```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
library work;

ENTITY testbench IS
END testbench;

ARCHITECTURE testbench_arch OF testbench IS
   --initialization and declaration of inputs
signal ren0_tb:   std_logic;
signal ren1_tb:   std_logic;
signal ren2_tb:   std_logic;
signal ren3_tb:   std_logic;
signal ren4_tb:   std_logic;
signal ren5_tb:   std_logic;
signal ren6_tb:   std_logic;
signal ren7_tb:   std_logic;
signal ren8_tb:   std_logic;
signal ren9_tb:   std_logic;
signal renA_tb:   std_logic;
signal renB_tb:   std_logic;
signal renC_tb:   std_logic;
signal renD_tb:   std_logic;
signal renE_tb:   std_logic;
signal renF_tb:   std_logic;
signal clr_tb  :  std_logic;
signal clk_tb  :  std_logic;
signal R0out_tb   :  std_logic;
signal R1out_tb   :  std_logic;
signal R2out_tb   :  std_logic;
signal R3out_tb   :  std_logic;
signal R4out_tb   :  std_logic;
signal R5out_tb   :  std_logic;
signal R6out_tb   :  std_logic;
signal R7out_tb   :  std_logic;
signal R8out_tb   :  std_logic;
signal R9out_tb   :  std_logic;
signal R10out_tb  :std_logic;
signal R11out_tb  :std_logic;
signal R12out_tb  :std_logic;
signal R13out_tb  :std_logic;
signal R14out_tb  :std_logic;
signal R15out_tb  :std_logic;
signal Zhighout_tb:  std_logic;
signal Zlowout_tb :std_logic;
signal InPortout_tb: std_logic;
signal HIin_tb    :std_logic;
signal PCin_tb    :std_logic;
signal LOin_tb    :std_logic;
signal Empty_tb   :  STD_LOGIC_VECTOR(31 DOWNTO 0);
signal readin_tb     :std_logic;
signal MDRin_tb   :  std_logic;
signal Mdatain_tb :STD_LOGIC_VECTOR(31 DOWNTO 0);
signal HIoutEn_tb :std_logic;
signal LOoutEn_tb :std_logic;
signal PCoutEn_tb :std_logic;
signal MDRoutEn_tb:  std_logic;
signal CoutEn_tb  :std_logic;
signal reny_tb    :std_logic;
signal operation_tb: STD_LOGIC_VECTOR(3 DOWNTO 0);
signal renz_tb    :std_logic;
```

```
63    signal rout2_tb       :std_logic_vector(31 downto 0);
64    signal rout3_tb       :std_logic_vector(31 downto 0);
65    signal rout4_tb       :std_logic_vector(31 downto 0);
66    signal rout5_tb       :std_logic_vector(31 downto 0);
67    signal rout6_tb       :std_logic_vector(31 downto 0);
68    signal rout7_tb       :std_logic_vector(31 downto 0);
69    signal rout8_tb       :std_logic_vector(31 downto 0);
70    signal rout9_tb       :std_logic_vector(31 downto 0);
71    signal routa_tb       :std_logic_vector(31 downto 0);
72    signal routb_tb       :std_logic_vector(31 downto 0);
73    signal routc_tb       :std_logic_vector(31 downto 0);
74    signal routd_tb       :std_logic_vector(31 downto 0);
75    signal route_tb       :std_logic_vector(31 downto 0);
76    signal routf_tb       :std_logic_vector(31 downto 0);
77    signal rout0_tb       :std_logic_vector(31 downto 0);
78    signal rout1_tb       :std_logic_vector(31 downto 0);
79    signal busmuxout_tb   :std_logic_vector(31 downto 0);
80    signal PCout_tb       :std_logic_vector(31 downto 0);
81    signal Zhigh_tb       :std_logic_vector(31 downto 0);
82    signal LOout_tb       :std_logic_vector(31 downto 0);
83    signal HIout_tb       :std_logic_vector(31 downto 0);
84    signal MDRout_tb   :std_logic_vector(31 downto 0);
85    signal routy_tb :     STD_LOGIC_VECTOR(31 DOWNTO 0);
86    signal Zlow_tb     :std_logic_vector(31 downto 0);
87
88
89    TYPE  State IS(default, Reg_load1, Reg_load2, Reg_load3, T0, T1, T2, T3, T4, T5);
90    SIGNAL Present_state: State := default;
91
92    component ELEC374
93          PORT
94       (
95          ren0      :    IN STD_LOGIC;
96          ren1      :    IN STD_LOGIC;
97          ren2      :    IN STD_LOGIC;
98          ren3      :    IN STD_LOGIC;
99          ren4      :    IN STD_LOGIC;
100         ren5      :    IN STD_LOGIC;
101         ren6      :    IN STD_LOGIC;
102         ren7      :    IN STD_LOGIC;
103         ren8      :    IN STD_LOGIC;
104         ren9      :    IN STD_LOGIC;
105         renA      :    IN STD_LOGIC;
106         renB      :    IN STD_LOGIC;
107         renC      :    IN STD_LOGIC;
108         renD      :    IN STD_LOGIC;
109         renE      :    IN STD_LOGIC;
110         renF      :    IN STD_LOGIC;
111         clr       :    IN STD_LOGIC;
112         clk       :    IN STD_LOGIC;
113         R0out     :    IN STD_LOGIC;
114         R1out     :    IN STD_LOGIC;
115         R2out     :    IN STD_LOGIC;
116         R3out     :    IN STD_LOGIC;
117         R4out     :    IN STD_LOGIC;
118         R5out     :    IN STD_LOGIC;
119         R6out     :    IN STD_LOGIC;
120         R7out     :    IN STD_LOGIC;
121         R8out     :    IN STD_LOGIC;
122         R9out     :    IN STD_LOGIC;
123         R10out       :    IN STD_LOGIC;
124         R11out       :    IN STD_LOGIC;
```

```
125         R12out      :   IN STD_LOGIC;
126         R13out      :   IN STD_LOGIC;
127         R14out      :   IN STD_LOGIC;
128         R15out      :   IN STD_LOGIC;
129         Zhighout    :   IN STD_LOGIC;
130         Zlowout     :   IN STD_LOGIC;
131         InPortout       :   IN STD_LOGIC;
132         HIin     :   IN STD_LOGIC;
133         PCin     :   IN STD_LOGIC;
134
135         LOin     :   IN STD_LOGIC;
136         Empty    :   IN STD_LOGIC_VECTOR(31 DOWNTO 0);
137         ReadIn :  IN  STD_LOGIC;
138         MDRin    :   IN STD_LOGIC;
139         Mdatain      :   IN STD_LOGIC_VECTOR(31 DOWNTO 0);
140         HIoutEn      :   IN STD_LOGIC;
141         LOoutEn      :   IN STD_LOGIC;
142         PCoutEn      :   IN STD_LOGIC;
143         MDRoutEn     :   IN STD_LOGIC;
144         CoutEn       :   IN STD_LOGIC;
145         reny     :   IN STD_LOGIC;
146         operation:   IN STD_LOGIC_VECTOR(3 DOWNTO 0);
147         renz     :   IN STD_LOGIC;
148         rout2    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
149         rout3    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
150         rout4    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
151         rout5    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
152         rout6    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
153         rout7    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
154         rout8    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
155         rout9    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
156         routa    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
157         routb    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
158         routc    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
159         routd    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
160         route    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
161         routf    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
162         rout0    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
163         rout1    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
164         busmuxout       :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
165         PCout    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
166         Zhigh    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
167         Zlow     :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
168         LOout    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
169         HIout    :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
170         MDRout       :   INOUT STD_LOGIC_VECTOR(31 DOWNTO 0);
171         routy    : INOUT STD_LOGIC_VECTOR(31 DOWNTO 0)
172
173      );
174   end component ELEC374;
175
176   BEGIN
177
178   DUT: ELEC374 PORT MAP(
179   ren0        =>ren0_tb,
180   ren1        =>ren1_tb,
181   ren2        =>ren2_tb,
182   ren3        =>ren3_tb,
183   ren4        =>ren4_tb,
184   ren5        =>ren5_tb,
185   ren6        =>ren6_tb,
186   ren7        =>ren7_tb,
```

```
187   ren8        =>ren8_tb,
188   ren9        =>ren9_tb,
189   renA        =>renA_tb,
190   renB        =>renB_tb,
191   renC        =>renC_tb,
192   renD        =>renD_tb,
193   renE        =>renE_tb,
194   renF        =>renF_tb,
195   clr         =>clr_tb,
196   clk         =>clk_tb,
197   R0out       =>R0out_tb,
198   R1out       =>R1out_tb,
199   R2out       =>R2out_tb,
200   R3out       =>R3out_tb,
201   R4out       =>R4out_tb,
202   R5out       =>R5out_tb,
203   R6out       =>R6out_tb,
204   R7out       =>R7out_tb,
205   R8out       =>R8out_tb,
206   R9out       =>R9out_tb,
207   R10out      =>R10out_tb,
208   R11out      =>R11out_tb,
209   R12out      =>R12out_tb,
210   R13out      =>R13out_tb,
211   R14out      =>R14out_tb,
212   R15out      =>R15out_tb,
213   Zhighout =>Zhighout_tb,
214   Zlowout     =>Zlowout_tb,
215   InPortout   =>InPortout_tb,
216   HIin     =>HIin_tb,
217   PCin     =>PCin_tb,
218   LOin     =>LOin_tb,
219   Empty    =>Empty_tb,
220   ReadIn      =>readin_tb,
221   MDRin    =>MDRin_tb,
222   Mdatain     =>Mdatain_tb,
223   HIoutEn     =>HIoutEn_tb,
224   LOoutEn     =>LOoutEn_tb,
225   PCoutEn     =>PCoutEn_tb,
226   MDRoutEn =>MDRoutEn_tb,
227   CoutEn      =>CoutEn_tb,
228   reny     =>reny_tb,
229   operation   =>operation_tb,
230   renz     =>renz_tb,
231   rout2    =>rout2_tb,
232   rout3    =>rout3_tb,
233   rout4    =>rout4_tb,
234   rout5    =>rout5_tb,
235   rout6    =>rout6_tb,
236   rout7    =>rout7_tb,
237   rout8    =>rout8_tb,
238   rout9    =>rout9_tb,
239   routa    =>routa_tb,
240   routb    =>routb_tb,
241   routc    =>routc_tb,
242   routd    =>routd_tb,
243   route    =>route_tb,
244   routf    =>routf_tb,
245   rout0    =>rout0_tb,
246   rout1    =>rout1_tb,
247   busmuxout   =>busmuxout_tb,
248   PCout    =>PCout_tb,
```

```
249   Zhigh    =>Zhigh_tb,
250   Zlow     =>Zlow_tb,
251   LOout    =>LOout_tb,
252   HIout    =>HIout_tb,
253   MDRout      =>MDRout_tb,
254   routy => routy_tb
255      );
256
257   clk_process : process
258      begin
259            clk_tb<='0','1' after 10 ns;
260            wait for 20 ns;
261   end process clk_process;
262
263   process (clk_tb)
264
265   begin
266      if (clk_tb'event and clk_tb ='1') then
267         case Present_state is
268            when Default =>
269                  Present_state<=Reg_load1;
270            when reg_load1 =>
271                  present_state<=Reg_load2;
272            when reg_load2 =>
273                  present_state<=Reg_load3;
274            when reg_load3 =>
275                  present_state<=T0;
276            when T0 =>
277                  present_state<=T1;
278            when T1 =>
279                  present_state<=T2;
280            when T2 =>
281                  present_state<=T3;
282            when T3 =>
283                  present_state<=T4;
284            when T4 =>
285                  present_state<=T5;
286            when others =>
287         end case;
288      end if;
289   end process;
290
291   Process(present_state)
292   begin
293         CASE present_state IS
294               WHEN default =>
295                  PCin_tb<='0'; Zlowout_tb<='0'; mdroutEn_tb<='0';
296                  R1Out_tb<='0';R2Out_tb<='0';R3Out_tb<='0';
297                  PCin_tb <= '0'; MDRin_tb <= '0';
298                  ren1_tb<='0'; readin_tb<='0';
299                  clr_tb <= '1', '0' after 4 ns;
300                  renz_tb <= '0';
301                  ren0_tb <= '0';
302                  ren1_tb <= '0';
303                  ren2_tb <= '0';
304                  ren3_tb <= '0';
305                  ren4_tb <= '0';
306                  ren5_tb <= '0';
307                  ren6_tb <= '0';
308                  ren7_tb <= '0';
309                  ren8_tb <= '0';
310                  ren9_tb <= '0';
```

```
311                  renA_tb <= '0';
312                  renB_tb <= '0';
313                  renC_tb <= '0';
314                  renD_tb <= '0';
315                  renE_tb <= '0';
316                  renF_tb <= '0';
317                  R0out_tb <= '0';
318                  R1out_tb <= '0';
319                  R2out_tb <= '0';
320                  R3out_tb <= '0';
321                  R4out_tb <= '0';
322                  R5out_tb <= '0';
323                  R6out_tb <= '0';
324                  R7out_tb <= '0';
325                  R8out_tb <= '0';
326                  R9out_tb <= '0';
327                  R10out_tb <= '0';
328                  R11out_tb <= '0';
329                  R12out_tb <= '0';
330                  R13out_tb <= '0';
331                  R14out_tb <= '0';
332                  R15out_tb <= '0';
333                  Zhighout_tb <= '0';
334                  InPortout_tb <= '0';
335                  HIin_tb <= '0';
336
337                  Loin_tb <='0';
338                  HIoutEn_tb <= '0';
339                  LOoutEn_tb<='0';
340                  PcoutEn_tb <= '0';
341                  CoutEn_tb<='0';
342                  reny_tb <= '0';
343                  operation_tb <= "0000";
344                  Mdatain_tb<=x"00000012";
345
346                  --routy_tb <= x"00000000";
347                  empty_tb <= x"00000000";
348                  operation_tb<="0000";
349
350
351
352               when reg_load1 =>
353                  mdatain_tb<= x"00006666";
354                  readin_tb<='1','0' after 10 ns;
355                  mdrin_tb<='1','0'after 10 ns;
356                  mdroutEn_tb<='1', '0' after 10 ns ;
357
358                  ren1_tb<='1', '0' after 10 ns ;
359
360               when reg_load2 =>
361                  mdatain_tb <= x"00000015";
362                  readin_tb<='1','0' after 10 ns;
363                  mdrin_tb<='1','0'after 10 ns;
364                  mdroutEn_tb<='1', '0' after 10 ns ;
365
366
367                  ren2_tb<='1', '0' after 10 ns ;
368
369               when reg_load3 =>
370                  mdatain_tb <= x"00000002";
371                  readin_tb<='1','0' after 10 ns;
372                  mdrin_tb<='1','0'after 10 ns;
```

```
373                  mdroutEn_tb<='1', '0' after 10 ns ;
374                  ren3_tb<='1', '0' after 10 ns ;
375
376               when T0 =>
377                  PCoutEn_tb<='1', '0' after 10 ns ;
378                  renz_tb<='1', '0' after 10 ns ;
379
380
381               when T1 =>
382                  Zlowout_tb<='1', '0' after 10 ns ; PCin_tb<='1', '0' after 10 ns ;
383                  readin_tb<='1', '0' after 10 ns ; MDRin_tb<='1', '0' after 10 ns ;
384                  --Mdatain_tb<=x"294c0000";
385
386               when T2 =>
387
388                  MDRoutEn_tb<='1', '0' after 10 ns ;
389
390               when T3 =>
391                  --
392                  r2out_tb<='1', '0' after 10 ns;
393                  reny_tb<='1', '0' after 10 ns ; --readin_tb<='1', '0' after 10 ns
    ;MDRin_tb<='1', '0' after 10 ns ;
394               when T4 =>
395                  r3out_tb<='1', '0' after 10 ns ; renz_tb<='1', '0' after 10 ns ;
396                  operation_tb<="0100";
397               when T5 =>
398                  zlowout_tb<='1', '0' after 10 ns ; ren1_tb<='1', '0' after 10 ns ;
    --operation_tb<="1111"; renz_tb<='0';
399               when others =>
400
401            end case;
402         end process;
403
404
405
406   end architecture testbench_arch;
407
```