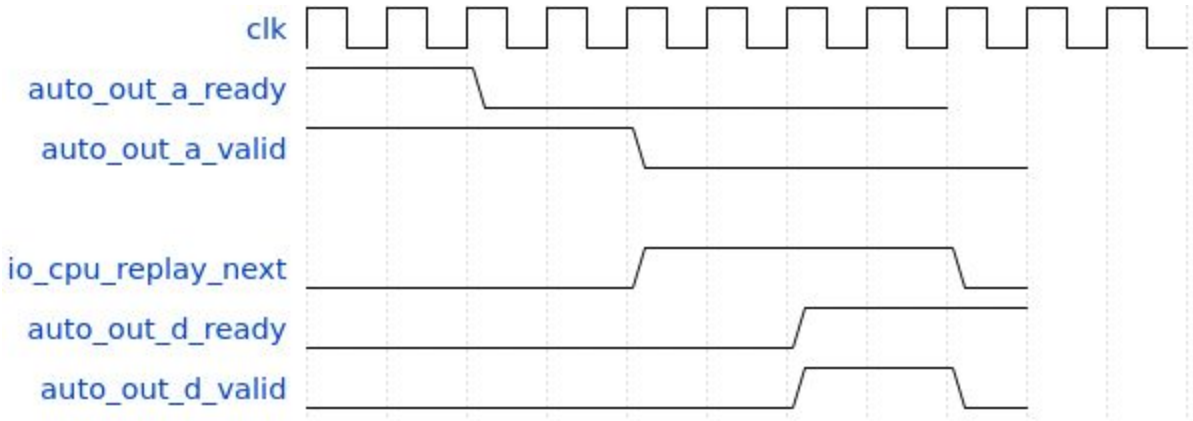
This Document specifies the values of the output signals during the DCache access. The signals specified below is the output signals of the NonBlockingDcache.v module. The values are extracted from the timing diagram of the dhrystone benchmark(dhrystone.riscv.vcd) using the rocketchip emulator.

Signals in NonBlockingDcache.v:

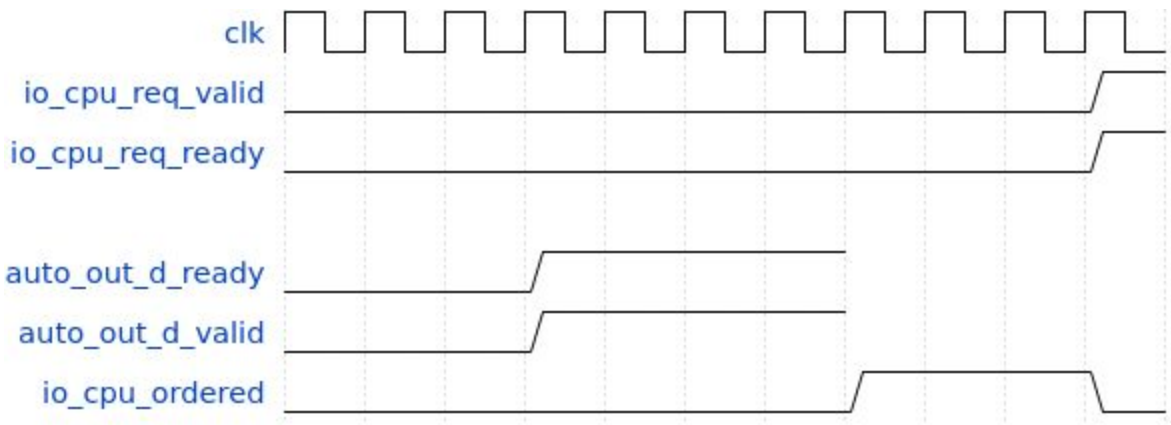
1. Value of output signal: io_cpu_replay_next
   when (auto_out_a_ready==1 && auto_out_a_valid==1) +2 ns
   then io_cpu_replay_next=1;
   auto_out_a_valid==0;



when (auto_out_d_ready==1 && auto_out_d_valid==1) (still io_cpu_replay_next=1) after 2 cc
io_cpu_replay_next=0;
auto_out_d_valid=0;



2. output signal: io_cpu_ordered
when (auto_out_d_ready==1 && auto_out_d_valid==1)
after 2 ns+2 ns:
io_cpu_ordered=1 until (io_cpu_req_valid==1 && io_cpu_req_ready==1)
after 2cc:
io_cpu_ordered=0;



3. output signal: io_cpu_req_ready:

If READ req:

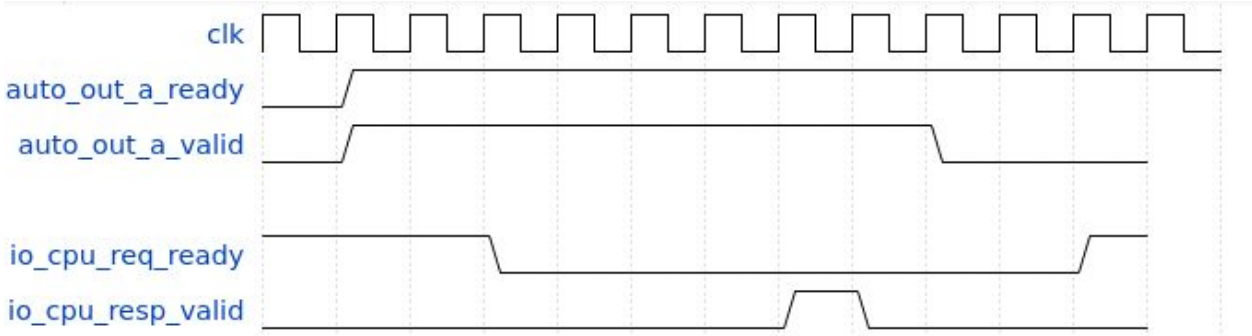
when (auto_out_a_ready==1 && auto_out_a_valid==1)+2 ns

then

io_cpu_req_ready==0;

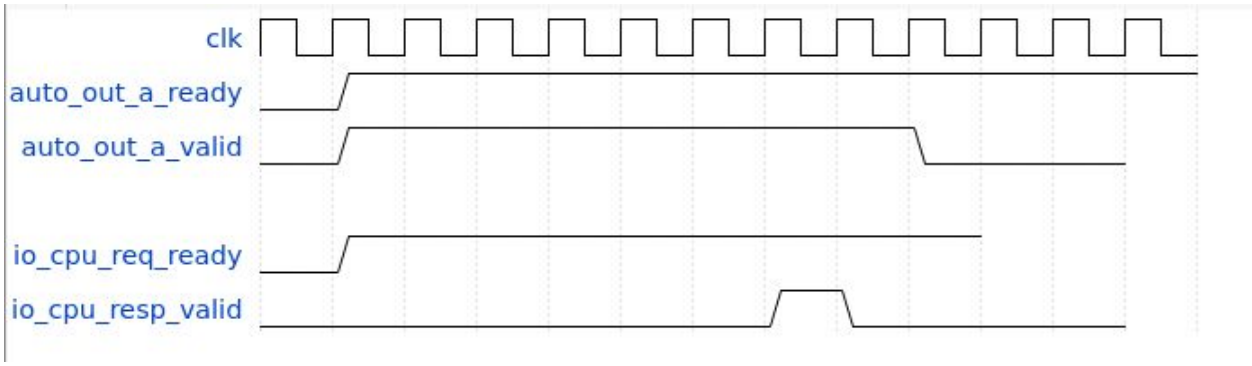
after (output io_cpu_resp_valid==1) +2ns+2ns

io_cpu_req_ready=1;



If WRITE req:

Then io_cpu_req_ready=1 ;//all time

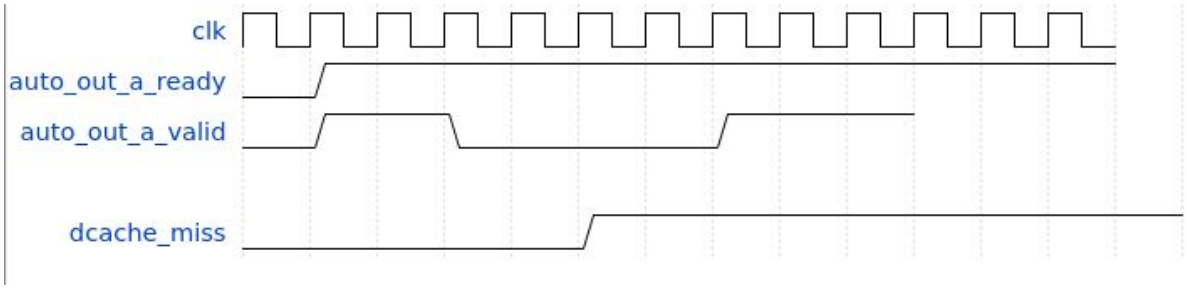


4. Output signal: auto_out_a_valid

when (auto_out_a_ready==1 && auto_out_a_valid==1)+2 ns

then auto_out_a_valid=0;

when miss+2ns:auto_out_a_valid==1;



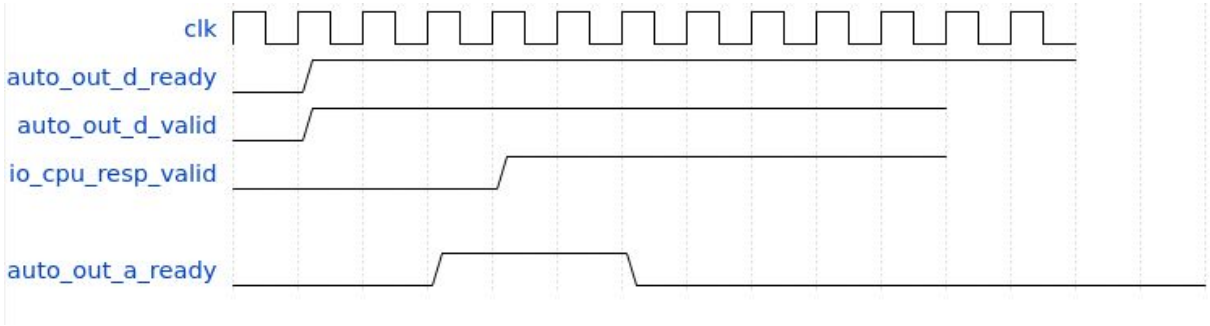
5. Output signal: auto_out_d_ready

when (auto_out_d_ready==1 && auto_out_d_valid==1)+2 ns

then auto_out_a_ready=1;

when (io_cpu_resp_valid==1)+2ns:

Then auto_out_a_ready=0;



6. auto_out_a_bits_opcode: READ:1

WRITE:1

7. output [2:0] auto_out_a_bits_param:READ:000

WRITE:000

8.output [3:0] auto_out_a_bits_size: READ:{{2'd0},io_cpu_req_bits_typ[1:0]}

WRITE:01

9.output auto_out_a_bits_source:READ:1

WRITE:1

10.output [31:0] auto_out_a_bits_address:io_cpu_req_bits_addr

12. output [63:0] auto_out_a_bits_data:READ:0

WRITE:0

13.output auto_out_a_bits_corrupt:0

******B Channel

14. output auto_out_b_ready

****** C Channel

15.(i) Voluntary Release/Write Back:c_opcode:111

```
output                  auto_out_c_valid,
output [2:0]   auto_out_c_bits_opcode:111,
output [2:0]   auto_out_c_bits_param:001,
output [3:0]   auto_out_c_bits_size:6,
output                  auto_out_c_bits_source:0,
output [31:0] auto_out_c_bits_address:,
output                  auto_out_c_bits_corrupt:0
```

(ii) ProbeACk (No Data): C_opcode: 100 //works only for 1 cc

```
output                  auto_out_c_valid,
output [2:0]   auto_out_c_bits_opcode:100,
output [2:0]   auto_out_c_bits_param:010,
output [3:0]   auto_out_c_bits_size:6,
output                  auto_out_c_bits_source:0,
output [31:0] auto_out_c_bits_address:auto_out_b_bits_address,
output                  auto_out_c_bits_corrupt:0
```

(ii) ProbeACkData (64*8bit Data transfer in 8 cc): C_opcode: 101 //works in 8 cc +2 cc then auto_out_c_valid=0 again

```
output                  auto_out_c_valid,
output [2:0]   auto_out_c_bits_opcode:101,
output [2:0]   auto_out_c_bits_param:000,
output [3:0]   auto_out_c_bits_size:6,
output                  auto_out_c_bits_source:0,
output [31:0] auto_out_c_bits_address:auto_out_b_bits_address,
output                  auto_out_c_bits_corrupt:0
```