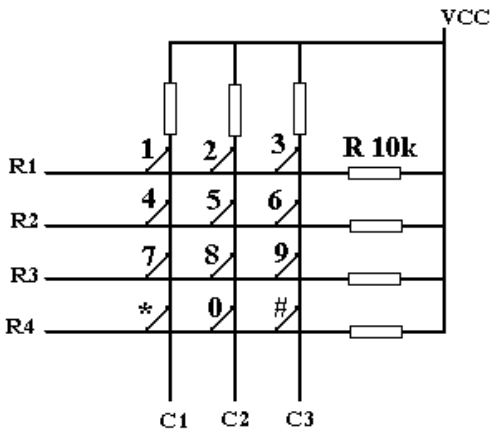
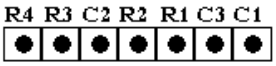
การทดลองที่ 8 วงจรเข้ารหัสคีย์บอร์ด





รูปที่ 1 วงจรคีย์บอร์ดและตำแหน่ง Connector

รูปที่ 2 บล๊อกไดอะแกรมของวงจรเข้ารหัสคีย์บอร์ดขนาด 3x4

VHDL Module ของวงจรเข้ารหัสคีย์บอร์ด

```
library IEEE;
use IEEE.std_logic_1164.all;
entity srow2 is
    port (pclk : in std_logic;
          peo : out std_logic;
          py  : out std_logic_vector(3 downto 0);
          kin : in std_logic_vector(2 downto 0);
          kout : out std_logic_vector(3 downto 0));
end srow2;
--
architecture srow_arch of srow2 is
--
component srow is
    port (pclk : in std_logic;
          peo : out std_logic;
          py  : out std_logic_vector(3 downto 0);
          kin : in std_logic_vector(2 downto 0);
          kout : out std_logic_vector(3 downto 0));
end component;
--
component reg3 is
    Port ( x : in std_logic_vector(3 downto 0);
               clk, ld : in std_logic;
                y : out std_logic_vector(3 downto 0));
end component;
--

signal bus1 : std_logic_vector (3 downto 0);
signal line : std_logic;

begin
  c1: srow port map(pclk, line, bus1, kin, kout);
  c2: reg3 port map(bus1, pclk, not (line), py);
  peo <= line;
end srow_arch;
```

VHDL Module ของวงจรรีจิสเตอร์

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reg3  is
    Port ( x : in std_logic_vector(3 downto 0);
               clk, ld : in std_logic;
                y : out std_logic_vector(3 downto 0));
end reg3;

architecture Behavioral of reg3 is
begin
  process (clk, ld)
  begin
      if clk = '1' and clk'event then
         if ld = '0' then
             y <= x;
         end if;
     end if;
  end process;
end Behavioral;
```

VHDL Module ของวงจรเข้ารหัสคีย์บอร์ดที่ไม่มีรีจิสเตอร์

```
library IEEE;
use IEEE.std_logic_1164.all;
entity srow is
    port (pclk : in std_logic;
          peo : out std_logic;
          py  : out std_logic_vector(3 downto 0);
          kin : in std_logic_vector(2 downto 0);
          kout : out std_logic_vector(3 downto 0));
end srow;
--
architecture srow_arch of srow is
--
component decoder1  is
    Port ( s : in std_logic_vector(1 downto 0);
           y : out std_logic_vector(3 downto 0));
end component;
--
component binco is
    port (clk : in std_logic;
            y : out std_logic_vector(1 downto 0));
end component;
--
component encode3 is
    Port ( xi : in std_logic_vector(2 downto 0);
           y : out std_logic_vector(1 downto 0);
           eo : out std_logic);
end component;
signal bus1 : std_logic_vector (1 downto 0);
begin
  c1: binco port map(pclk, bus1);
  c2: decoder1 port map(bus1, kout);
  c3: encode3 port map(kin, py(1 downto 0), peo);
  py(3 downto 2) <= bus1;
end srow_arch;
```

วงจรนับ

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_unsigned.ALL;
entity binco is
    port (clk : in std_logic;
            y : out std_logic_vector(1 downto 0));
end binco;
architecture binco_beh of binco is
type state_type is (a, b, c, d);
signal state: state_type;
begin
  process (clk)
  begin
    if clk = '1' and clk'event then
       case state is
              when a =>  state <= b;
              when b =>  state <= c;
              when c =>  state <= d;
              when d =>  state <= a;
       end case;
```

```
    end if;
end process;
process (state)
   begin
      case state is
      when a => y <= "00";
      when b => y <= "01";
      when c => y <= "10";
      when d => y <= "11";
      end case;
   end process;
end binco_beh;
```

วงจรถอดรหัส

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity decoder1 is
    Port ( s : in std_logic_vector(1 downto 0);
           y : out std_logic_vector(3 downto 0));
end decoder1;

architecture Behavioral of decoder1 is

begin
      process (s)
      begin
         case s is
               when "00" =>
                    y(0) <= '0';
                    y(1) <= '1';
                    y(2) <= '1';
                    y(3) <= '1';
                  when "01" =>
                    y(0) <= '1';
                    y(1) <= '0';
                    y(2) <= '1';
                    y(3) <= '1';
                  when "10" =>
                    y(0) <= '1';
                    y(1) <= '1';
                    y(2) <= '0';
                    y(3) <= '1';
                  when others =>
                    y(0) <= '1';
                    y(1) <= '1';
                    y(2) <= '1';
                    y(3) <= '0';
            end case;
      end process;
end Behavioral;
```

วงจรเข้ารหัส

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
```

```
entity encode3 is
    Port ( xi : in std_logic_vector(2 downto 0);
           y : out std_logic_vector(1 downto 0);
           eo : out std_logic);
end encode3;

architecture Behavioral of encode3 is

begin
      process (xi)
      begin
        if    xi(2) = '0' then
                y <= "10";
              eo <= '1';
        elsif xi(1) = '0' then
                y <= "01";
              eo <= '1';
        elsif xi(0) = '0' then
                y <= "00";
              eo <= '1';
        else
               y <= "00";
             eo <= '0';
        end if;
      end process;
end Behavioral;
```