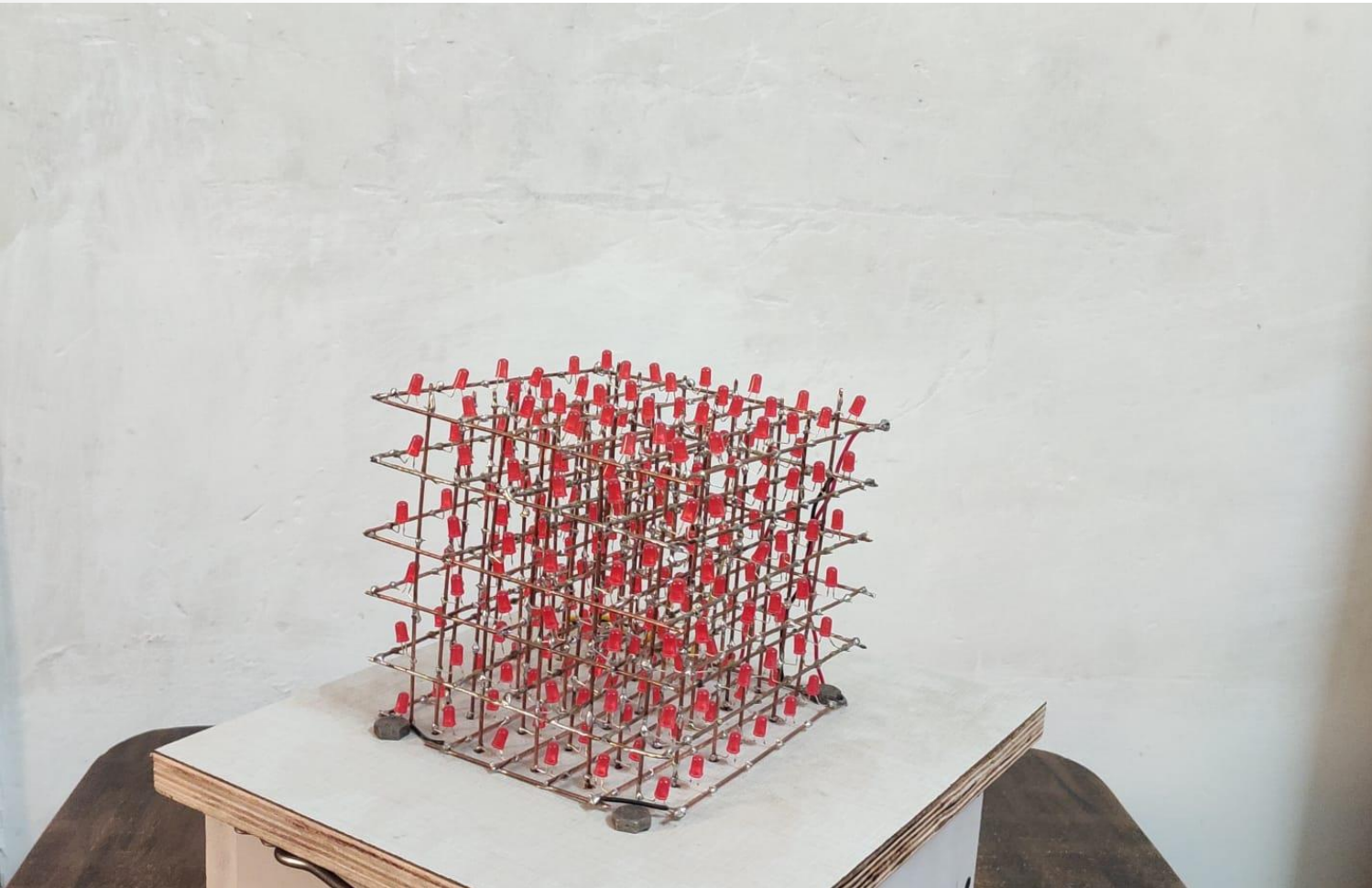# 6X6X6 ARDUINO BASED LED CUBE

DIKSHANT

21116038

ECE

THE BRAIN OF THIS CUBE IS **ARDUINO UNO.**

Arduino Uno is a microcontroller board which is based on the microcontroller ATmega328P. Basically this acts as an interface between the control circuit and the PC. It stores all the programs and generates the desired clock signals which is communicated with a shift register 74HC595, the heart of the system,using SPI protocol.
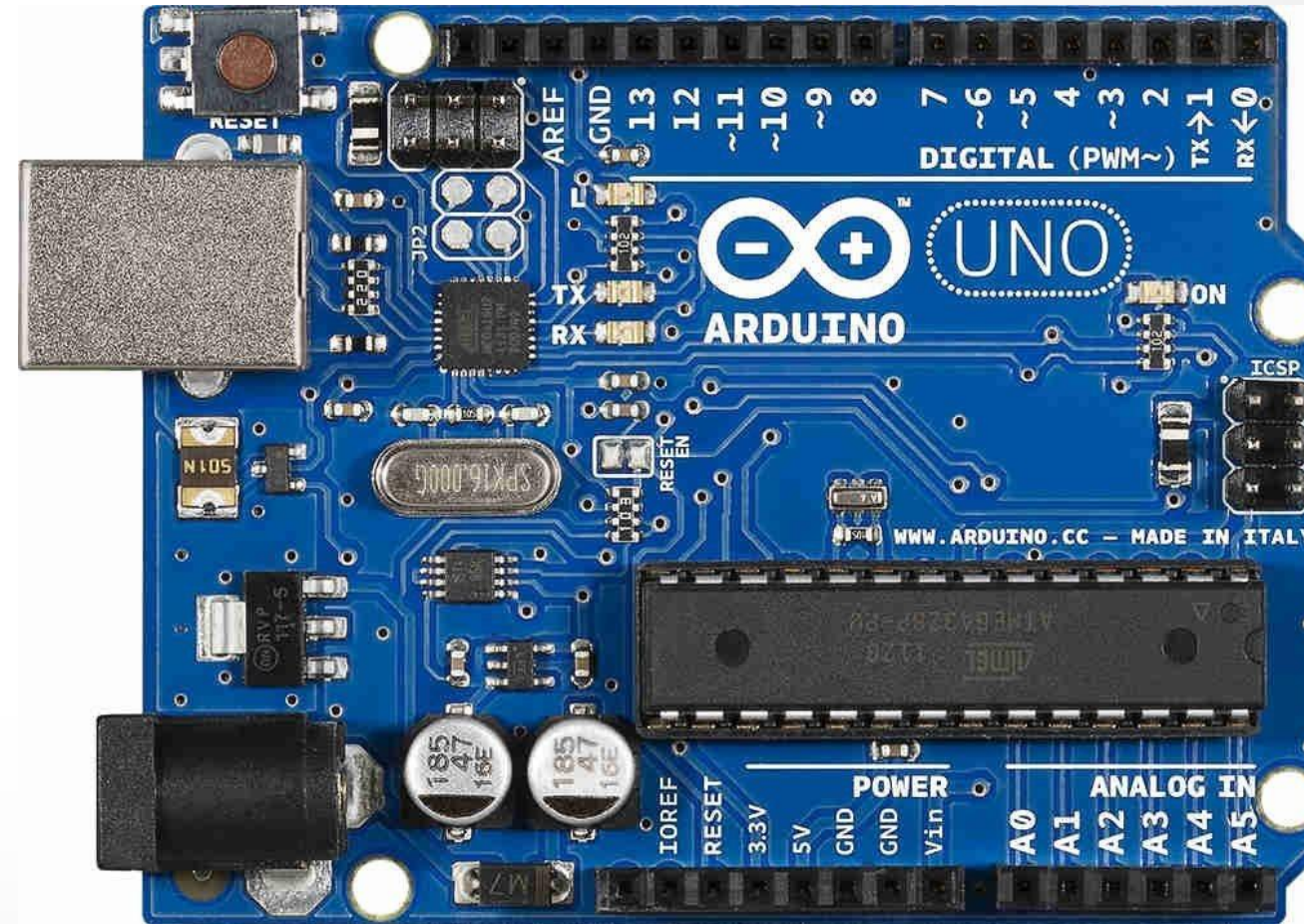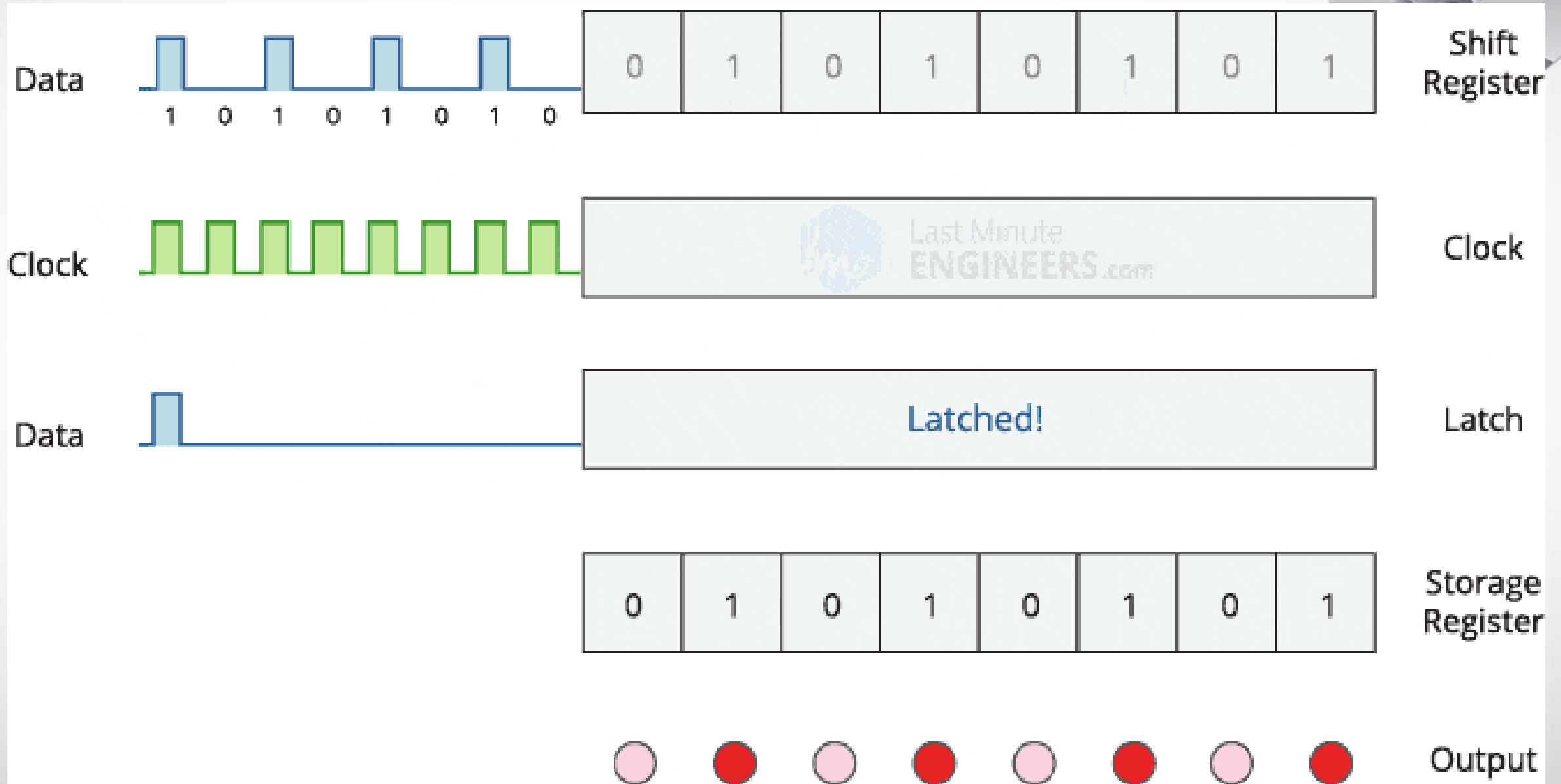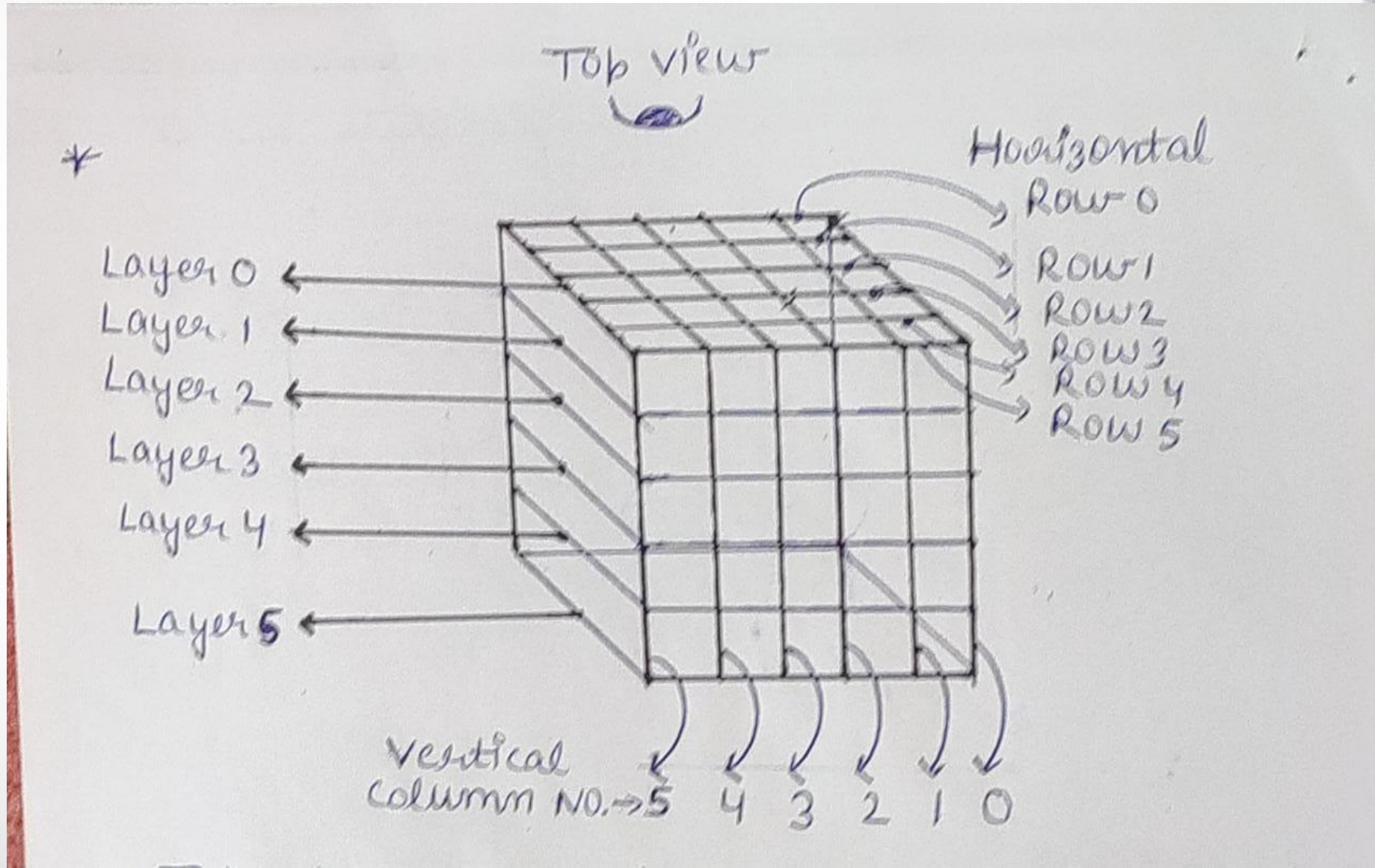
74HC595
THE HEART OF THE SYSTEM
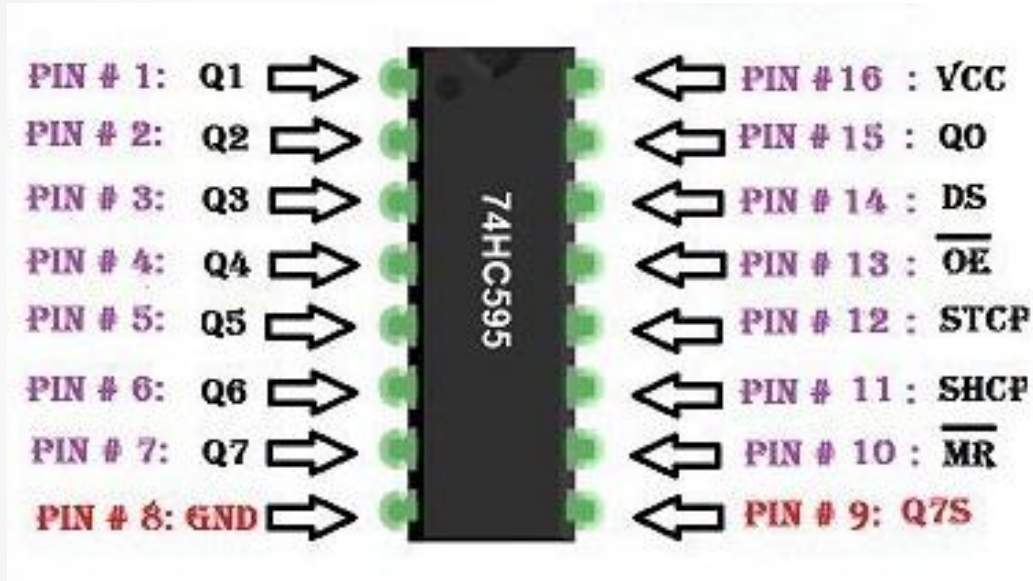
The 74HC595 is an 8-bit serial in parallel out shift register that contains a shift and a storage register. It contains a total of 8 outputs which can be easily expanded using the daisy chaining method.

PIN # 1: Q1
PIN # 2: Q2
PIN # 3: Q3
PIN # 4: Q4
PIN # 5: Q5
PIN # 6: Q6
PIN # 7: Q7
PIN # 8: GND

PIN # 16 : VCC
PIN # 15 : Q0
PIN # 14 : DS
PIN # 13 : OE
PIN # 12 : STCP
PIN # 11 : SHCP
PIN # 10 : MR
PIN # 9: Q7S

74HC595

74HC595
L8E6D4
UnG0906D

The arduino generates the required clock, latch and data signal for this IC and the IC outputs the data according to these clock signals.This can be better understood in the following animation.

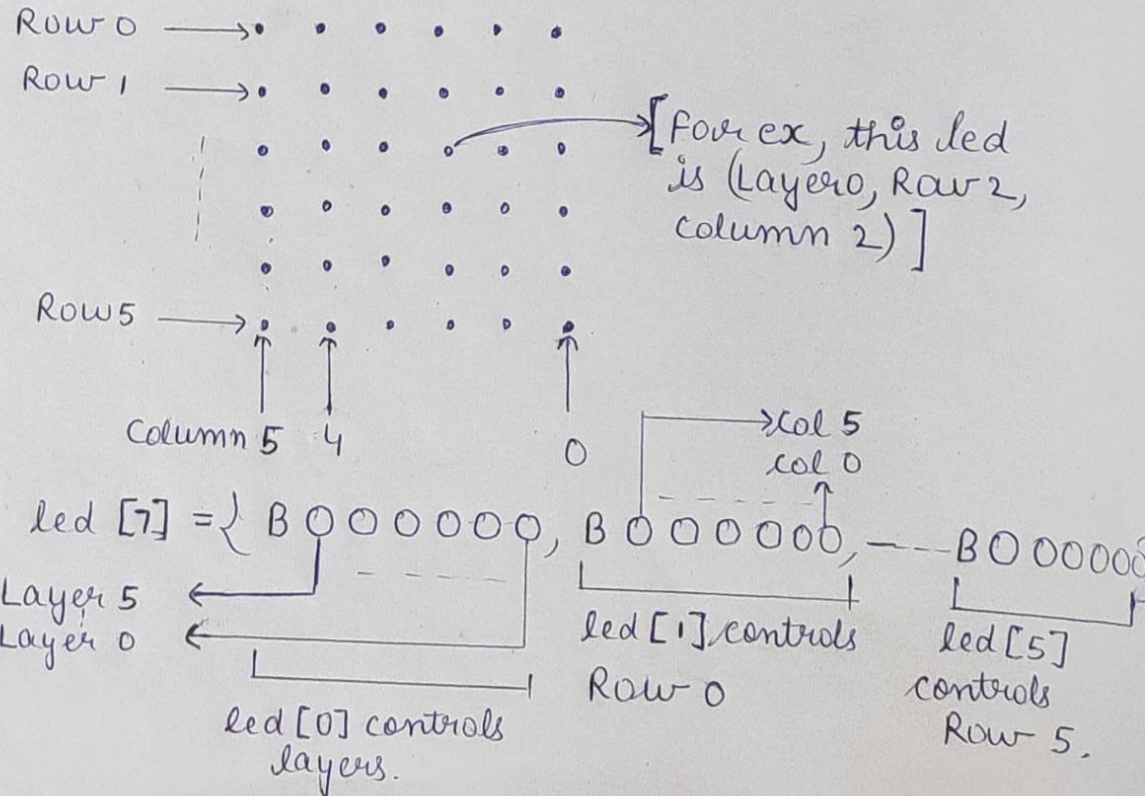# See how clock controls inflow of data and latch controls the outflow of data.

USING MULTIPLEXING

* Top view on 2D (see Layer 0 from top)

Row 0 ⟶ • • • • • •
Row 1 ⟶ • • • • • •
• • • • • • ⟶ [For ex, this led is (Layer 0, Row 2, column 2)]
• • • • • •
• • • • • •
Row 5 ⟶ • • • • • •

Column 5 4 ⟶ 0 ⟶ col 5 col 0

byte led [7] = { B 0 0 0 0 0 0, B 0 0 0 0 0 0, --- B 0 0 0 0 0 0 }

Layer 5 ←
Layer 0 ←

led [0] controls layers.

led [1] controls Row 0

led [5] controls Row 5.

(LSB first is used which gives that last output of a shift register is. controlled by LSB of a 6 bit data)

Daisy chained

Serial Data Input from Arduino → Shift Register 1 → Shift Register 2 → Shift Register 3 --------- Shift Register 7

Last 6 outputs controls columns of Row 5
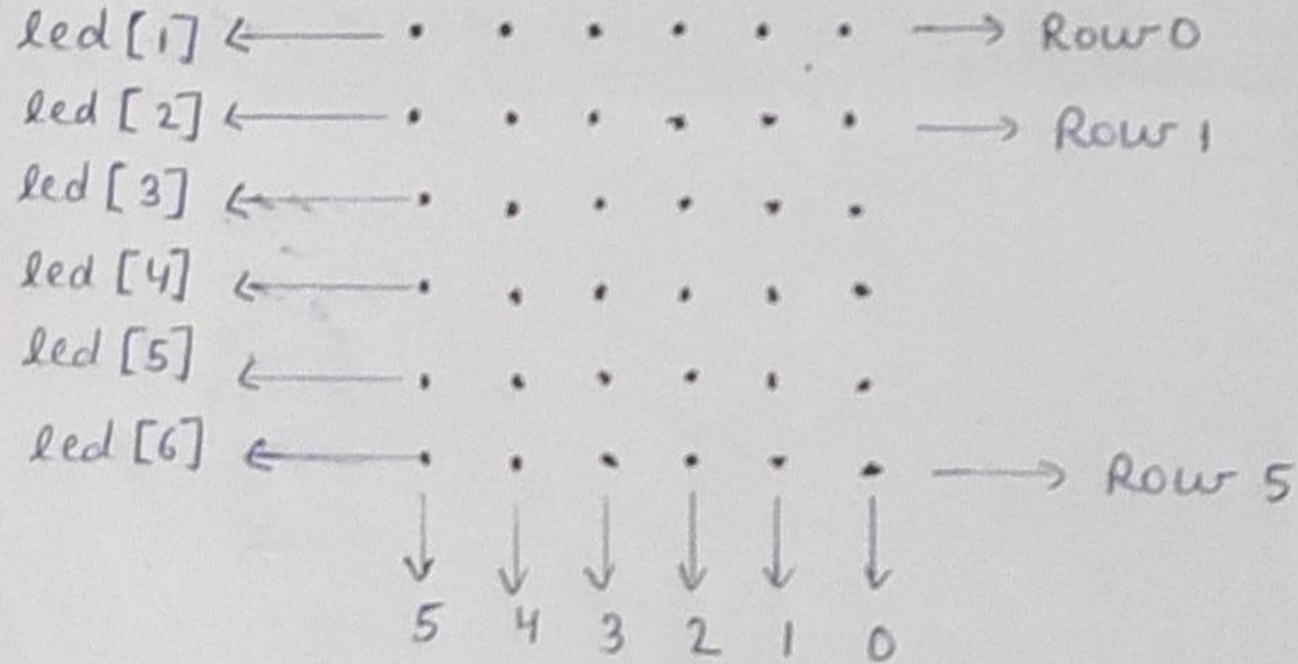
Last 6 outputs controls columns of Row 3.

Last 6 outputs controls layers.
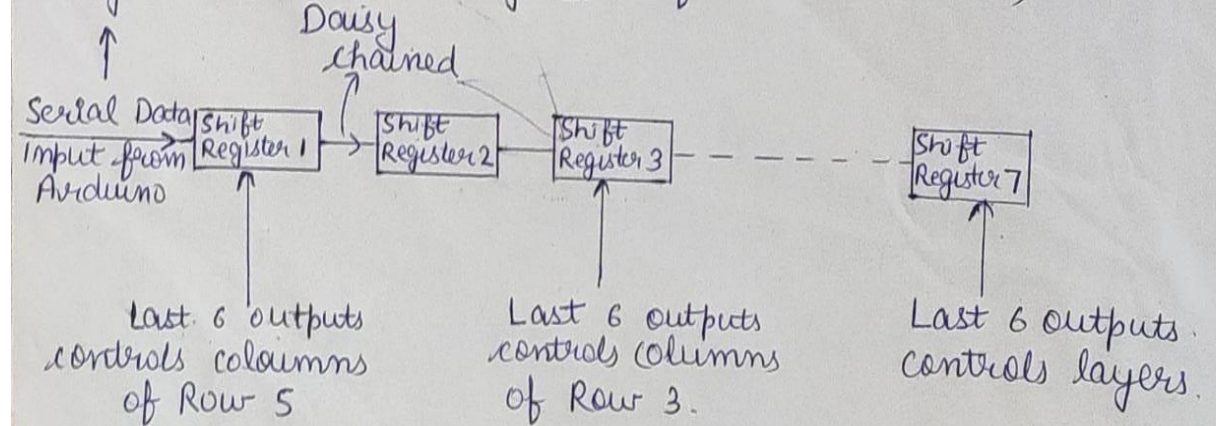
In the array led [7] = { B000000, B000000, --- B000000}, each entry of led array controlled a shift register thereby allowing to choose a row in the cube. While the each bit in an entry [LSB in B000000 controls column 0] controls the column of that row. The entry led [0] controls layers of cube [LSB controls layer 0]. So, led [0] controls layers, and led [1] to led [6] controls rows.

THE MULTIPLEXING TECHNIQUE USED HERE ALLOWS US TO CONTROL THE 216 LED'S USING ONLY AN ARRAY LED[] OF JUST 7 ELEMENTS.

led[] = {B000000, B000000, B000000, B000000, B000000, B000000, B000000}

Im, B0000000 , the individual bit controls coloumn.
↑↑↑↑↑↑↑
5 4 3 2 1 0th bit

led [1] ←———— ·   ·   ·   ·   ·   ·   ——→ Row 0
led [2] ←———— ·   ·   ·   ·   ·   ·   ——→ Row 1
led [3] ←———— ·   ·   ·   ·   ·   ·
led [4] ←———— ·   ·   ·   ·   ·   ·
led [5] ←———— ·   ·   ·   ·   ·   ·
led [6] ←———— ·   ·   ·   ·   ·   ·   ——→ Row 5

↓   ↓   ↓   ↓   ↓   ↓
5   4   3   2   1   0

- led [0] controls layers for 0th bit of led [0] controlling top layer.

So for example, the LSB of the element led[1] controls the rightmost coloumn(Col 0) of the backmost layer(Row 0). To select the LED from the 6 LED, 1 in each floor/layer, bits of element led[0] are controlled where LSB controls topmost floor (Layer 0) and MSB controls bottommost floor(Layer 5). Thus we get control of each led in these 216 LED's.

THE CONTROL CIRCUIT
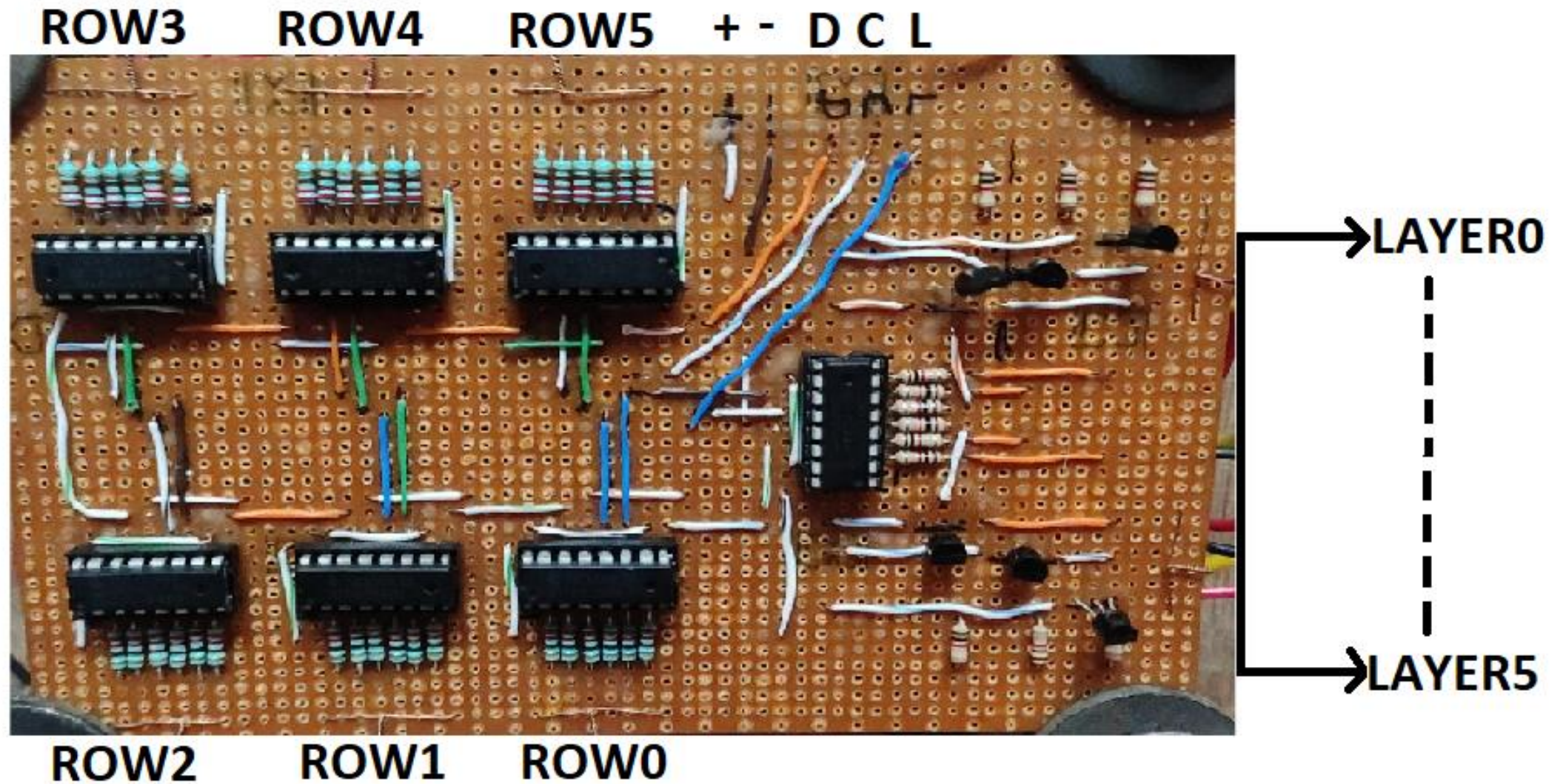
D = DATA, C = CLOCK, L = LATCH, + = VCC, - = GROUND

THANKS