1、SRIO和DDR连接示意图 dsp\_top.h



2、DSP调用DDR的nb\_transport\_fw()函数

bool DMAC::SRIOTRANS2OTHER(tlm::tlm\_phase PHASE, DMAC\_CMD CMD,unsigned int CH\_ID, unsigned char\* DATA\_PTR , sc\_time DELAY){

tlm::tlm\_generic\_payload p\_trans2other;

p\_trans2other.set\_data\_ptr(DATA\_PTR);

p\_trans2other.set\_address(CMD);

p\_trans2other.set\_data\_length(CH\_ID); //use data length to label the core id

p\_trans2other.set\_streaming\_width(1); //use streaming\_width 1 to label the SRIO transport

init\_socket2ddr->nb\_transport\_fw(p\_trans2other, PHASE, DELAY);//trans2ddr

if(p\_trans2other.get\_response\_status() == tlm::TLM\_OK\_RESPONSE){

return true;

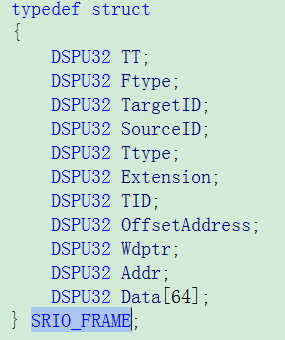
}else{

return false;

}

}

其中DATA\_PTR数据格式为SRIO\_FRAME



3、DDR调用DSP的nb\_transport\_fw()函数

if(!srio\_rd\_data\_vec[coreid].empty() && srio\_rd\_data\_vec[coreid][0].second == 0)

{

//last accept, trans new if there is

p\_trans2core.set\_streaming\_width(1); //use streaming\_width 1 to label the SRIO transport

p\_trans2core.set\_data\_ptr((unsigned char\*)&srio\_rd\_data\_vec[coreid][0].first);

init\_socket2dmac0->nb\_transport\_fw(p\_trans2core, ph, tt);

if(p\_trans2core.get\_response\_status() == tlm::TLM\_OK\_RESPONSE ){

srio\_rd\_data\_vec[coreid].erase(srio\_rd\_data\_vec[coreid].begin()); //accept then erase

}

}

srio\_rd\_data\_vec[]的数据结构为



第一个SRIO\_FRAME为传输的帧，第二个int为延时。