

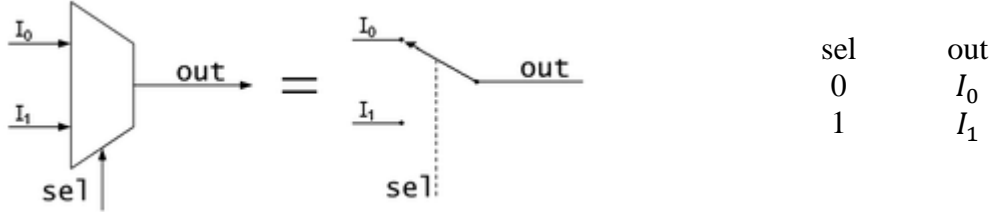
Deney4) MUX & ALU

Bu deneyde, Mux ve ALU VHDL dili ile yazılarak **ALTERA-DE0** kartı üzerinde çalıştırılacaktır.

	Family:	Name:
ALTERA-DE0	Cyclone III	EP3C16F484C6

3a) MUX

Mux yapısı, seçme anahtarına bağlı olarak girişleri çıkışa yönlendirmektedir. Bu deneyde 4bit veri yoluna sahip 2to1 mux yapısını tasarlanacaktır.



mux2to1_4bit.vhd

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

entity mux2to1_4bit is
    Port ( in0 : in  STD_LOGIC_VECTOR (3 downto 0);
          in1 : in  STD_LOGIC_VECTOR (3 downto 0);
          outG : out STD_LOGIC_VECTOR (3 downto 0);
          s : in  STD_LOGIC);
end mux2to1_4bit;

architecture Behavioral of mux2to1_4bit is

begin

process (in0,in1,s)
begin
    case s is
        when '0' =>
            outG <= in0;

        when '1' =>
            outG <= in1;

    when others =>
        outG <= "XXXX";
    end case;
end process;
end Behavioral;
```

3b) ALU

Bir işlemcinin temel yapılarından birisi aritmetik ve logic birimdir. Arithmetic ve logic birimlerin fonksiyon tablosu aşağıdaki gibidir.

Select		Input	$G = (A \mid Y \mid C_{in})$	
S_1	S_0	Y	$C_{in} = 0$	$C_{in} = 1$
0	0	all 0s	$G = A$ (transfer)	$G = A + 1$ (increment)
0	1	B	$G = A + B$ (add)	$G = A + B + 1$
1	0	\overline{B}	$G = A + \overline{B}$	$G = A + \overline{B} + 1$ (subtract)
1	1	all 1s	$G = A - 1$ (decrement)	$G = A$ (transfer)

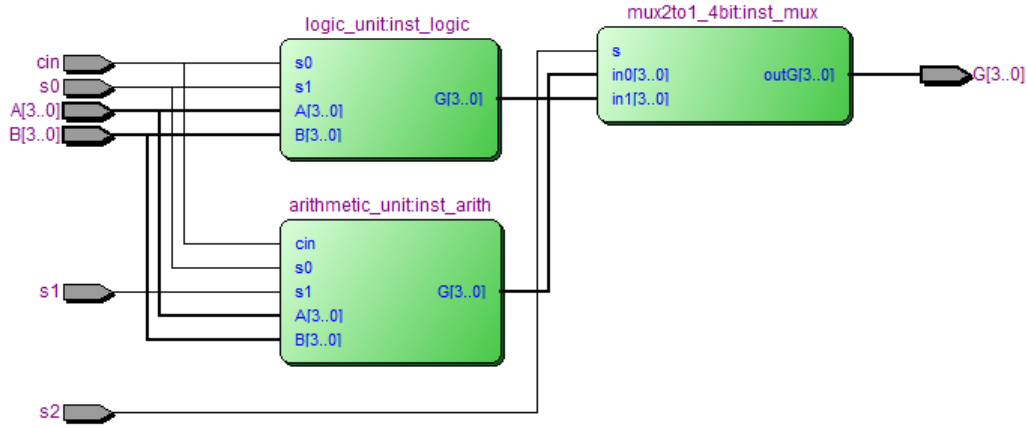
Şekil 1 arithmetic unit

S_1	S_0	Output	Operation
0	0	$G = A \wedge B$	AND
0	1	$G = A \vee B$	OR
1	0	$G = A \oplus B$	XOR
1	1	$G = \overline{A}$	NOT

Şekil 2 logic unit

arithmetic_unit.vhd	logic_unit.vhd
<pre> library IEEE; use IEEE.STD_LOGIC_1164.ALL; use IEEE.STD_LOGIC_ARITH.ALL; use IEEE.STD_LOGIC_UNSIGNED.ALL; entity arithmetic_unit is Port (A : in STD_LOGIC_VECTOR (3 downto 0); B : in STD_LOGIC_VECTOR (3 downto 0); G : out STD_LOGIC_VECTOR (3 downto 0); s0 : in STD_LOGIC; s1 : in STD_LOGIC; cin : in STD_LOGIC); end arithmetic_unit; architecture Behavioral of arithmetic_unit is signal sel: std_logic_vector(2 downto 0); begin sel<=s1 & s0 & cin; process (A,B,sel) begin case sel is when "000" => G <= A; when "001" => G <= A+1; when "010" => G <= A+B; when "011" => G <= A+B+1; when "100" => G <= A+not(B); when "101" => G <= A+not(B)+1; when "110" => G <= A-1; when "111" => G <= A; when others => G <= "XXXXX"; end case; end process; end Behavioral; </pre>	

Bu iki birim, bir mux ile birleştirilerek ALU elde edilir. ALU'nun bağlantı yapısı aşağıdaki gibidir.



ALU'nun fonksiyon tablosu aşağıdaki gibidir. Tasarladığımız sistemin aşağıdaki tabloyu sağladığını kontrol ediniz.

Function Table for ALU

Operation Select				Operation	Function
S_2	S_1	S_0	C_{in}		
0	0	0	0	$G = A$	Transfer A
0	0	0	1	$G = A + 1$	Increment A
0	0	1	0	$G = A + B$	Addition
0	0	1	1	$G = A + B + 1$	Add with carry input of 1
0	1	0	0	$G = A + \overline{B}$	A plus 1s complement of B
0	1	0	1	$G = A + \overline{B} + 1$	Subtraction
0	1	1	0	$G = A - 1$	Decrement A
0	1	1	1	$G = A$	Transfer A
1	X	0	0	$G = A \wedge B$	AND
1	X	0	1	$G = A \vee B$	OR
1	X	1	0	$G = A \oplus B$	XOR
1	X	1	1	$G = \overline{A}$	NOT (1s complement)

Dip not:

* Aynı proje içinde “arithmetic_unit.vhd”, “logic_unit.vhd”, “mux2to1_4bit.vhd” ve “ALU.vhd” isimli **dört ayrı kod sayfası** olduğuna dikkat ediniz!!!

*Hierarchy olarak “ALU.vhd” dosyasının **top level** olmasına dikkat ediniz!!!

*Tasarımı DE0 FPGA kartına yüklerken toplam 12 anahtar girişi gereklidir, kart üzerinde 10 adet “toggle switch” bulunmaktadır. Diğer iki anahtar girişi için “pushbutton” kullanabilirsiniz. Kullanım kolaylığı için s1 ve s2 girişlerini pushbutton'lara atamanız önerilir. (pushbutton terstir, yani basınca 0, basılı değilken 1 gönderir)

*Mux yapısı, IF, Case, With-Select, When yapılarından birisi ile tasarlanacaktır. Bu yapılar hakkında bilgiyi VHDL1 isimli sunumdan bulabilirsiniz.

*modelsim simülasyonunda binary gözüken sayıları decimal'e çevirmek için, seçilen giriş/çıkış'a sağ tık ile gelen seçeneklerden “radix” seçilip decimal, unsigned vb.. seçilebilir.

*Girişlerin 10 tabanında girebilirsiniz. (örn 5 girmek için; 10#5 şeklinde yazılabilir.)

