#include "hls\_design\_meta.h"

const Port\_Property HLS\_Design\_Meta::port\_props[]={

Port\_Property("ap\_clk", 1, hls\_in, -1, "", "", 1),

Port\_Property("ap\_rst", 1, hls\_in, -1, "", "", 1),

Port\_Property("ap\_start", 1, hls\_in, -1, "", "", 1),

Port\_Property("ap\_done", 1, hls\_out, -1, "", "", 1),

Port\_Property("ap\_idle", 1, hls\_out, -1, "", "", 1),

Port\_Property("ap\_ready", 1, hls\_out, -1, "", "", 1),

Port\_Property("a\_address0", 4, hls\_out, 0, "ap\_memory", "mem\_address", 1),

Port\_Property("a\_ce0", 1, hls\_out, 0, "ap\_memory", "mem\_ce", 1),

Port\_Property("a\_q0", 8, hls\_in, 0, "ap\_memory", "mem\_dout", 1),

Port\_Property("a\_address1", 4, hls\_out, 0, "ap\_memory", "mem\_address", 1),

Port\_Property("a\_ce1", 1, hls\_out, 0, "ap\_memory", "mem\_ce", 1),

Port\_Property("a\_q1", 8, hls\_in, 0, "ap\_memory", "mem\_dout", 1),

Port\_Property("b\_address0", 4, hls\_out, 1, "ap\_memory", "mem\_address", 1),

Port\_Property("b\_ce0", 1, hls\_out, 1, "ap\_memory", "mem\_ce", 1),

Port\_Property("b\_q0", 8, hls\_in, 1, "ap\_memory", "mem\_dout", 1),

Port\_Property("b\_address1", 4, hls\_out, 1, "ap\_memory", "mem\_address", 1),

Port\_Property("b\_ce1", 1, hls\_out, 1, "ap\_memory", "mem\_ce", 1),

Port\_Property("b\_q1", 8, hls\_in, 1, "ap\_memory", "mem\_dout", 1),

Port\_Property("res\_address0", 4, hls\_out, 2, "ap\_memory", "mem\_address", 1),

Port\_Property("res\_ce0", 1, hls\_out, 2, "ap\_memory", "mem\_ce", 1),

Port\_Property("res\_we0", 1, hls\_out, 2, "ap\_memory", "mem\_we", 1),

Port\_Property("res\_d0", 16, hls\_out, 2, "ap\_memory", "mem\_din", 1),

};

const char\* HLS\_Design\_Meta::dut\_name = "matrixmul";