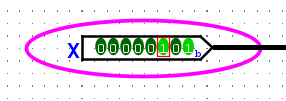
**TASK 1. Implementing a 2’s Complement Convertor**

In this task we used these steps to convert the 8 bit quantity to 2’s complement;

1. We select the input as binary form of the quantity.
   * For example my student number is 2587285.
   * The first bit of this number is 5.
   * 8 bit binary representation of 5 is ‘00000101’.



1. And then we swap each 0 with 1 and each 1 with 0.

* For the swap of each bit we use not gate with ‘Data Bits’ of 8.

metin, ekran görüntüsü, çizgi, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. In the next step we add 1 to this number as the last step of the method.
   * For adding 1 to the number we use 8 bit full adder with the constant with value of ‘0x1’.

ekran görüntüsü, metin, çizgi, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

* + Steps of the implementation of 8 bit full adder:
    1. 1 bit full adder:

diyagram, çizgi, plan, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Açıklama otomatik olarak oluşturuldu

* + 1. 2 bit full adder:

metin, ekran görüntüsü, diyagram, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

* + 1. 4 bit full adder:

metin, ekran görüntüsü, diyagram, çizgi içeren bir resim

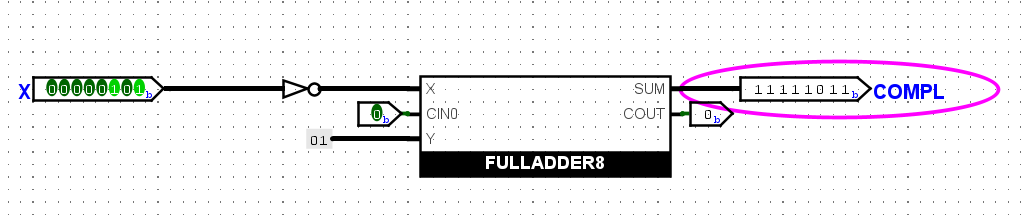
Açıklama otomatik olarak oluşturuldu

* + 1. 8 bit full adder:

diyagram, plan, metin, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. As the last part of the circuit we wired the adder to the output which is labeled as COMPL.
   * The output value gives **1111 1011** which is the 2’s complement of the value ‘5’.



1. Here the test vector file and the testing results for student number ‘2587285’:
   * Test Vector file:

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

* + Result:

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

**TASK 2. Implementing Two Sub-circuits: Rotate Left and Rotate Right**

For the Rotate Left Circuit these are the steps of implementation:

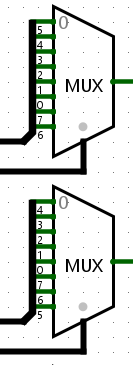
1. We have two inputs that are X (the value to be rotated) and Y (the rotation amount).
   * For Y we need 3 bits for the max value of 7 which is the maximum rotation amount for 8 bit quantity.

yazı tipi, ekran görüntüsü, çizgi, diyagram içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. Each multiplexer chooses one of several possible inputs (either directly from the original input or shifted positions).
2. There are 8 multiplexers (one for each bit) in the circuit, which allows the system to select between different shifted values based on the control input (Y).
   * The bit positions arranged for ‘swap’ operation.

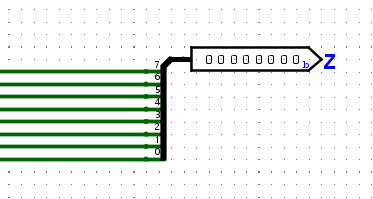
diyagram, çizgi, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu  diyagram, çizgi, ekran görüntüsü, metin içeren bir resim

Açıklama otomatik olarak oluşturuldu diyagram, çizgi, plan, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. After we get each bit from 8 MUX we again use a splitter to assemble them 8 bits.
   * Then we get output of shifted 8 bits.



1. Complete Circuit:

diyagram, taslak, çizgi, paralel içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. Here the test vector file and the testing results for student number ‘2587285’:
   * Test Vector File:

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

* + Result:

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

For the Rotate Right Circuit the steps of implementation are same except MUX and Output splitter bit arrangements:

1. Bit arrangement of 8 MUX for rotating right and swap operation:

diyagram, çizgi, metin, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu diyagram, metin, çizgi, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu diyagram, çizgi, plan, dikdörtgen içeren bir resim

Açıklama otomatik olarak oluşturuldu diyagram, çizgi, ekran görüntüsü, plan içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. The output splitter bit arrangement for rotating right and swap operation:

metin, ekran görüntüsü, çizgi, paralel içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. Complete Circuit:

diyagram, plan, çizgi, paralel içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. Here the test vector file and the testing results for student number ‘2587285’:
   * Test Vector File:

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

* + Result:

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

**TASK 3. Implementing an 8-bit Full Adder**

1. Three inputs: X, Y, and CIN, along with two outputs: SUM and COUT.
   * Inputs & Outputs:

metin, ekran görüntüsü, yazı tipi, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu ekran görüntüsü, yazı tipi, diyagram, metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. In the input stage we use 2 splitters to split each bit and wire them with 8 x 1 bit full adders:

metin, diyagram, çizgi, ekran görüntüsü içeren bir resim

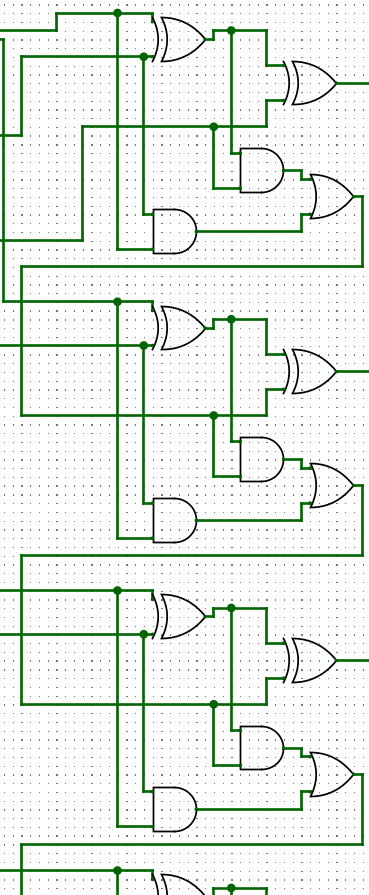
Açıklama otomatik olarak oluşturuldu

1. One of the eight 1 bit adder circuit:
   * Design:

diyagram, plan, çizgi, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

* + Each of the Carry Out wires wired to the next Carry In of next adder:



1. Lastly we have 2 outputs:
   * The Carry Output comes from 8th OR Gate 8 x 1 bit adder.
   * For the Sum Output we use splitter to merge the 8 bit which comes from 8 x 1 bit adders:

ekran görüntüsü, yazı tipi, metin, diyagram içeren bir resim

Açıklama otomatik olarak oluşturuldu

Note: In the above section of Task 1 we have simplified design of the 8 bit full adder. In this section for clarification we showed ever 1 bit full adder. You may see the every bit arrangements which splitted by splitters.

1. Complete Circuit:

metin, diyagram, paralel, plan içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. Here the test vector file and the testing results for student number ‘2587285’:
   * Test Vector File:

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

* + Result:

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

**TASK 4. Designing a Custom Comparator**

1. We have two inputs that are X and Y both 8 bits.
   * The design will compare these two inputs.

yazı tipi, çizgi, ekran görüntüsü, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu yazı tipi, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

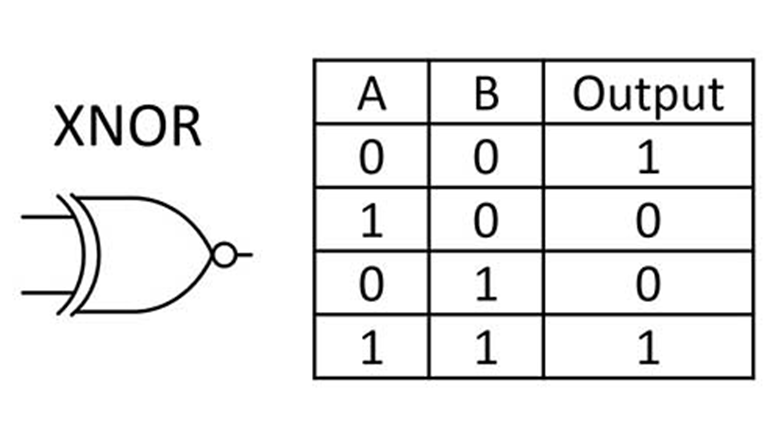
1. We used splitters to split each bit and compare them seperately.
   * The logic is comparing 2 bits which are the seperate bits of X and Y inputs.
   * For example, we will compare the 1st bit of X and the 1st bit of Y then, 2nd bit of X and 2nd bit of Y etc.

ekran görüntüsü, metin, çizgi, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu ekran görüntüsü, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. For the comparison we used XNOR Gates which gives 1 for both same input and 0 for different inputs.
   * XNOR Truth Table:



* + Design of Gates:

diyagram, çizgi, plan, teknik çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. For the output of the 8 XNOR Gates we used one 8 bit AND Gate to see if these all 8 comparison is 1 or 0.
   * If all the comparison results are 1 from XNOR Gates then the 8 bit is the same so all inputs of AND Gate are 1 which means (1) EQUAL.
   * If all the comparison results are not 1 from XNOR Gates then the 8 bit is not the same so all inputs of AND Gate are not 1 which means (0) NOT EQUAL.

diyagram, çizgi, plan, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. The output of AND Gate is shows if these two 8 bit inputs are equal (1) or not (0).
   * Output labeled as EQUAL.

diyagram, yazı tipi, simge, sembol, grafik içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. Complete Circuit:

diyagram, plan, çizgi, şematik içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. Here the test vector file and the testing results for student number ‘2587285’:
   * Test Vector File:

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

* + Result:

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

**TASK 3. BUILDING AN ALU**

1. In the design of ALU we have two 8 bit inputs X and Y.
   * Addition to this we have one 2 bit input to select which function to perform called opcode.

ekran görüntüsü, yazı tipi, çizgi, metin içeren bir resim

Açıklama otomatik olarak oluşturuldu simge, sembol, diyagram, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. Also we have one 8 bit output and five 1-bit outputs (condition code flag).
   * One 8 bit output labeled as RESULT.
   * Five 1 bit flags labeled as:
     1. EQ, OVF, CF, ZF, SF.

metin, ekran görüntüsü, yazı tipi, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. This ALU will perform depends on the opcode as:

metin, çizgi, ekran görüntüsü, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. For first operation of ‘ADD’ we have 8 bit Full Adder:
   * To negative binary number summation we have 2’s Complement design.

metin, ekran görüntüsü, çizgi, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, diyagram, çizgi, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. For second operation of ‘OR’ and ‘AND’ we have 8 bit OR and AND Gates:

diyagram, çizgi, taslak, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. For the 4th and the last operation we have 8 bit Right Rotator:

çizgi, ekran görüntüsü, yazı tipi, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. After the operation set we have Custom Comparator for EQ Flag:

metin, ekran görüntüsü, çizgi, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. For the CF flag we have the Cout output of the 8 bit Full Adder:

diyagram, çizgi, ekran görüntüsü, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. For the ZF flag we have 8 bit OR Gate and after NOT Gate which takes seperately 8 bits of summation result.
   * With this design it sets to 1 if the results equal to zero.

diyagram, çizgi, yazı tipi, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. For the SF Flag we are using the 8th bit of the summation result.
   * With this design SF flag is set to 1 if the result is negative.

çizgi, ekran görüntüsü, burmak, çekmek, ingiliz anahtarı içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. For the OVF Flag we have design of AND, OR and NOT gates which sets flag to 1 when requested operation results in signed overflow.

diyagram, plan, çizgi, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. Complete Ciurcuit of ALU:

diyagram, plan, teknik çizim, şematik içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. Here the test vector file and the testing results for student number ‘2587285’ and ‘2587293’:
   * Test Vector File:

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

* + Result:

metin, yazı tipi, ekran görüntüsü, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu