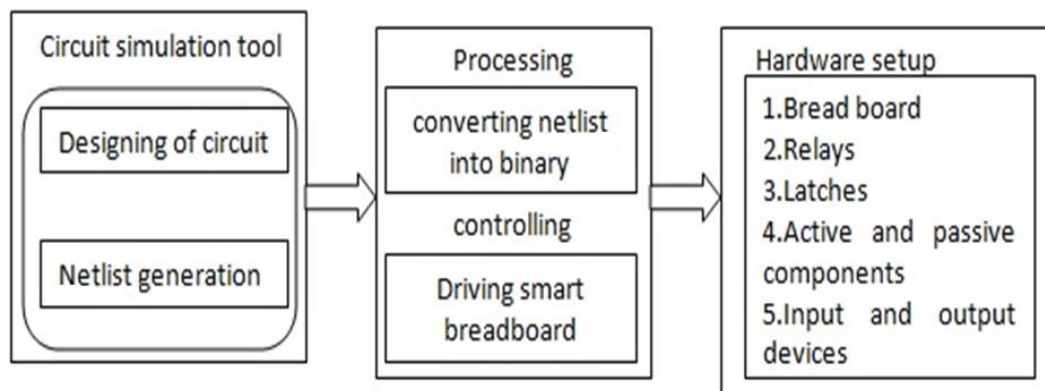


## **HARDWARE DESCRIPTION OF SMART BREAD BOARD SETUP**

### **1.1 Overview**

Virtual labs provide a remotely operated lab environment for students to perform experiments. This can be useful in testing the equipment's and devices remotely. The lab has various equipment that can be remotely started or stopped. The virtual lab can be defined as virtual studying and learning environment that stimulates the real lab. It provides the students with tools, materials and lab sets on computer in order to perform experiments subjectively or within a group at anywhere and anytime.

The circuit simulation tool functions at the client end. A client can login to the V-lab website from the remote place and access the tool to design the test circuit. At the server end, the circuit is rigged up in a smart way without any human intervention and output is made visible to the client with the help of video conferencing. This feature of testing and analyzing the circuits virtually, from the remote place is said to be virtual electronics lab.



**Fig 1:** Block diagram of Virtual electronics lab

## 1.2 Hardware setup

The hardware setup consists of the following components:

- 1) Raspberry Pi 3B+ board
- 2) 1:16 De-multiplexer
- 3) 8:8 D-latches
- 4) Analog switches
- 5) Node lines
- 6) Breadboard and components.

The Multisim tool is used as a circuit designing tool, the required circuit is designed and respective circuit netlist is generated. Once the netlist got generated the format is sent to the V-labs server or host. The acquired netlist is read by the program that drive the **smart bread board**.

The Raspberry pi is used as the controller in this project which has some advanced features in the lights of 26 GPIOs, Ethernet Lan port, Camera port, 4USB ports, HDMI port, SD card extension, etc. The program to read the netlist and to control the relays is developed using software proficiency like embedded C, Python, JAVA, etc.

Once the program reads the netlist the user needs to map the relation between pins of the components and nodelines with respect to netlist generated and standard notation given to the smart bread board setup as shown in the figure 1.2 below:

Standard notation for SBB	A1	A2	A3	A4	B1	B2	B3	B4
Notation in Netlist	T0	D01	R1	CRO P1	T1	D02	R2	CRO P2

Fig 1.2: Mapping Table

After mapping the notations, dump the program to the controller. The controller sends the control signal to de-multiplexer. The control signal indicates the specific latch to be enabled sequentially as per the mapping table via de-multiplexer. 1:16 Demultiplexer consists of four select lines S0,S1,S2,S3. The input given to the select lines is 4bits binary data from 0000 to 1111 that is  $2^4$  yielding 16 outputs. Each of these 16 outputs are given to the LE pins of all the latches that selects one particular latch at a time. This is indicated by **red colored** lines from o/p of De-mux given to the LE of latches. Practically there are 15 latches in the setup. There is a data pin in the Demux that is set high in order to give active high (indicated as logic 1) input to the LE pin of the latch that is selected.

Once the specific latch is selected and its LE pin is activated, the netlist data consisting of 8 bits is to be given to the Analog switches through D-Latches in order to connect pins of the components and nodelines as defined in the 8bit: netlist data. this is indicated by **blue colored** data lines from GPIOs of Raspberry Pi to the i/p of 8:8 D-Latch. The O/P from this latch is given as I/P to the shorted I/P pins of 15 D-Latches

The 15 D-latches are arranged in parallel in the setup, in which all pin1s, pin2s, pin3s,. Pin8s of all the 15 latches are shorted. The hint of setup shown below fig 1.3 :

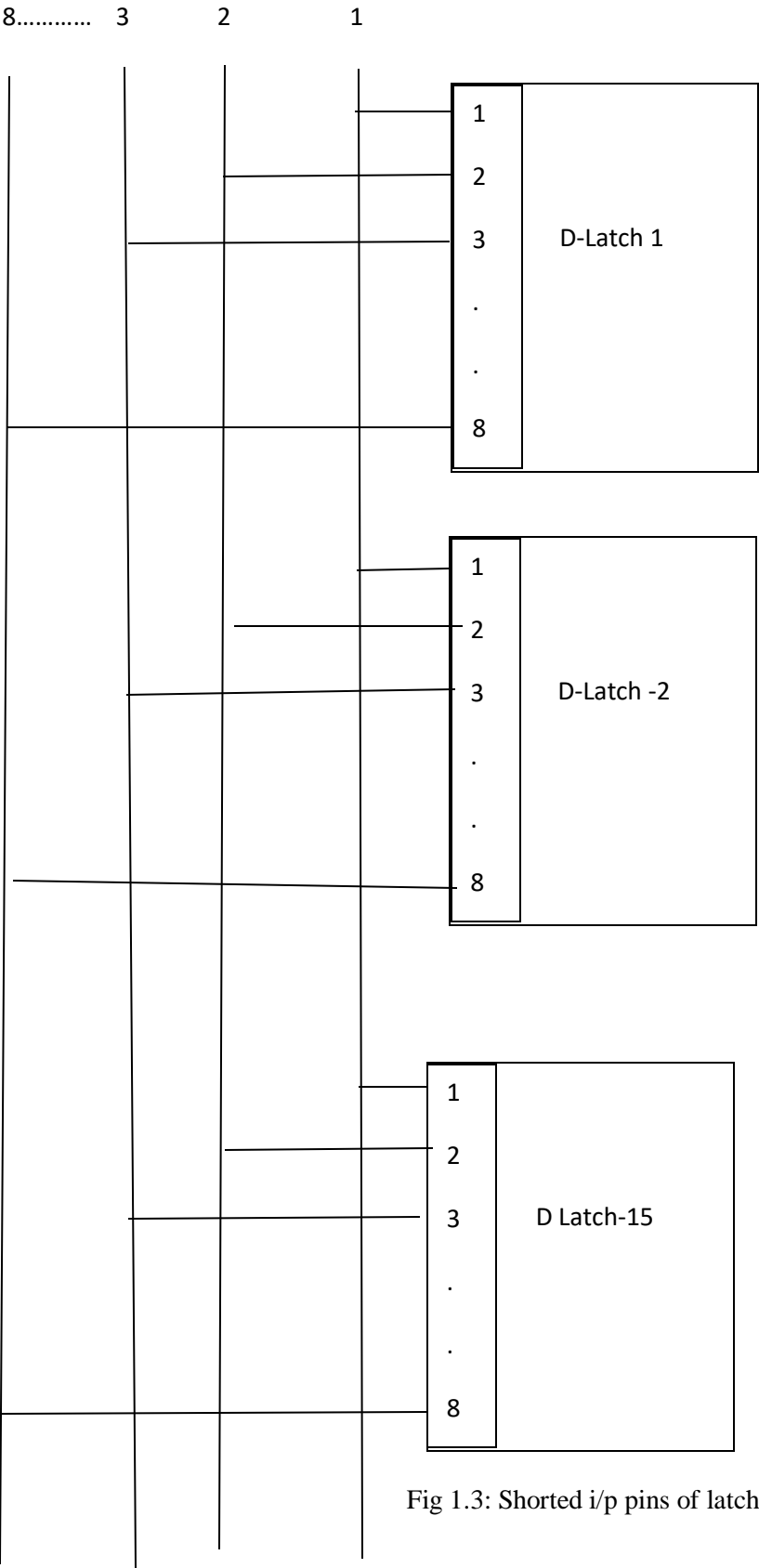


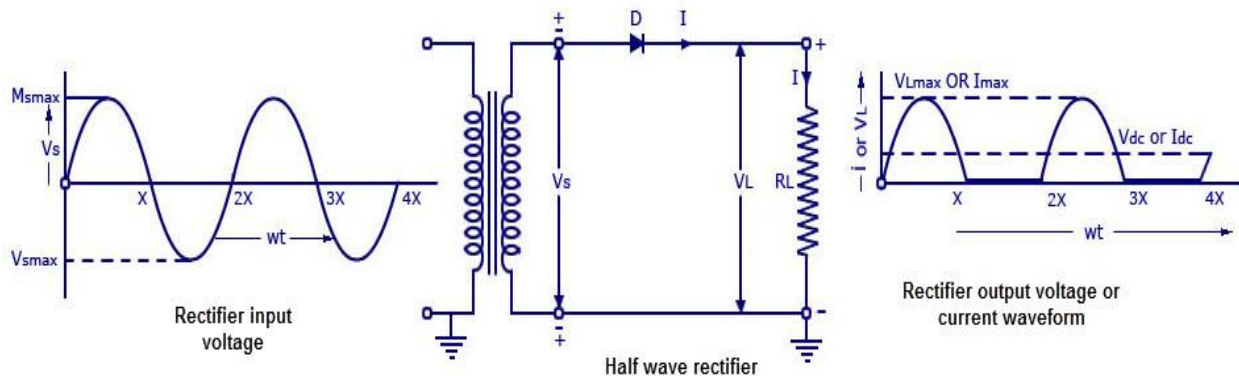
Fig 1.3: Shorted i/p pins of latches

The connection between the particular pins of the component and node lines is established by a switch. One switch connects a pin of a component to a node. In this hardware setup there are 6 nodes and 10 components. Therefore it requires 120 switches in order to connect all the pins to all the nodes as per the formula below,

$$\text{No of Switches} = 2(\text{No of Components} * \text{No of nodes})$$

Each analog switch consists of four switches. There are 8 O/P pins in D-Latches. These 8 O/Ps are given as I/P to 8 switches. Thus, one D-latch is capable of driving two analog switches. A Relay or switch consists of four terminals namely NC, NO, Common, Pin. NC is connected to a node in node lines setup and NO is connected to a pin of the component in the bread board. Pin is connected to O/P of the latch. Whenever power flows into the Pin, NC and NO terminals are shorted which means that connection is established between pin and node. Finally the entire circuit is rigged up in smart way by the relays driven by the controller.

### 1.3 Precautions



Consider the above half wave rectifier circuit shown above, the mapping should be done and precautions are taken in following way:

- Initially all the nodes and components are connected each other with the help of relay board.
- The diode node terminal is activated first
- Once the diode got activated then the resistor node is activated.
- If there are some capacitors and other passive components, activate them.
- After all the passive components got activated, then activate the active components like AC or DC voltage source.
- Finally activate the output device like CRO, Multimeter.

**Note:** If initially the AC voltage source is activated means, the circuit will get burst, So the connection is likely to be done as stated above by connecting passive component first.