*NAME: Kundrapu N R Sai Akash*

*ROLL NO: 20EC10043*

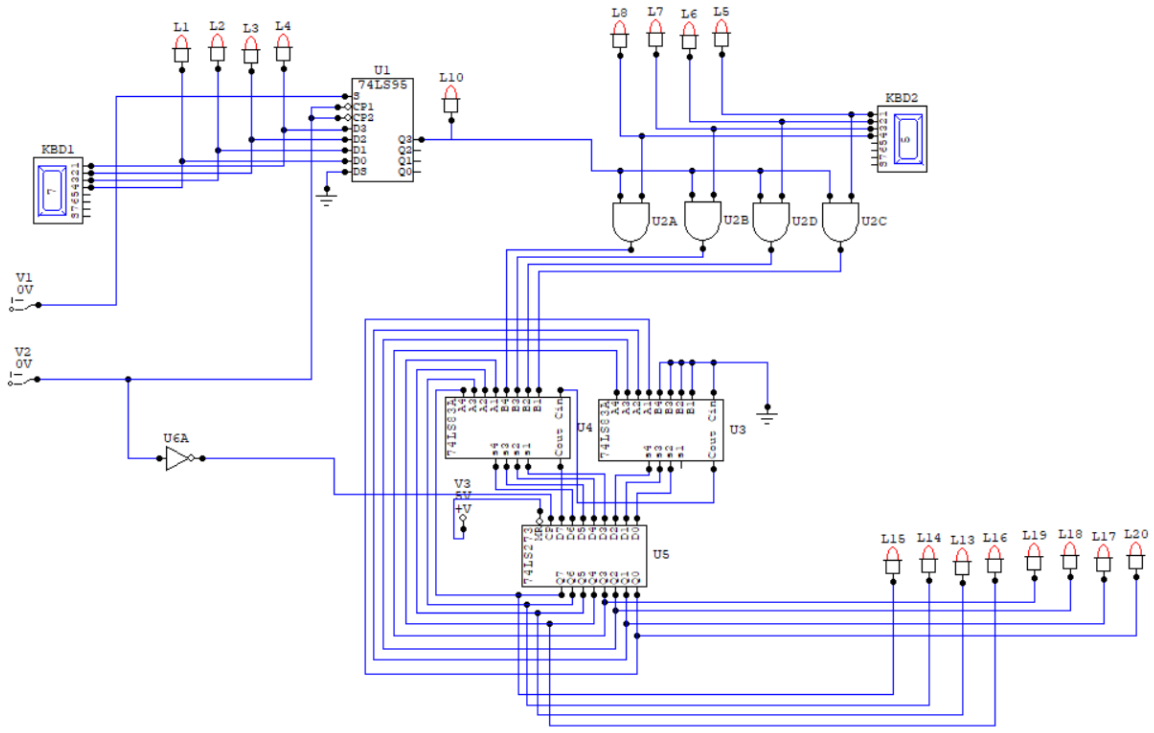
# *DIGITAL LABORATORY EXPERIMENT 9*

## AIM:

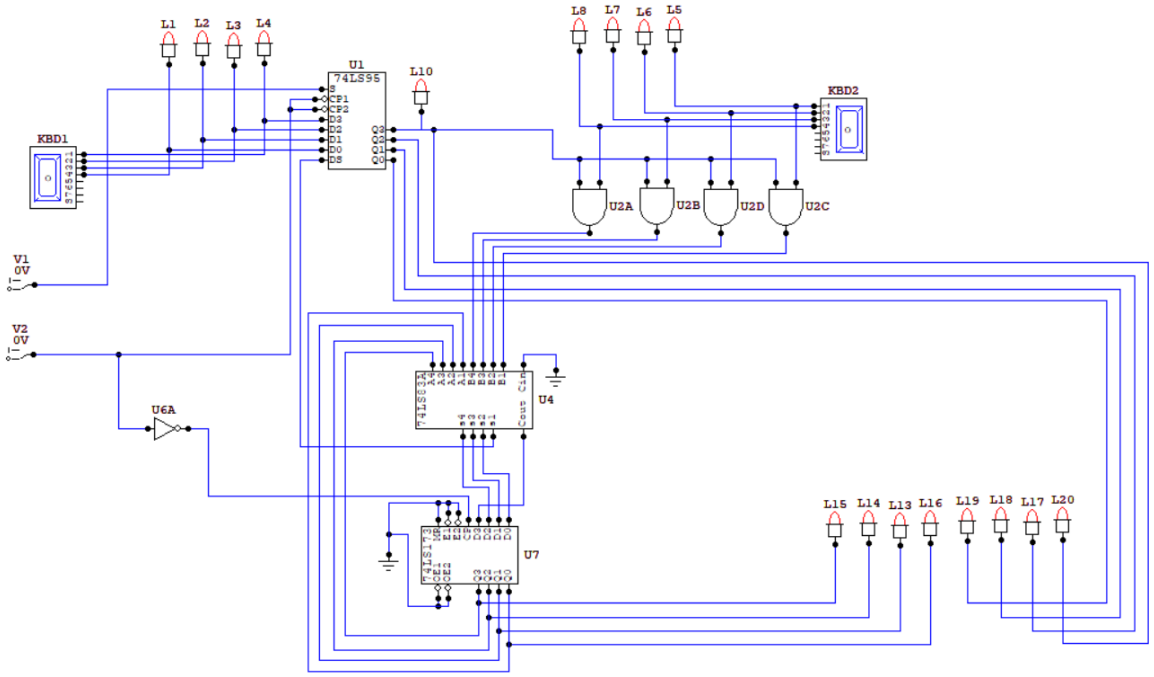
*Implementing 4x4-bit serial-parallel multiplier.*

## CIRCUIT DIAGRAM

*The required materials for this experiment are, ASCII keyboards, IC 7495(4-bit shift register), IC 7483(4-bit adders), IC 74273(8-bit register), IC 7408(2-In AND gates) and some logic displays.*



*When we want to optimize the circuit, which is to reduce the circuitry involved in the experiment, we use a 4-bit register along with the shift register instead of 8-bit register and a single 4-bit full adder instead of 8-bit adder. Other than that, all others are going to remain the same.*




## BREIF THEORY AND EXPLAINATION

*In this experiment, before going to the designing of the circuit, lets look at what we are trying to do. We want to create a 4x4 multiplier with the help of register and a 8-bit adder. For this, we use the concept of partial products.*

*a3 a2 a1 a0*

*b3 b2 b1 b0*

*---------------*

*Final Product = 8PP3 +4PP2 + 2PP1 + PP0, where as*

*PPi = a3a2a1a0 * bi*

*So, when we multiply the multiplicand with the multiplier, we get PP0 + PP1(x2) + PP2(x4) + PP3(x8). Here, we multiply x2 for every next partial product because, when we do that, we can shift the number to left by adding a 0 at the end [1101(13) → 11010(26)], which is essentially what we do in normal multiplication and then add all the partial products to get the correct answer.*

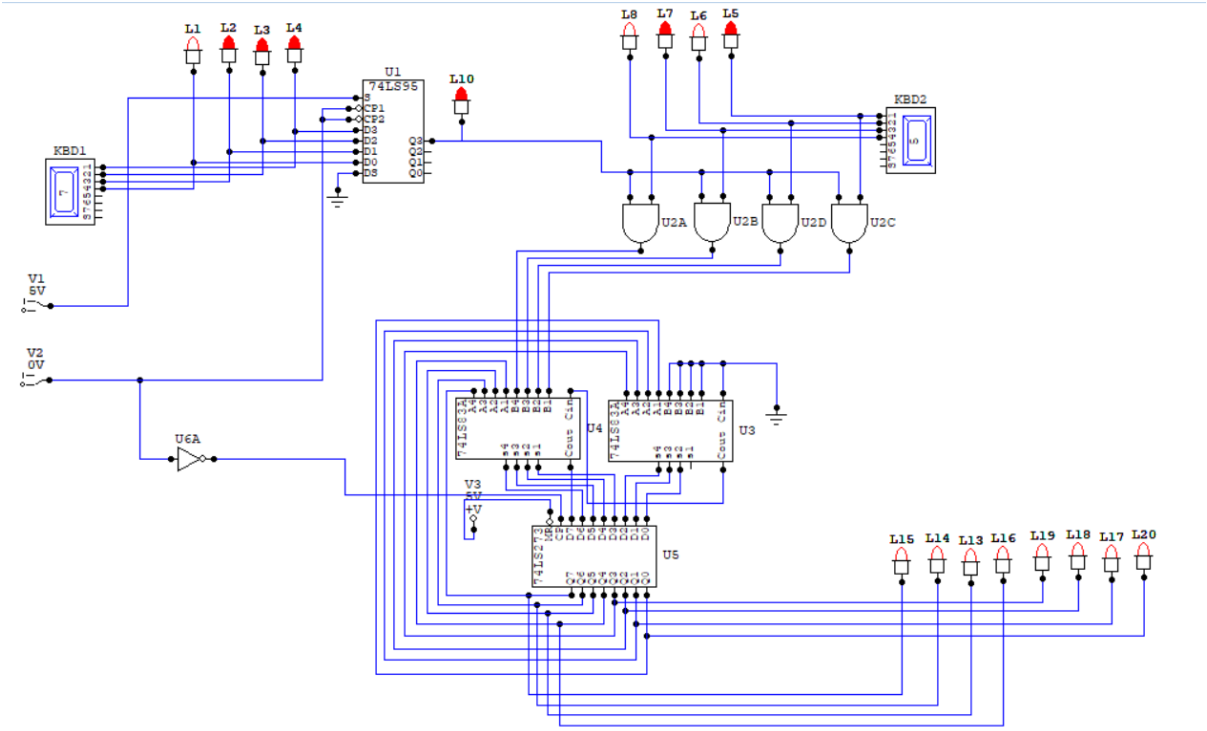
*In order to get the expression like that, we use the 8-bit adder to do a different job which is to get the average of both given inputs A and B, Input B of the adder is initially 0 and then it is the result we get in the subsequent additions, A has the lower nibble to be all 0000, so that the number which is the upper nibble is multiplied by 16, which upon first addition goes to 8PP0, then the number is adder by 16PP1 and divided by 2, which results in 8PP1 + 4PP0,... and so on, we get the expected answer of 8PP3 + 4PP2 + 2PP1 + PP0.*

*The circuit implementation of the first part is very much straight forward, where we use the 4-bit shift register, to shift the values of bi, to get the respective PPi for the addition purpose. We use the right shift property of the register to get b0 first and then b1, b2 and so on b3.*

*For the second part however, we want to reduce the cost of the experiment, so we use a 4-bit register instead of 8-bit register and a single 4-bit adder in place of a 8-bit register. At the adder part, we do 4bit mean of A and B inputs of adder, which results in PP3/2 + PP2/4 + PP1/8 + PP0/16, and we give these are the upper nibble of our result. As for the lower nibble, we use the right shift register where we give the remainder of each iterative division as the serial input, which slowly gets fed as the Q3Q2Q1Q0 as the 4clock pulses go, which results in our lower nibble.*

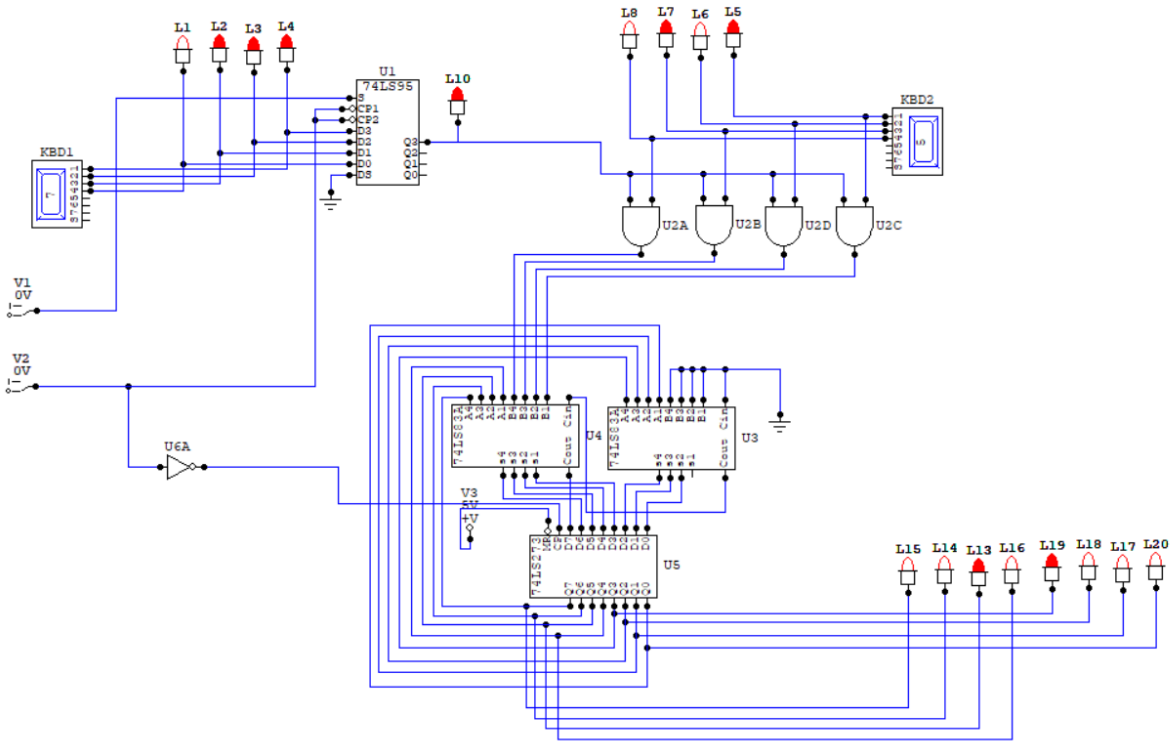
## SNAPSHOTS FROM THE EXPERIMENT

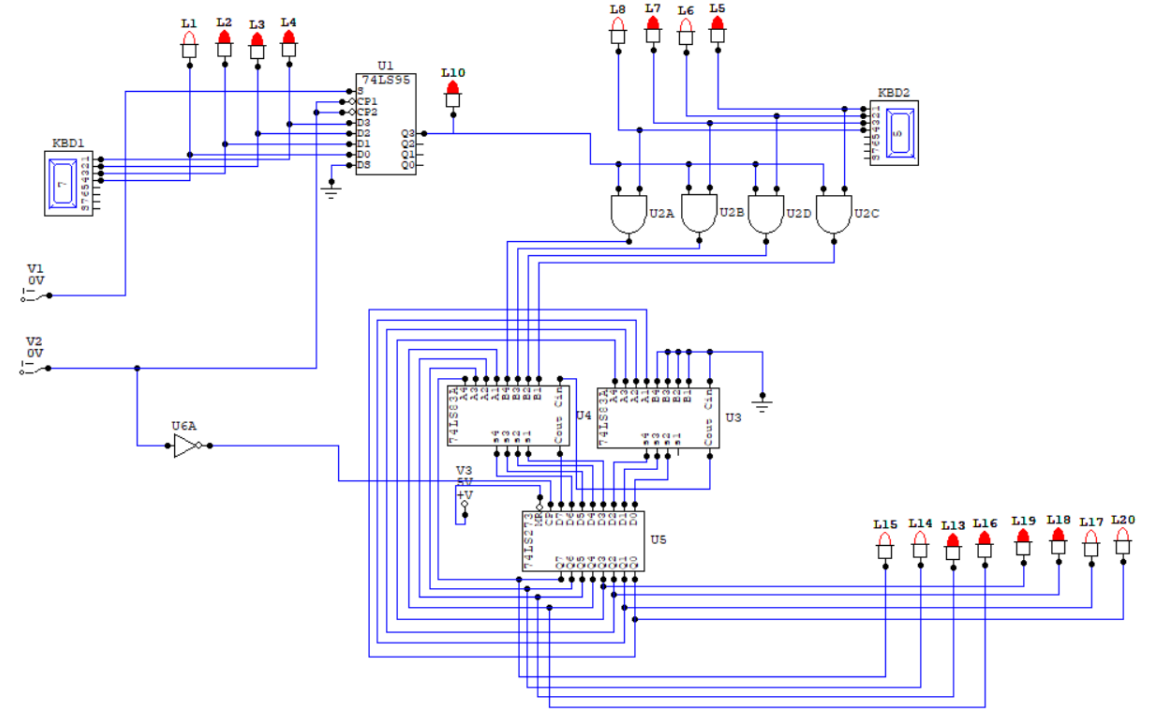
*For the first part of the experiment, we are taking the example of 7*5=35.*

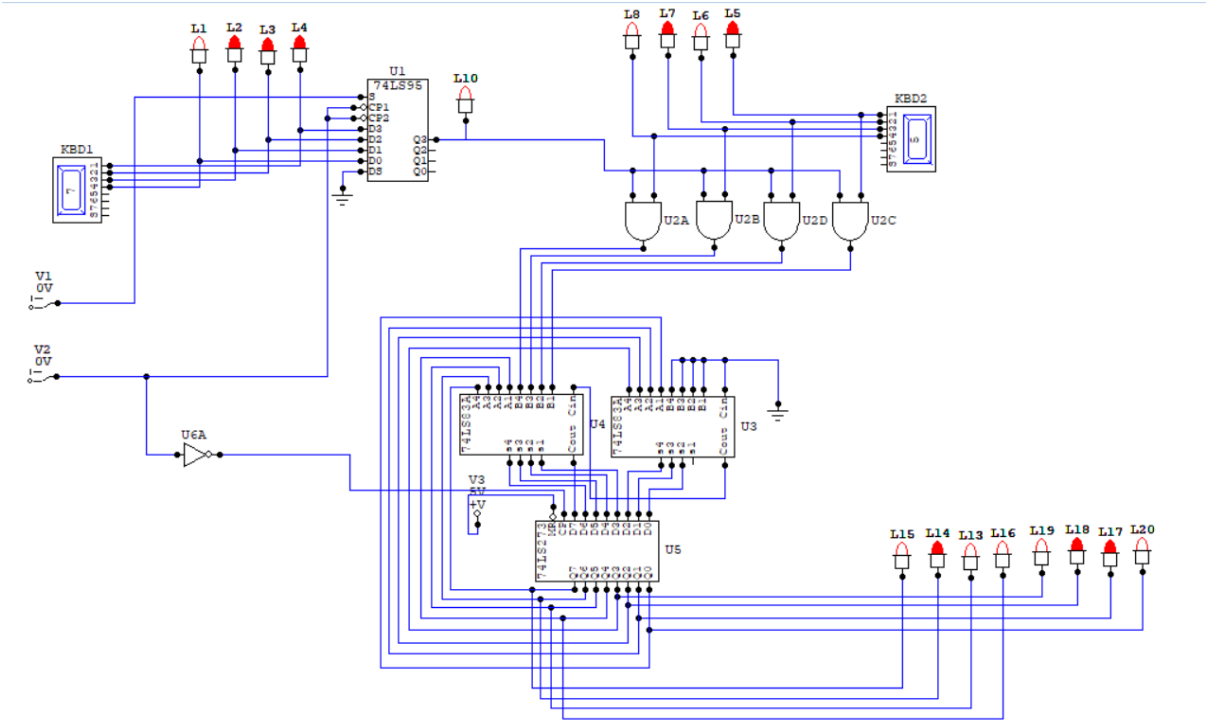


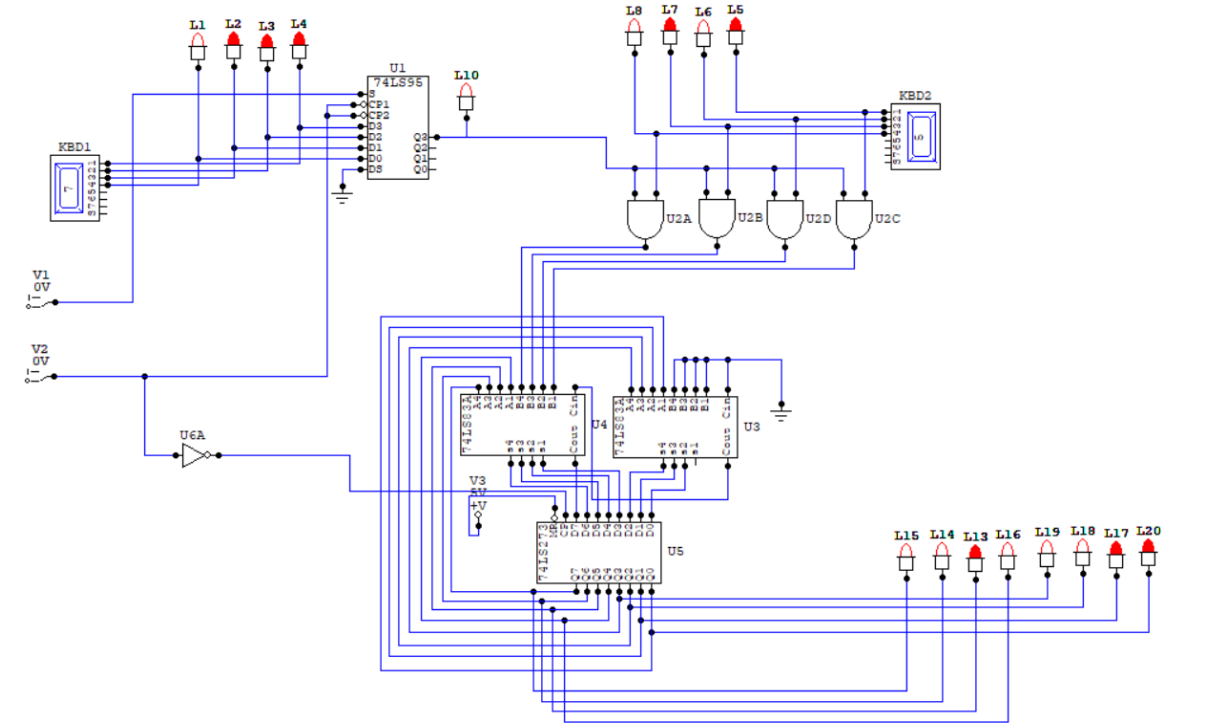

*S=1, and a Clock pulse to CP loads the parallel input to the output. (Parallel Loading)*

*Then, we set S=0 which sets the register to right shift mode and send 4clock pulses to do 4iterations to get the desired result.*

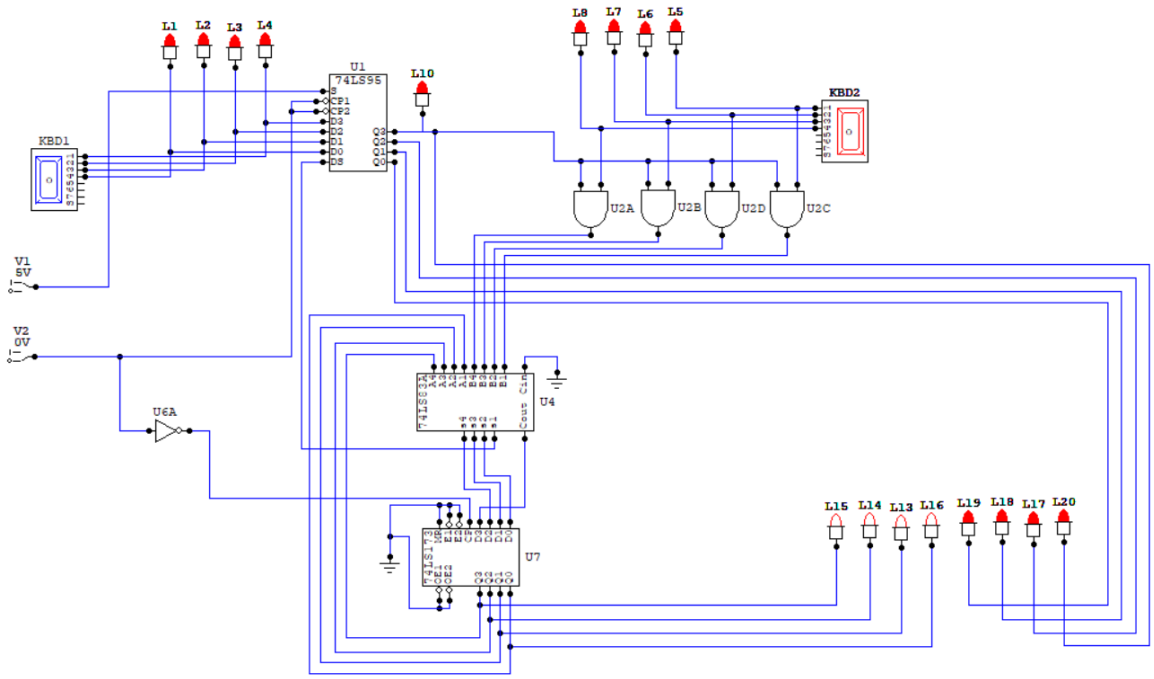

L1
L2
L3
L4
L8
L7
L6
L5
U1
74LS95
L10
KBD1
KBD2
U2A
U2B
U2D
U2C
V1
0V
V2
0V
U6A
U4
U3
V3
5V
U5
L15
L14
L13
L16
L19
L18
L17
L20


L1
L2
L3
L4
L8
L7
L6
L5
U1
74LS95
L10
KBD1
KBD2
U2A
U2B
U2D
U2C
V1
0V
V2
0V
U6A
U4
U3
V3
5V
U5
L15
L14
L13
L16
L19
L18
L17
L20


L1
L2
L3
L4
L8
L7
L6
L5
U1
74LS95
L10
KBD1
KBD2
U2A
U2B
U2D
U2C
V1
0V
V2
0V
U6A
U4
U3
V3
5V
U5
L15
L14
L13
L16
L19
L18
L17
L20

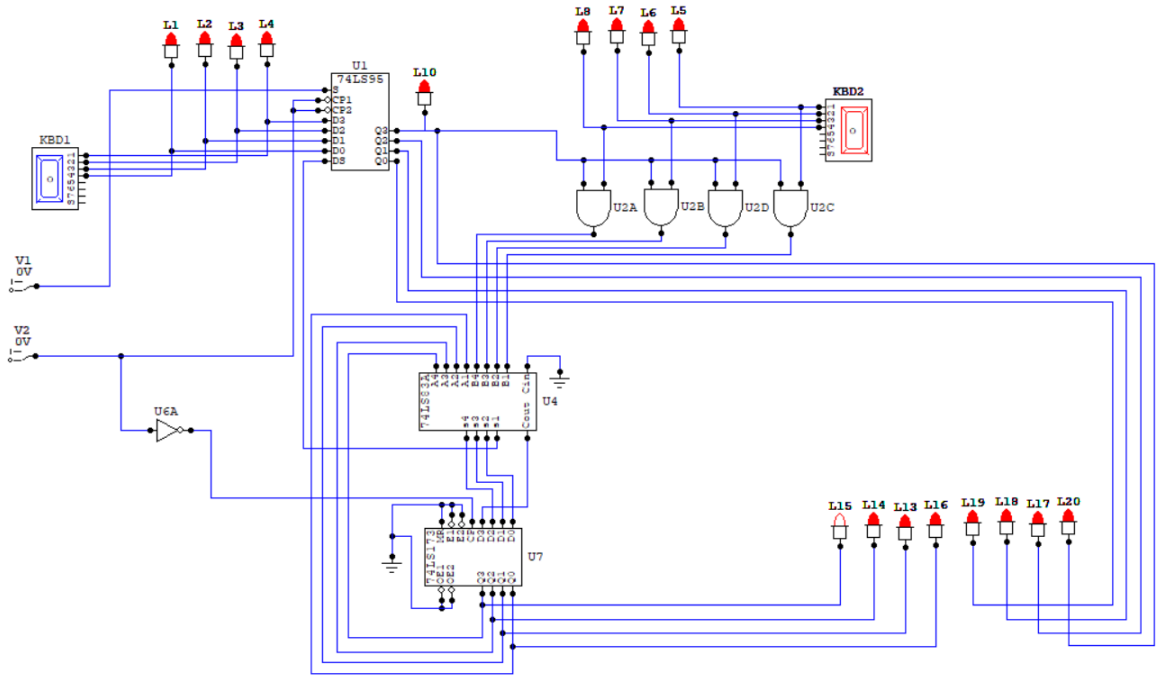

L1
L2
L3
L4
L8
L7
L6
L5
U1
74LS95
L10
KBD1
KBD2
U2A
U2B
U2D
U2C
V1
0V
V2
0V
U6A
U4
U3
V3
5V
U5
L15
L14
L13
L16
L19
L18
L17
L20

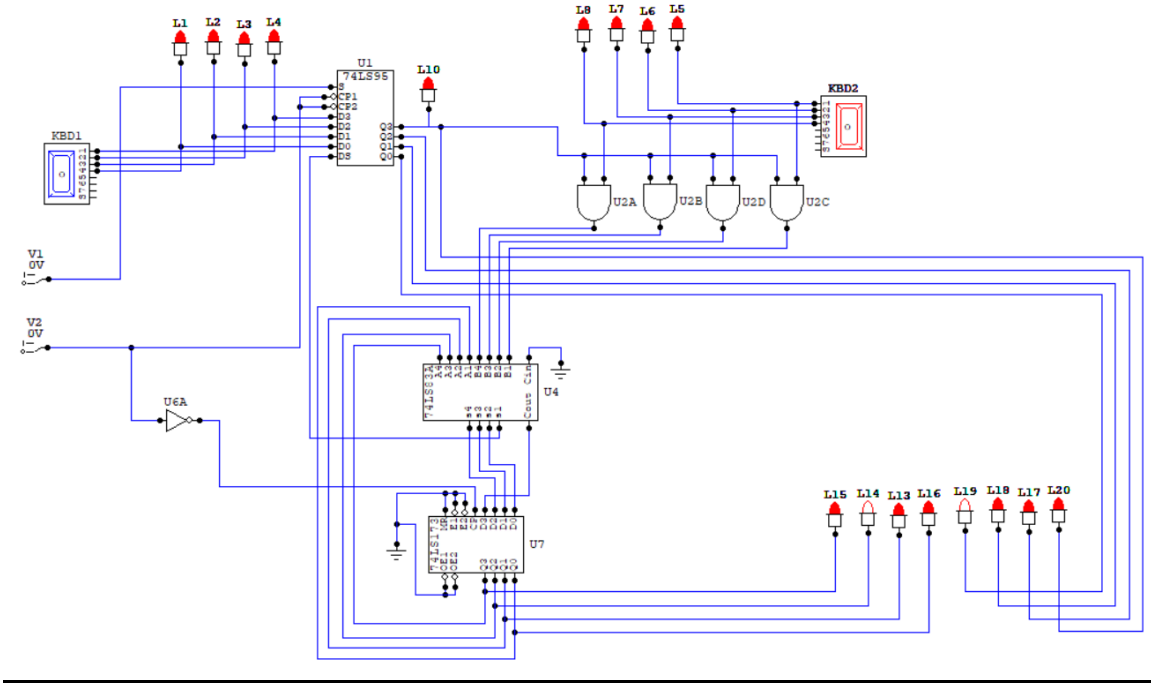
*For the second part of the experiment, we are taking the example of 15*15=225.*

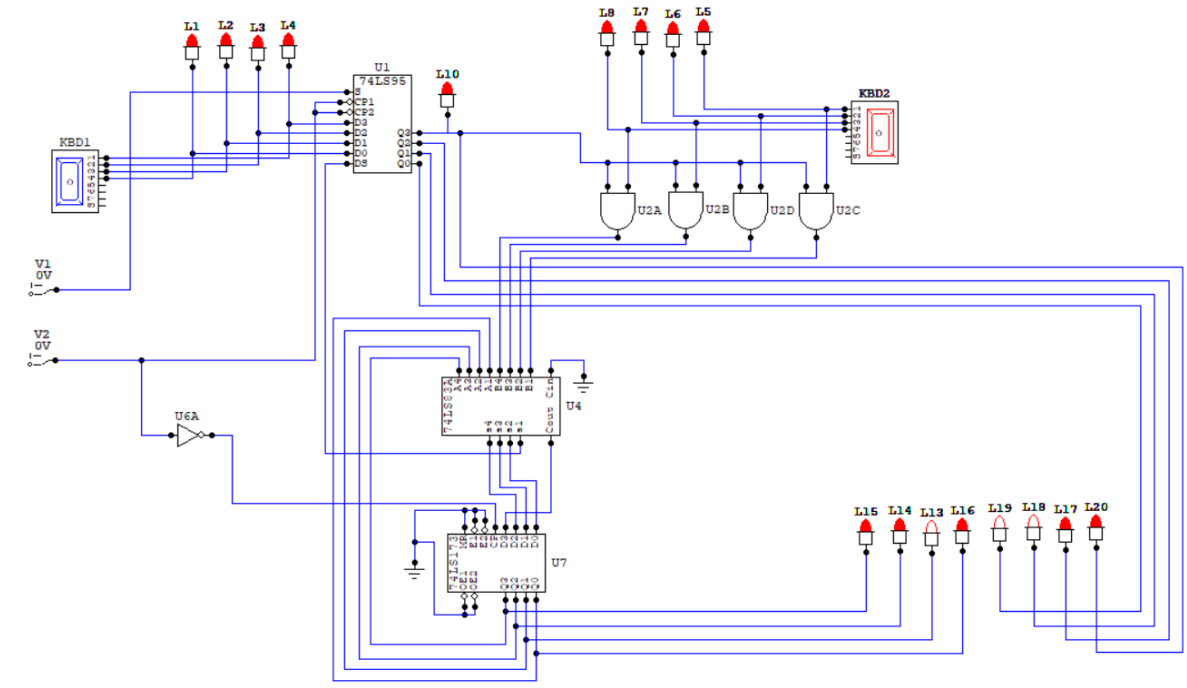


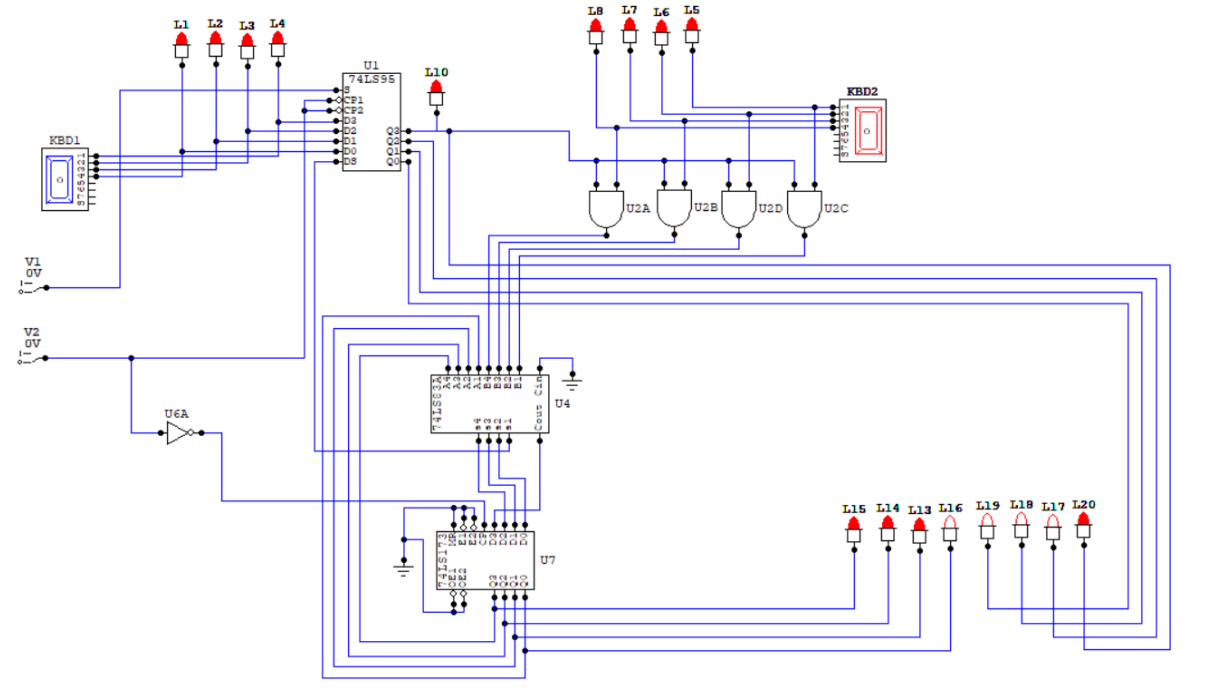

*S=1, and a Clock pulse to CP loads the parallel input to the output. (Parallel Loading)*

*Then, we set S=0 which sets the register to right shift mode and send 4clock pulses to do 4iterations to get the desired result.*


L1
L2
L3
L4
L8
L7
L6
L5
U1
74LS95
L10
KBD1
KBD2
U2A
U2B
U2D
U2C
V1
0V
V2
0V
U6A
U4
U7
L15
L14
L13
L16
L19
L18
L17
L20


L1
L2
L3
L4
L8
L7
L6
L5
U1
74LS95
L10
KBD1
KBD2
U2A
U2B
U2D
U2C
V1
0V
V2
0V
U6A
U4
U7
L15
L14
L13
L16
L19
L18
L17
L20


L1
L2
L3
L4
U1
74LS95
L10
KBD1
L8
L7
L6
L5
KBD2
U2A
U2B
U2D
U2C
V1
0V
V2
0V
U6A
U4
U7
L15
L14
L13
L16
L19
L18
L17
L20


L1
L2
L3
L4
U1
74LS95
L10
KBD1
L8
L7
L6
L5
KBD2
U2A
U2B
U2D
U2C
V1
0V
V2
0V
U6A
U4
U7
L15
L14
L13
L16
L19
L18
L17
L20

## OBSERVATION AND CONCLUSION

*As we see, when we check the values corresponding to the given inputs, they match accordingly, and when we try to reduce the cost by removing few circuit elements, we combine the shift register's serial input and right shift to generate the lower nibble and 4-bit adder's mean calculation as the upper nibble of our desired product.*

*For the purpose of checking, I will attach both the .ckt files along with this experiment. This takes us to the end of this experiment.*