`include "axi\_bus.sv"

module core2axi\_wrap

#(

parameter AXI\_ADDR\_WIDTH = 32,

parameter AXI\_DATA\_WIDTH = 32,

parameter AXI\_USER\_WIDTH = 6,

parameter AXI\_ID\_WIDTH = 6,

parameter REGISTERED\_GRANT = "FALSE"

)

(

input logic clk\_i,

input logic rst\_ni,

input logic data\_req\_i,

output logic data\_gnt\_o,

output logic data\_rvalid\_o,

input logic [AXI\_ADDR\_WIDTH-1:0] data\_addr\_i,

input logic data\_we\_i,

input logic [3:0] data\_be\_i,

output logic [31:0] data\_rdata\_o,

input logic [31:0] data\_wdata\_i,

AXI\_BUS.Master master

);

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\* AXI2APB WRAPER \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

core2axi

#(

.AXI4\_ADDRESS\_WIDTH ( AXI\_ADDR\_WIDTH ),

.AXI4\_RDATA\_WIDTH ( AXI\_DATA\_WIDTH ),

.AXI4\_WDATA\_WIDTH ( AXI\_DATA\_WIDTH ),

.AXI4\_ID\_WIDTH ( AXI\_ID\_WIDTH ),

.AXI4\_USER\_WIDTH ( AXI\_USER\_WIDTH ),

.REGISTERED\_GRANT ( REGISTERED\_GRANT )

)

core2axi\_i

(

.clk\_i ( clk\_i ),

.rst\_ni ( rst\_ni ),

.data\_req\_i ( data\_req\_i ),

.data\_gnt\_o ( data\_gnt\_o ),

.data\_rvalid\_o ( data\_rvalid\_o ),

.data\_addr\_i ( data\_addr\_i ),

.data\_we\_i ( data\_we\_i ),

.data\_be\_i ( data\_be\_i ),

.data\_rdata\_o ( data\_rdata\_o ),

.data\_wdata\_i ( data\_wdata\_i ),

.aw\_id\_o ( master.aw\_id ),

.aw\_addr\_o ( master.aw\_addr ),

.aw\_len\_o ( master.aw\_len ),

.aw\_size\_o ( master.aw\_size ),

.aw\_burst\_o ( master.aw\_burst ),

.aw\_lock\_o ( master.aw\_lock ),

.aw\_cache\_o ( master.aw\_cache ),

.aw\_prot\_o ( master.aw\_prot ),

.aw\_region\_o ( master.aw\_region ),

.aw\_user\_o ( master.aw\_user ),

.aw\_qos\_o ( master.aw\_qos ),

.aw\_valid\_o ( master.aw\_valid ),

.aw\_ready\_i ( master.aw\_ready ),

.w\_data\_o ( master.w\_data ),

.w\_strb\_o ( master.w\_strb ),

.w\_last\_o ( master.w\_last ),

.w\_user\_o ( master.w\_user ),

.w\_valid\_o ( master.w\_valid ),

.w\_ready\_i ( master.w\_ready ),

.b\_id\_i ( master.b\_id ),

.b\_resp\_i ( master.b\_resp ),

.b\_valid\_i ( master.b\_valid ),

.b\_user\_i ( master.b\_user ),

.b\_ready\_o ( master.b\_ready ),

.ar\_id\_o ( master.ar\_id ),

.ar\_addr\_o ( master.ar\_addr ),

.ar\_len\_o ( master.ar\_len ),

.ar\_size\_o ( master.ar\_size ),

.ar\_burst\_o ( master.ar\_burst ),

.ar\_lock\_o ( master.ar\_lock ),

.ar\_cache\_o ( master.ar\_cache ),

.ar\_prot\_o ( master.ar\_prot ),

.ar\_region\_o ( master.ar\_region ),

.ar\_user\_o ( master.ar\_user ),

.ar\_qos\_o ( master.ar\_qos ),

.ar\_valid\_o ( master.ar\_valid ),

.ar\_ready\_i ( master.ar\_ready ),

.r\_id\_i ( master.r\_id ),

.r\_data\_i ( master.r\_data ),

.r\_resp\_i ( master.r\_resp ),

.r\_last\_i ( master.r\_last ),

.r\_user\_i ( master.r\_user ),

.r\_valid\_i ( master.r\_valid ),

.r\_ready\_o ( master.r\_ready )

);

endmodule