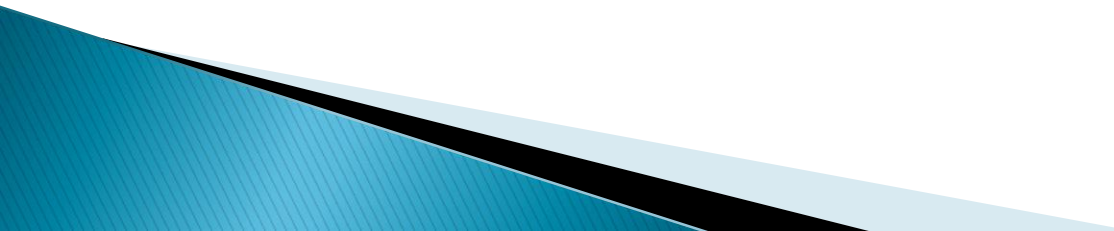


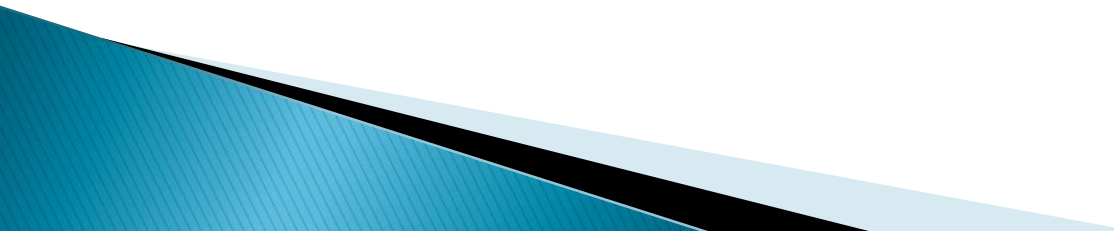
MICROCONTROLLER

UNIT-IV
Lecture-6

KEYBOARD INTERFACING

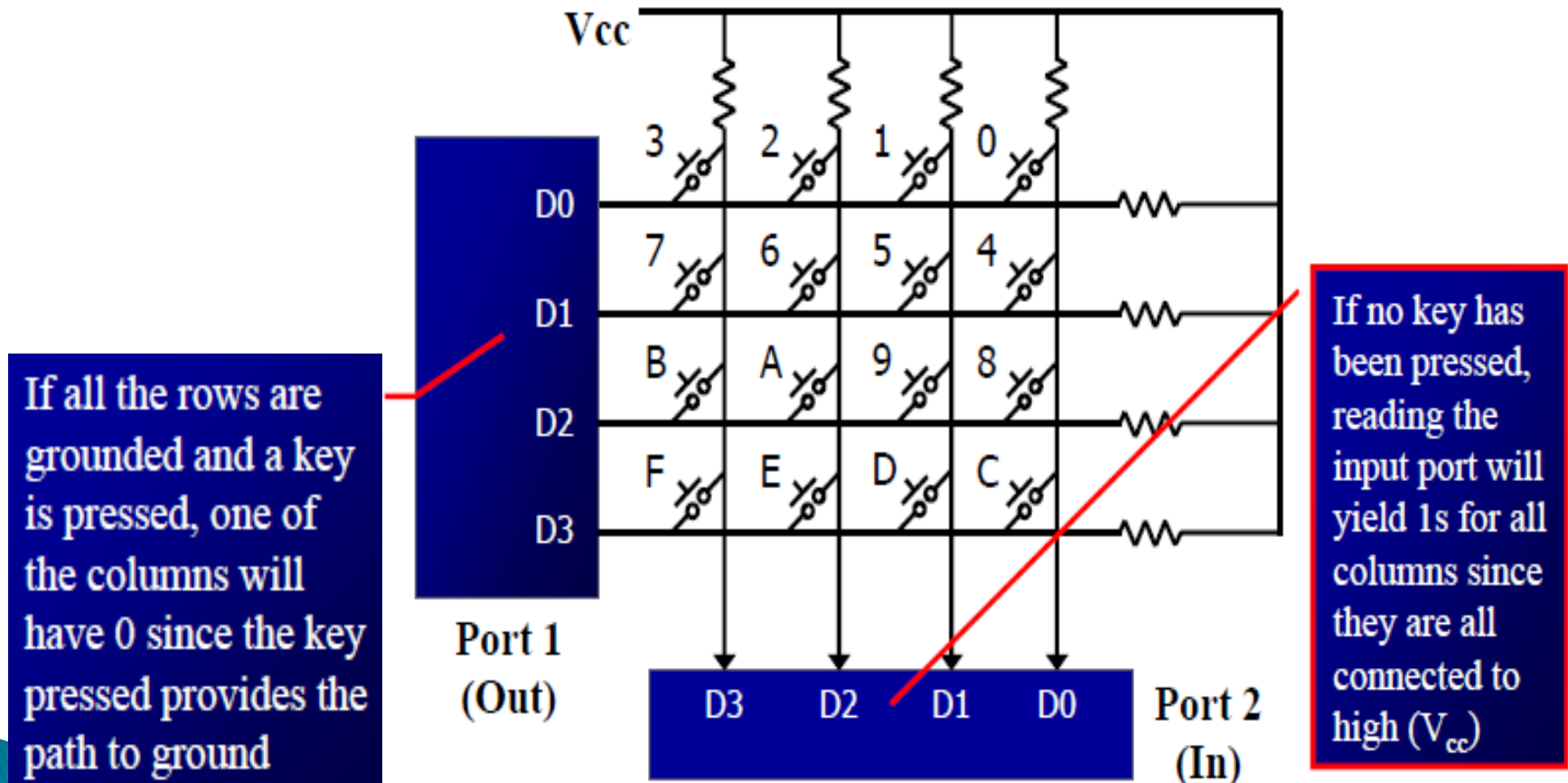
- ▶ Keyboards are organized in a matrix of rows and columns
 - ▶ The CPU accesses both rows and columns through ports
 - ▶ Therefore, with two 8-bit ports, an 8 x 8 matrix of keys can be connected to a microprocessor
 - ▶ When a key is pressed, a row and a column make a contact
- 

Scanning and Identifying the Key

- ▶ Otherwise, there is no connection between rows and columns
 - ▶ In IBM PC keyboards, a single microcontroller takes care of hardware and software interfacing
 - ▶ A 4x4 matrix connected to two ports
 - ▶ The rows are connected to an output port and the columns are connected to an input port
- 

Contd.

Matrix Keyboard Connection to ports



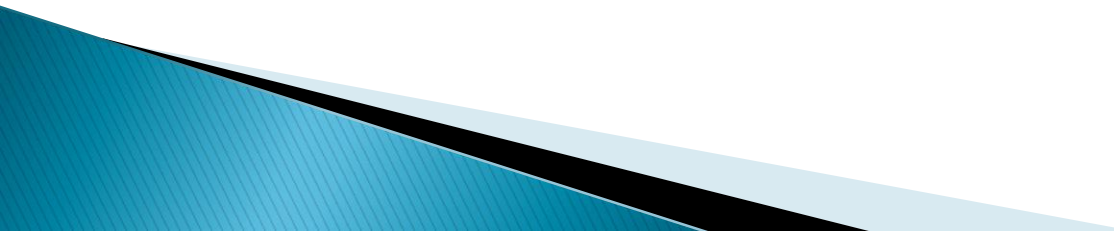
Grounding Rows and Reading Columns

- ▶ To detect a pressed key, the microcontroller grounds all rows by providing 0 to the output latch, then it reads the columns
- ▶ If the data read from columns is $D3 - D0 = 1111$, no key has been pressed and the process continues till key press is detected
- ▶ If one of the column bits has a zero, this means that a key press has occurred
- ▶ For example, if $D3 - D0 = 1101$, this means that a key in the D1 column has been pressed

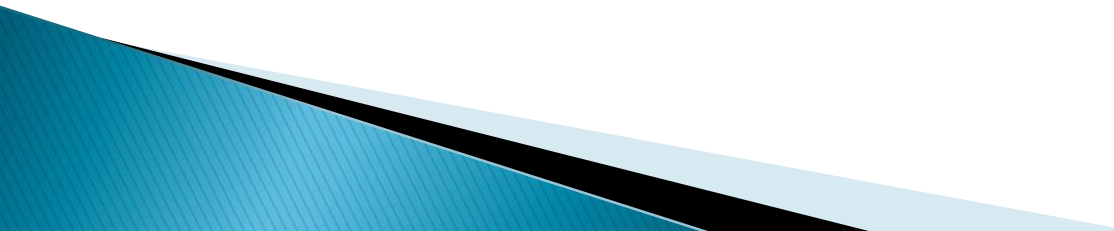
Contd.

- ▶ After detecting a key press, microcontroller will go through the process of identifying the key
- ▶ Starting with the top row, the microcontroller grounds it by providing a low to row D0 only
- ▶ It reads the columns, if the data read is all 1s, no key in that row is activated and the process is moved to the next row
- ▶ It grounds the next row, reads the columns, and checks for any zero

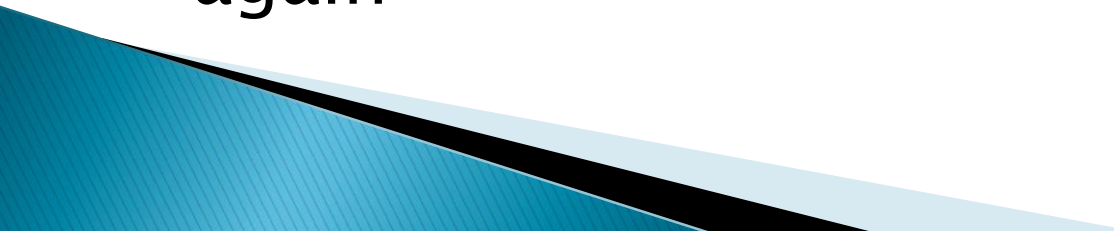
Contd.

- ▶ This process continues until the row is identified
 - ▶ After identification of the row in which the key has been pressed
 - ▶ Find out which column the pressed key belongs to
 - ▶ For detection and identification of key activation goes through the following stages:
- 

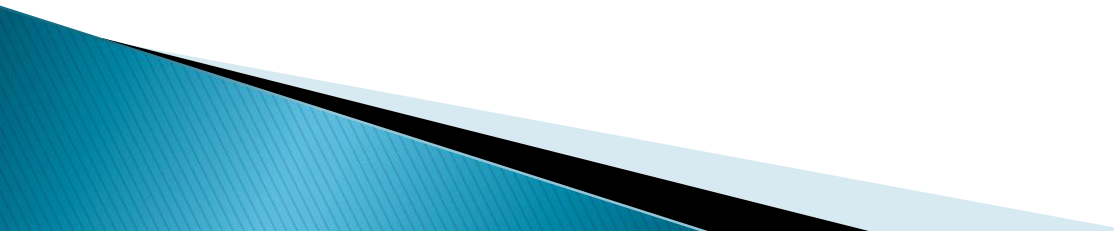
Contd.

- ▶ 1. To make sure that the preceding key has been released, 0s are output to all rows at once, and the columns are read and checked repeatedly until all the columns are high
 - ▶ When all columns are found to be high, the program waits for a short amount of time before it goes to the next stage of waiting for a key to be pressed
- 

Contd.

- ▶ 2. To see if any key is pressed, the columns are scanned over and over in an infinite loop until one of them has a 0 on it
 - ▶ Remember that the output latches connected to rows still have their initial zeros (provided in stage 1), making them grounded
 - ▶ After the key press detection, it waits 20 ms for the bounce and then scans the columns again
- 

Contd.

- (a) it ensures that the first key press detection was not an erroneous one due a spike noise
 - (b) the key press. If after the 20–ms delay the key is still pressed, it goes back into the loop to detect a real key press
 - ▶ To detect which row key press belongs to, it grounds one row at a time, reading the columns each time
- 

Contd.

- ▶ If it finds that all columns are high, this means that the key press cannot belong to that row
 - Therefore, it grounds the next row and continues until it finds the row the key press belongs to
- ▶ Upon finding the row that the key press belongs to, it sets up the starting address for the look-up table holding the scan codes (or ASCII) for that row