



COL215 DIGITAL LOGIC AND SYSTEM DESIGN

VHDL – Combinational Logic
Modules

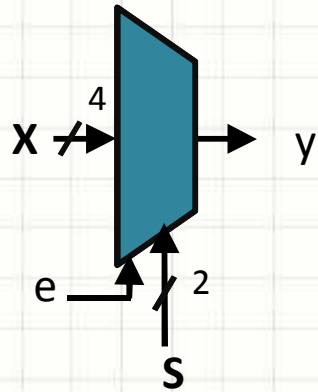
18 August 2017



Outline

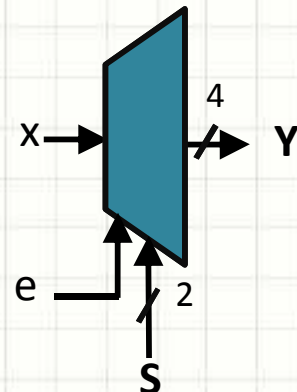
- Combinational logic modules in VHDL
- 3-port switch example

Combinational Modules



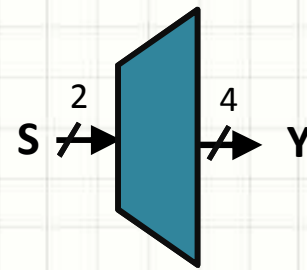
4:1 mux

e	s_1s_0	y
0	- -	0
1	00	x_0
1	01	x_1
1	10	x_2
1	11	x_3



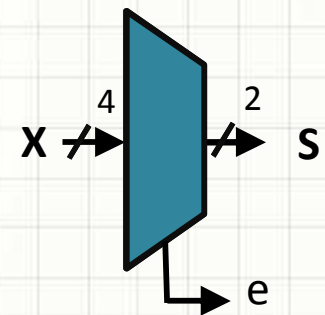
1:4 de-mux

e	s_1s_0	$y_3y_2y_1y_0$
0	- -	0000
1	00	000x
1	01	00x0
1	10	0x00
1	11	x000



2:4 decoder

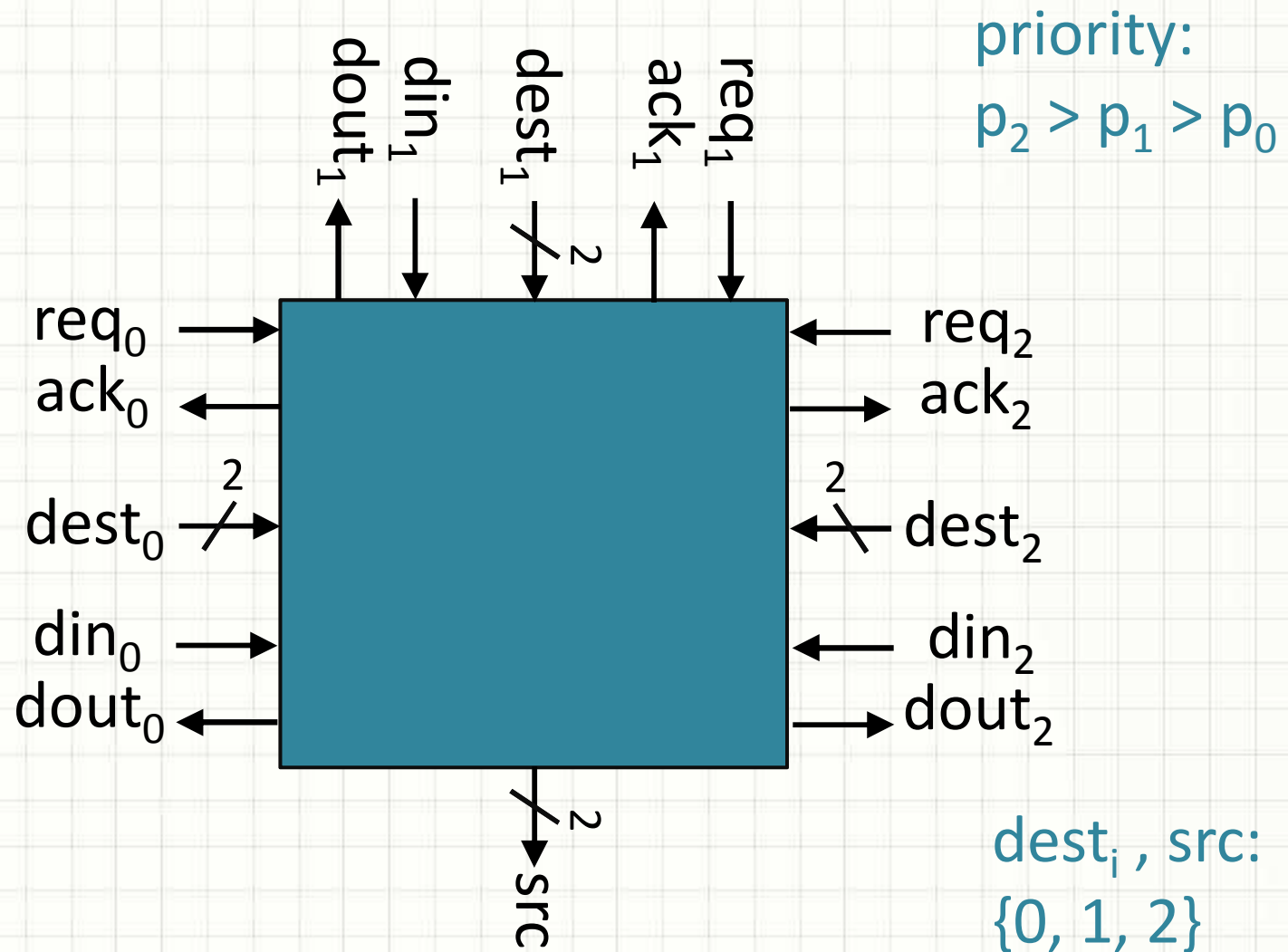
s_1s_0	$y_3y_2y_1y_0$
00	0001
01	0010
10	0100
11	1000



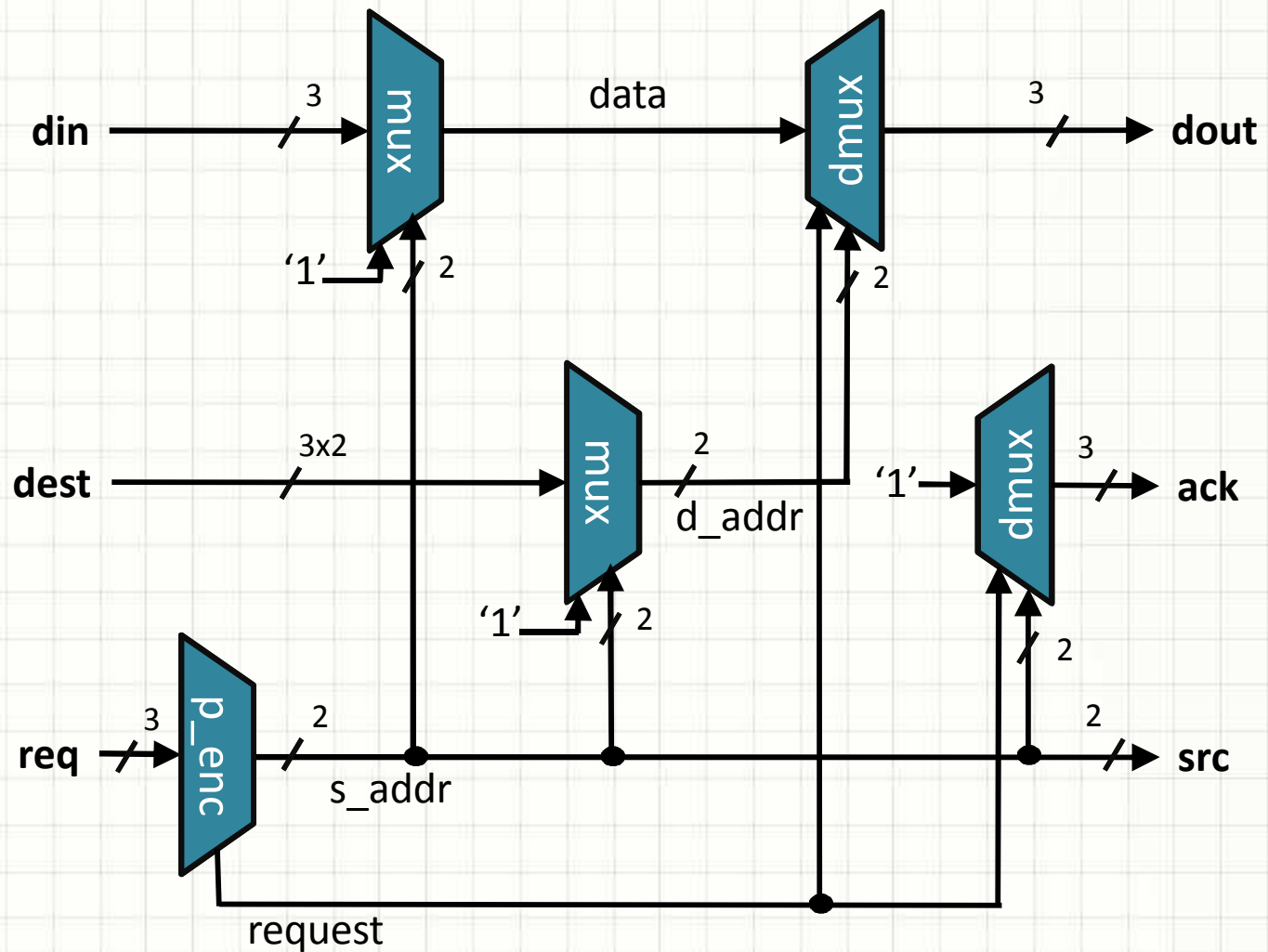
4 input priority encoder

$x_3x_2x_1x_0$	e s_1s_0
0000	0 - -
0001	1 00
001-	1 01
01--	1 10
1---	1 11

Lab exercise 2 : 3-Port Switch

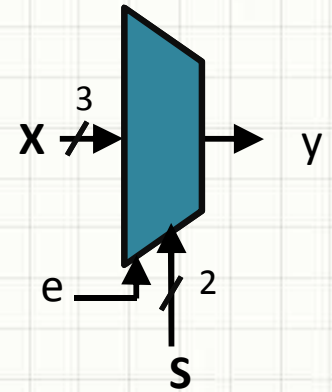


3-Port Switch Design



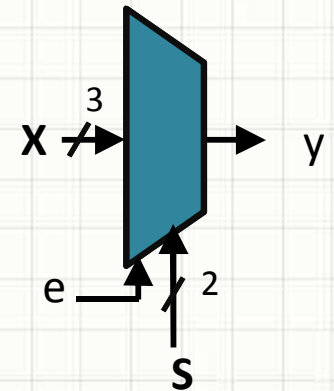
3 : 1 Mux

```
ENTITY mux_3_1 IS
  PORT (X: IN    bit_vector (2 DOWNT0 0);
        S: IN    bit_vector (1 DOWNT0 0);
        e: IN    bit;
        y: OUT bit
  );
END mux_3_1;
```



CASE statement

```
ARCHITECTURE casestmt OF mux_3_1 IS
BEGIN
  PROCESS (S, X, e)
  BEGIN
    IF e = '1' THEN
      CASE S IS
        WHEN "00"      => y <= X(0);
        WHEN "01"      => y <= X(1);
        WHEN OTHERS    => y <= X(2);
      END CASE;
    ELSE y <= '0';
    ENDIF;
  END PROCESS;
END ARCHITECTURE casestmt;
```

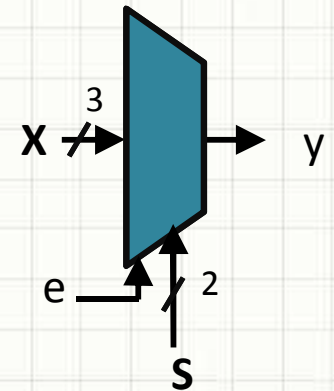


3:1 mux

e	s ₁ s ₀	y
0	- -	0
1	0 0	x ₀
1	0 1	x ₁
1	1 0	x ₂
1	1 1	??

Selected Signal Assignment

```
ARCHITECTURE ssa OF mux_3_1 IS
BEGIN
    SIGNAL t : bit;
    WITH S SELECT
        t <= X(0) WHEN "00",
            X(1) WHEN "01",
            X(2) WHEN OTHERS;
    y <= t AND e;
END ARCHITECTURE ssa;
```

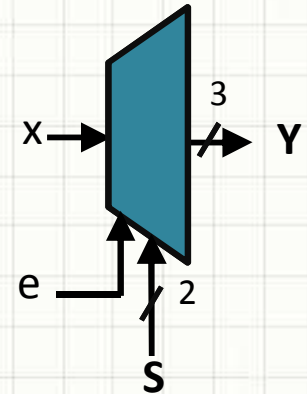


3:1 mux

e	s ₁ s ₀	y
0	- -	0
1	0 0	x ₀
1	0 1	x ₁
1	1 0	x ₂
1	1 1	??

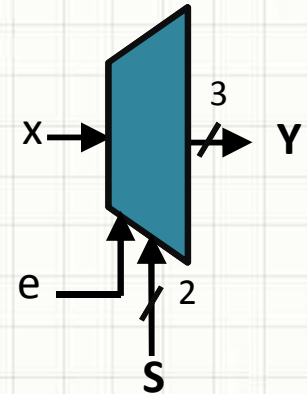
1 to 3 De-mux

```
ENTITY de-mux_1_3 IS
  PORT (x: IN    bit;
        e: IN    bit;
        S: IN    bit_vector (1 DOWNT0 0);
        Y: OUT bit_vector (2 DOWNT0 0)
  );
END de-mux_1_3;
```



CASE statement

```
ARCHITECTURE casestmt OF de-mux_1_3 IS
BEGIN
  PROCESS (S, x, e)
  BEGIN
    IF (x AND e) THEN
      CASE S IS
        WHEN "00"      => Y <= "001";
        WHEN "01"      => Y <= "010";
        WHEN OTHERS    => Y <= "100";
      END CASE;
    ELSE Y <= "000";
    ENDIF;
  END PROCESS;
END ARCHITECTURE casestmt;
```



1:3 de-mux

e	s ₁ s ₀	Y ₂ Y ₁ Y ₀
0	- -	0 0 0
1	0 0	0 0 x
1	0 1	0 x 0
1	1 0	x 0 0
1	1 1	??

Selected Signal Assignment

ARCHITECTURE ssa OF de-mux_1_3 IS
BEGIN

SIGNAL T : bit_vector (2 DOWNT0 0);

Y <= T WHEN (x AND e) ELSE

"000";

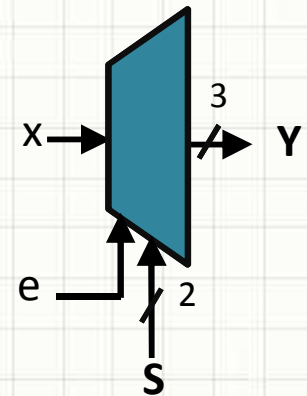
WITH S SELECT

T <= "001" WHEN "00",

"010" WHEN "01",

"100" WHEN OTHERS;

END ARCHITECTURE ssa;



1:3 de-mux

e	s ₁ s ₀	Y ₂ Y ₁ Y ₀
0	- -	0 0 0
1	0 0	0 0 x
1	0 1	0 x 0
1	1 0	x 0 0
1	1 1	??

3 Input Priority Encoder

ENTITY Priority_3 IS

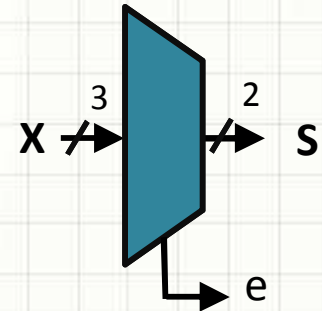
PORT (X: IN bit_vector (2 DOWNT0 0);

S: OUT bit_vector (1 DOWNT0 0);

e: OUT bit

);

END Priority_3;



IF statement

```
ARCHITECTURE ifstmt OF Priority_3 IS  
BEGIN
```

```
  PROCESS (X)
```

```
  BEGIN
```

```
    IF      X(2) = '1' THEN S <= "10"; e <= '1';
```

```
    ELSIF X(1) = '1' THEN S <= "01"; e <= '1';
```

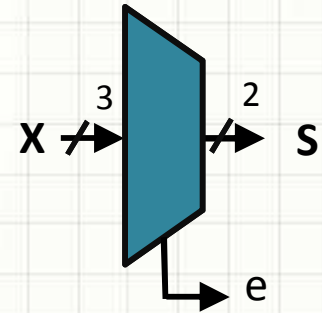
```
    ELSIF X(0) = '1' THEN S <= "00"; e <= '1';
```

```
    ELSE                                S <= "00"; e <= '0';
```

```
  END IF;
```

```
  END PROCESS;
```

```
END ARCHITECTURE ifstmt;
```



3 input priority encoder

$x_2x_1x_0$	$e \ s_1s_0$
0 0 0	0 - -
0 0 1	1 0 0
0 1 -	1 0 1
1 - -	1 1 0

IF statement

ARCHITECTURE ifstmt OF Priority_3 IS
BEGIN

PROCESS (X)

BEGIN

IF X(2) = '1' THEN S <= "10";

ELSIF X(1) = '1' THEN S <= "01";

ELSE S <= "00";

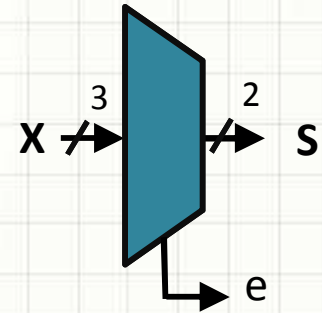
END IF;

IF X = "000" THEN e <= '0'; ELSE e <= '1';

END IF;

END PROCESS;

END ARCHITECTURE ifstmt;



3 input priority encoder

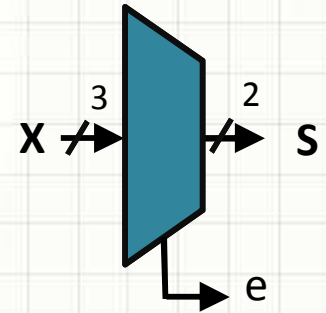
$x_2x_1x_0$	$e \ s_1s_0$
0 0 0	0 - -
0 0 1	1 0 0
0 1 -	1 0 1
1 - -	1 1 0

Conditional signal assignment

```
ARCHITECTURE cond OF Priority_3 IS  
BEGIN
```

```
    S <= "10" WHEN X(2) = '1' ELSE  
        "01" WHEN X(1) = '1' ELSE  
        "00";
```

```
    e <= '0' WHEN X = "000" ELSE '1';  
END ARCHITECTURE cond;
```

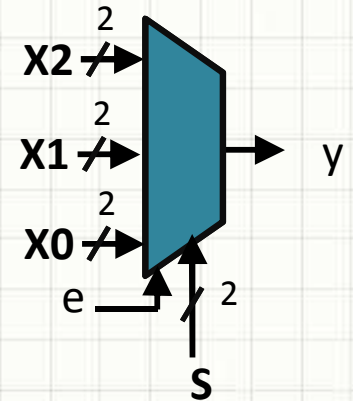


3 input priority
encoder

$x_2x_1x_0$	$e \ s_1s_0$
0 0 0	0 - -
0 0 1	1 0 0
0 1 -	1 0 1
1 - -	1 1 0

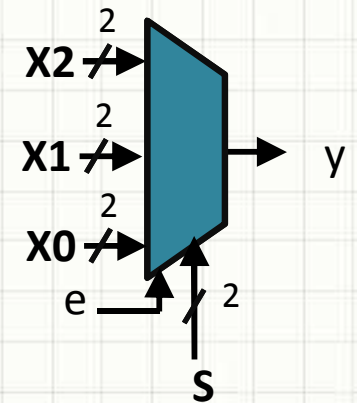
3 : 1 Mux, 2 bit wide

```
ENTITY mux_3_1_2bit IS
  PORT (X2: IN    bit_vector (1 DOWNT0 0);
        X1: IN    bit_vector (1 DOWNT0 0);
        X0: IN    bit_vector (1 DOWNT0 0);
        S: IN     bit_vector (1 DOWNT0 0);
        e: IN     bit;
        y: OUT bit
  );
END mux_3_1_2bit;
```



CASE statement

```
ARCHITECTURE casestmt OF mux_3_1_2bit IS
BEGIN
  PROCESS (S, X0, X1, X2, e)
  BEGIN
    IF e = '1' THEN
      CASE S IS
        WHEN "00"      => y <= X0;
        WHEN "01"      => y <= X1;
        WHEN OTHERS    => y <= X2;
      END CASE;
    ELSE y <= '0';
    ENDIF;
  END PROCESS;
END ARCHITECTURE casestmt;
```



3:1 mux

e	s ₁ s ₀	y
0	- -	0
1	0 0	x0
1	0 1	x1
1	1 0	x2
1	1 1	??

Selected Signal Assignment

ARCHITECTURE ssa OF mux_3_1_2bit IS
BEGIN

SIGNAL t : bit_vector (1 DOWNTO 0);

WITH S SELECT

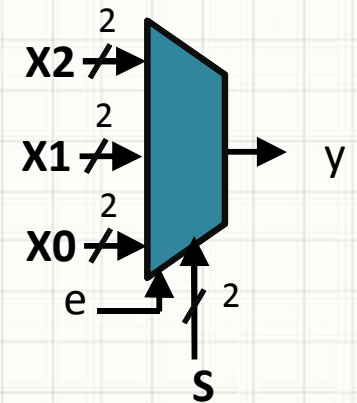
t <= X0 WHEN "00",

X1 WHEN "01",

X2 WHEN OTHERS;

y <= t AND e;

END ARCHITECTURE ssa;



3:1 mux

e	s ₁ s ₀	y
0	- -	0
1	0 0	x0
1	0 1	x1
1	1 0	x2
1	1 1	??



THANKS