1. VOICE COMMUNICATION DEVICE (MODEL: DM-VC-1)

1.1 INTRODUCTION

The portable wireless communication device (Fig.1) supports full-duplex communication, which allows both miners to speak and listen simultaneously unlike walky-talky. Miners tacking and voice communication system helps to monitor miners current location, path history display, emergency notification to the control room, warning of miners entry into the restricted area, locating of trapped miners, full-duplex voice communication, voice broadcasting to the underground miners, display of video footage of important underground mine locations, etc. These IP-based communication devices are used to communicate with a particular miner and also with multiple miners when broadcasting is required. The developed communication device is based on Raspberry Pi model 3, which has a RAM of 4 GB and 4× ARM Cortex-A53, 1.2 GHz processor. Figure 1 represents the block diagram of voice communication system.

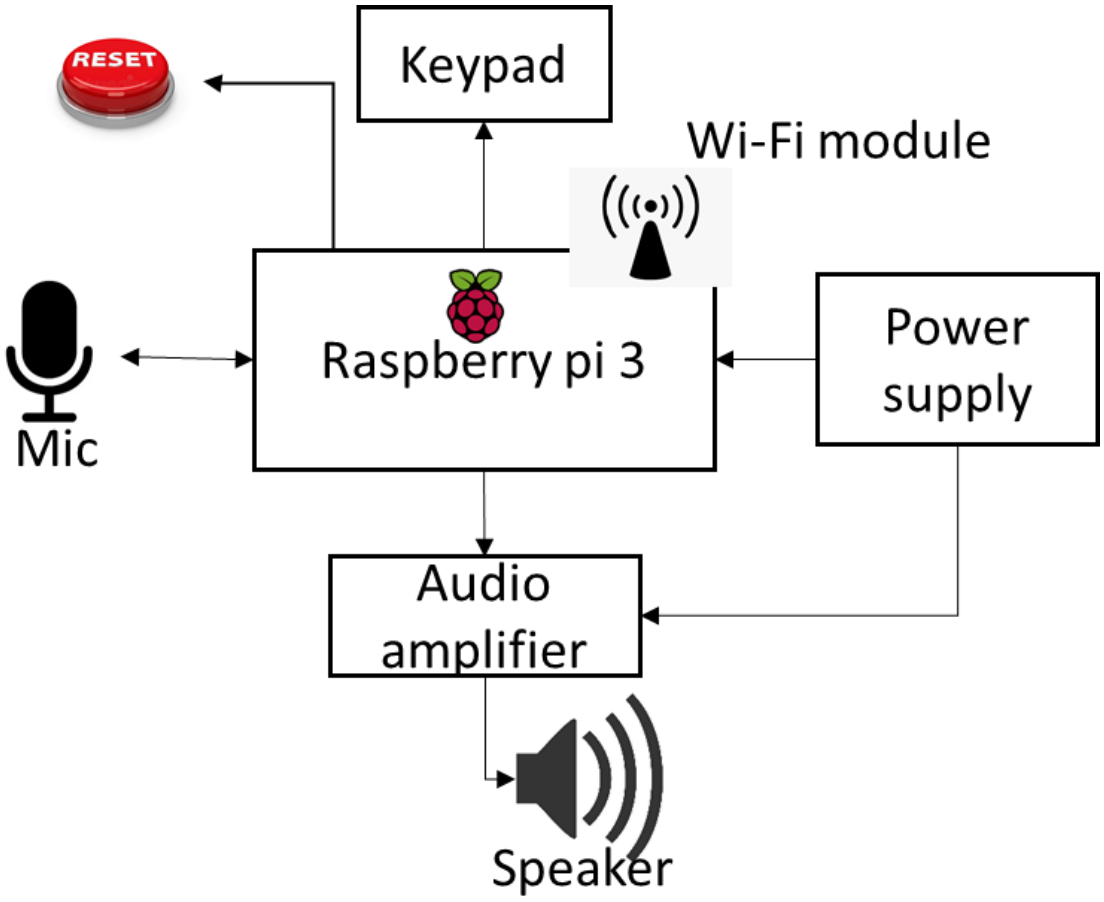


Figure 1 Block diagram of water level monitoring unit

1.2 PROGRAMMABLE COMPONENTS

I. Raspberry pi (3 Model B+)

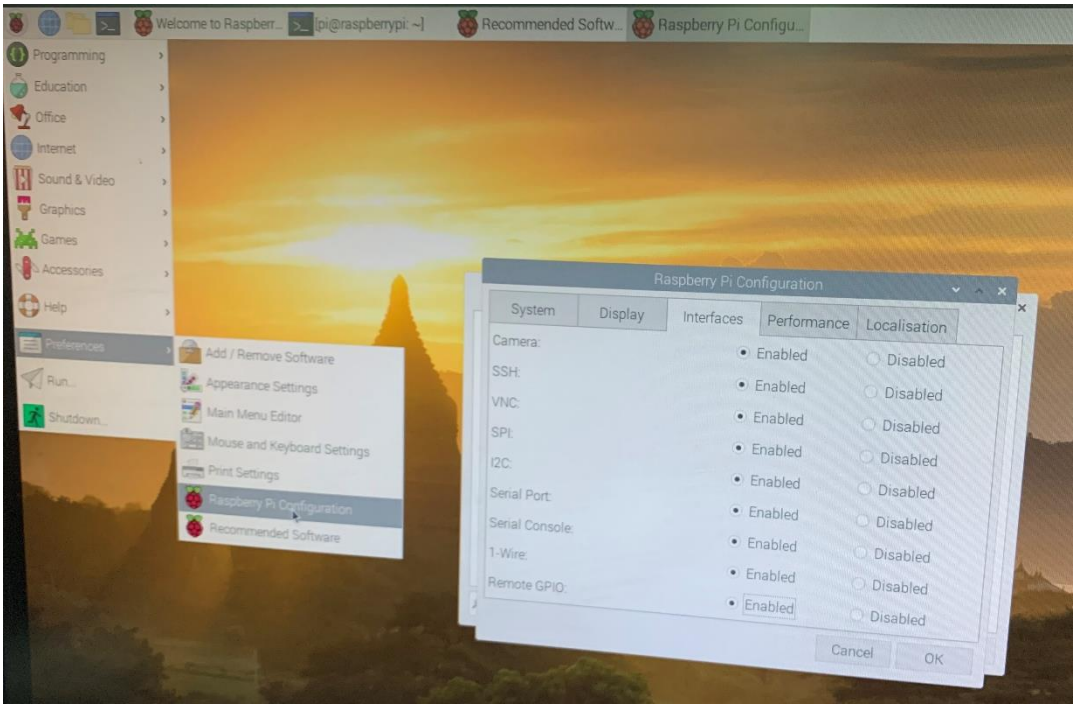
1.3 SOFTWARE REQUIREMENTS ON PC

i. MATLAB R2017b and later versions with the support packages given below:

- MATLAB Support Package for Raspberry Pi Hardware
- Simulink Support Package for Raspberry Pi Hardware
- Raspberry Pi Hardware Resource Manager

1.4 PREREQUISITES FOR CODE INSTALLATION

- Raspberry pi with Raspberry pi (32 bit) operating system
- 'SSH' should be enabled on Raspberry pi. Open Preferences>Raspberry Pi Configuration>Interfaces and enable 'SSH' and other required interfaces as shown below.



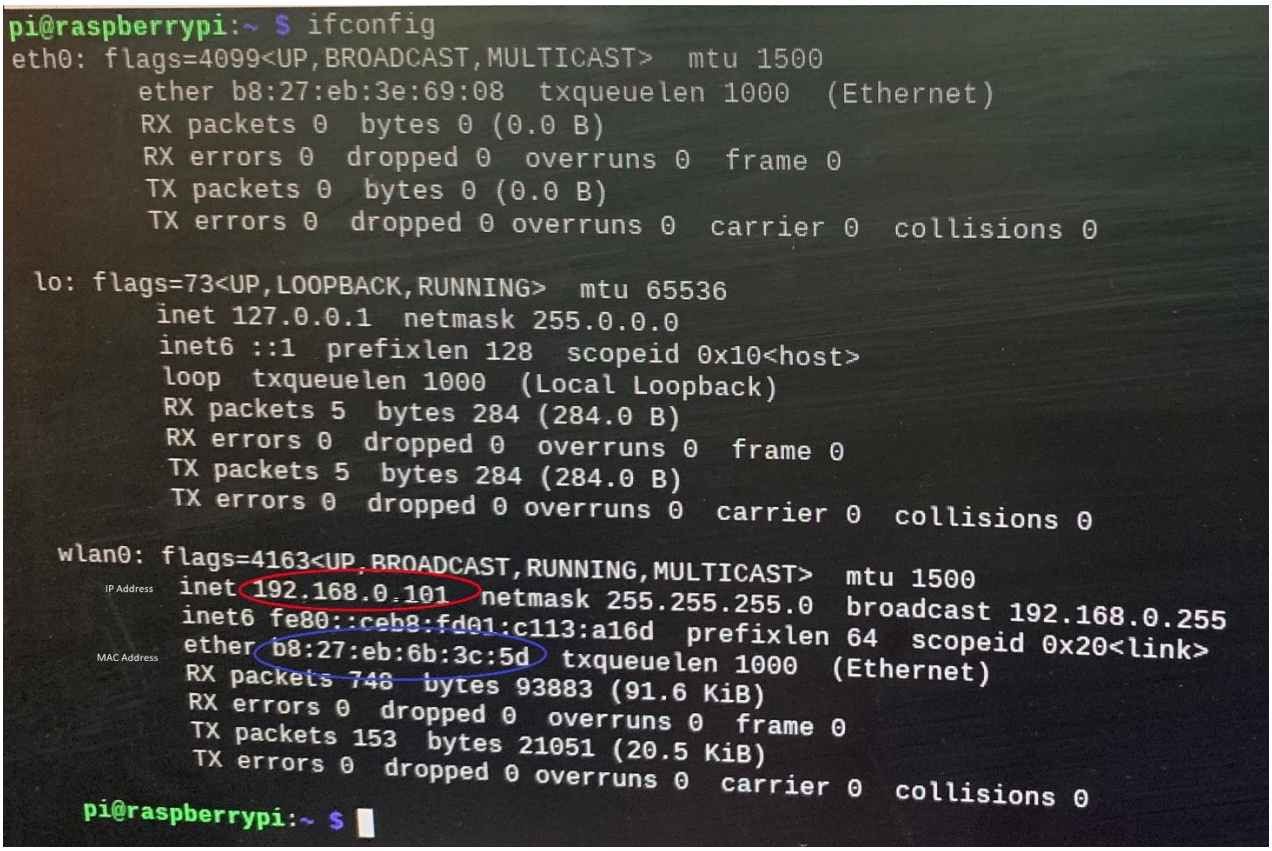
- Raspberry pi and the computer with MATLAB installed should be connected to the same wireless network with or without Internet access. Internet access is required on both devices only during "MATLAB Support Packages for Raspberry Pi Hardware" installation on raspberry pi from MATLAB.

1.5 CODE INSTALLATION PROCEDURE

1 Install 'MATLAB Support Packages for Raspberry Pi Hardware on raspberry pi from MATLAB: To install MATLAB support packages for raspberry pi, following prerequisites and steps are required:

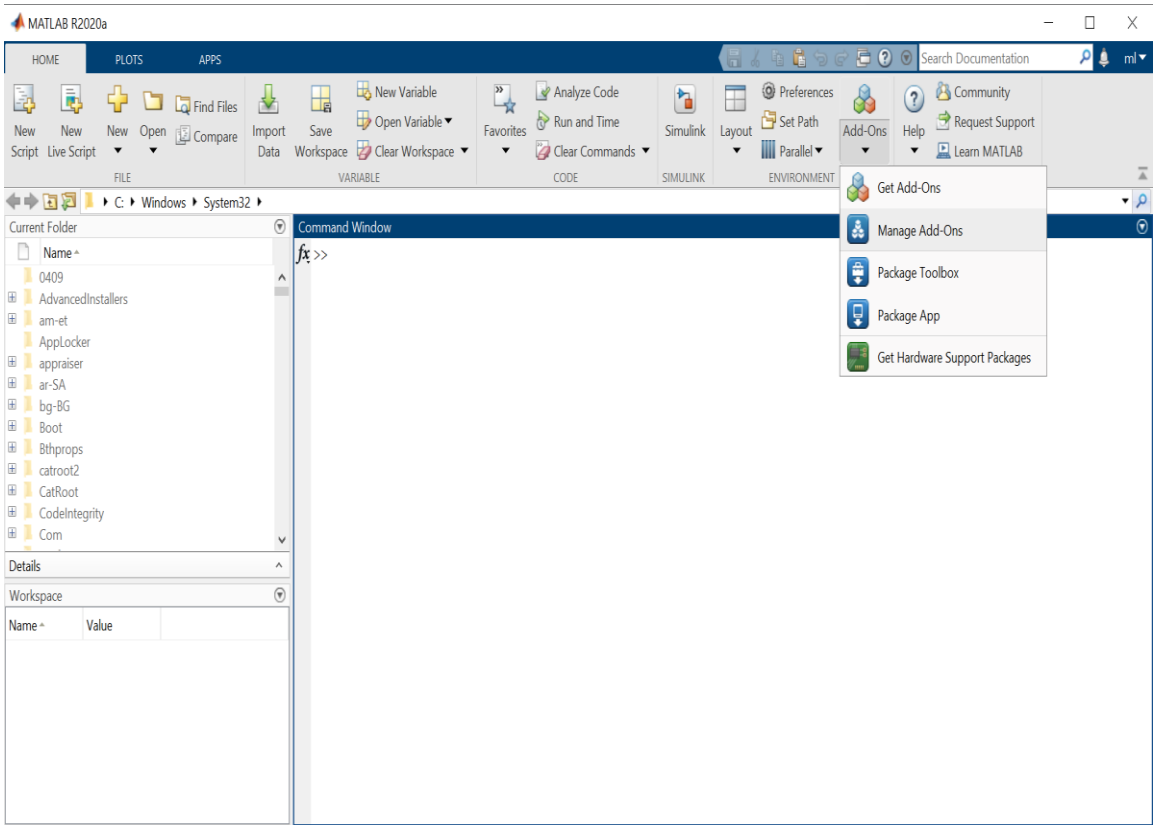
PREREQUISITES:

- Raspberry pi (RPI) and the computer on which MATLAB is installed should be connected to the same wireless network with Internet access.
- IP address and MAC address of Raspberry pi required for below steps should be checked by "ifconfig" command in RPI's Command Line as shown below. Ip address and MAC address obtained for sample device :
  IP Address : 192.168.1.101
  MAC Address: b8:27:eb:6b:3c:5d

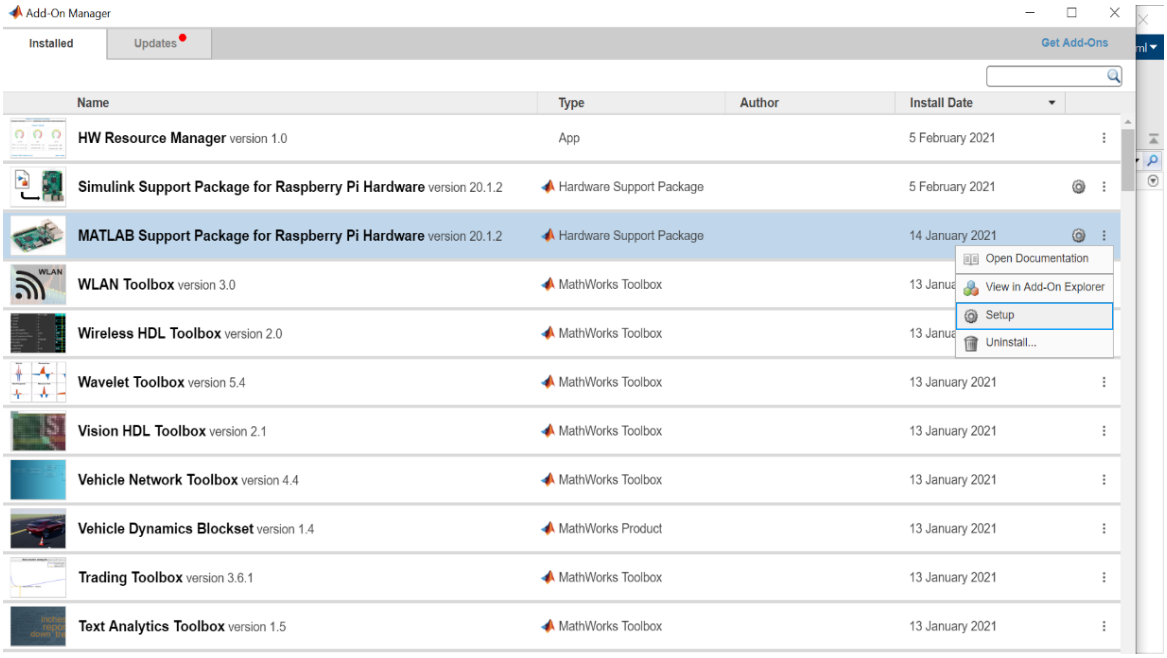


STEPS:

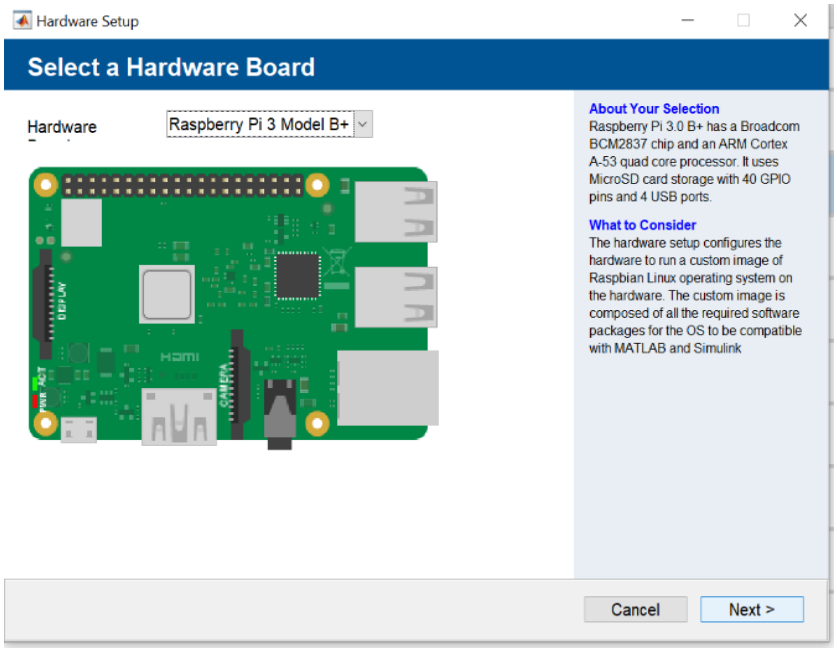
a. Open MATLAB and then click on Add-Ons>Manage Add-ons> present in the Home section of the menu on the top.



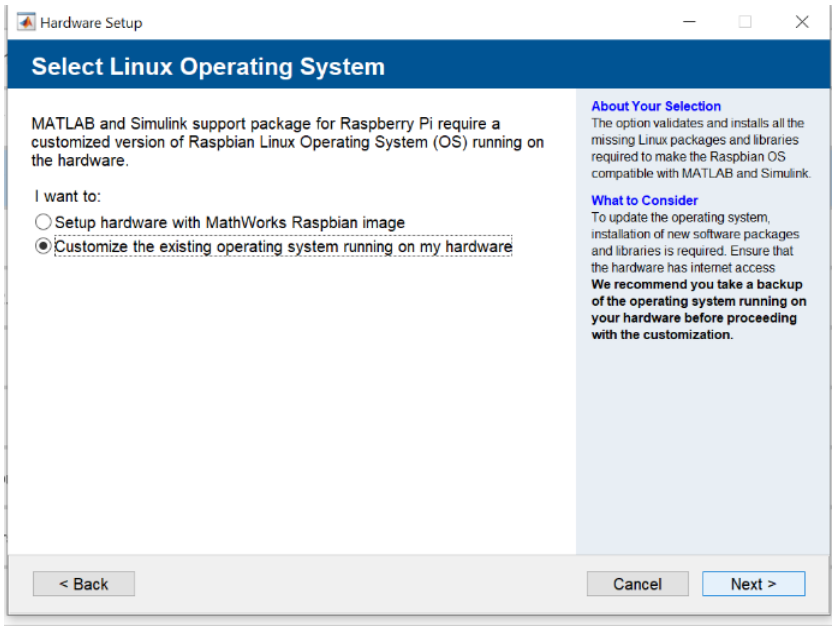
b. Select “MATLAB Support Packages for Raspberry Pi Hardware” and right-click on it and then select “Setup” from the drop-down list.



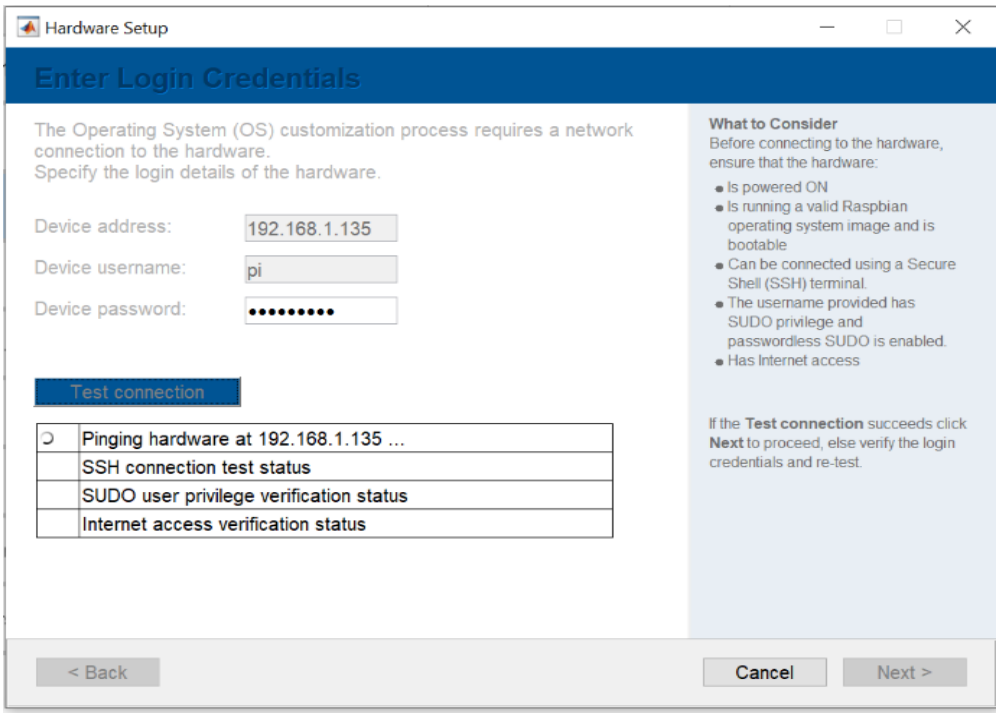
c. And a new window “Hardware Setup“ pops as shown below. Select “Raspberry Pi 3 Model B+” in Hardware and click Next.



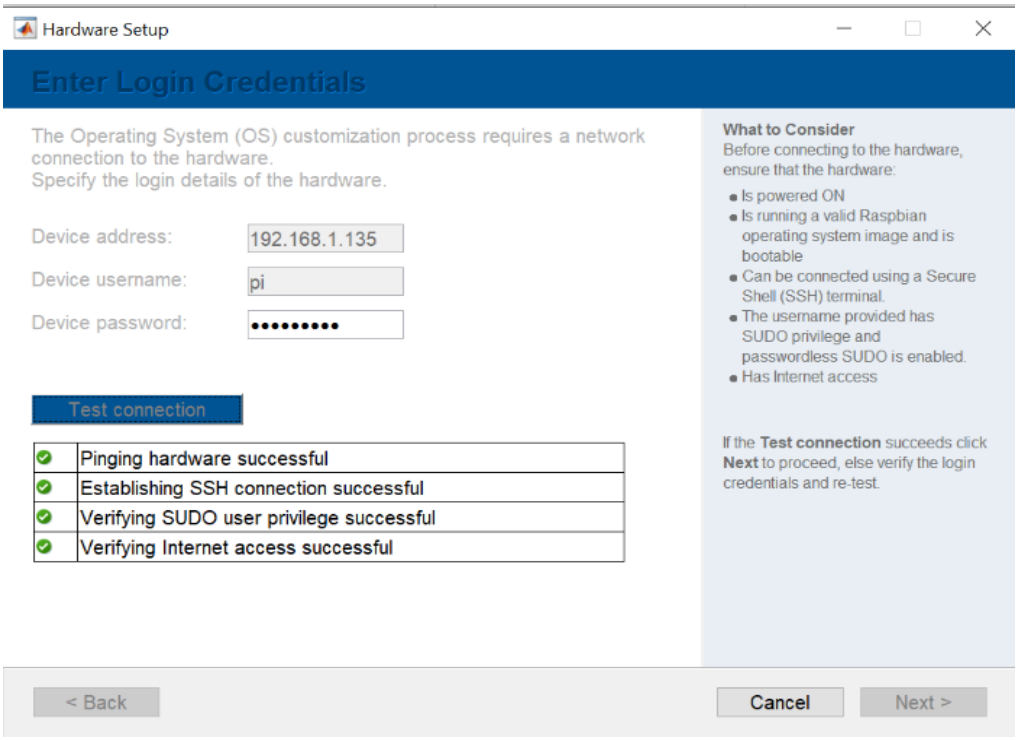
d. Select the second option i.e., “Customize the existing operating system running on my hardware” and click on Next.



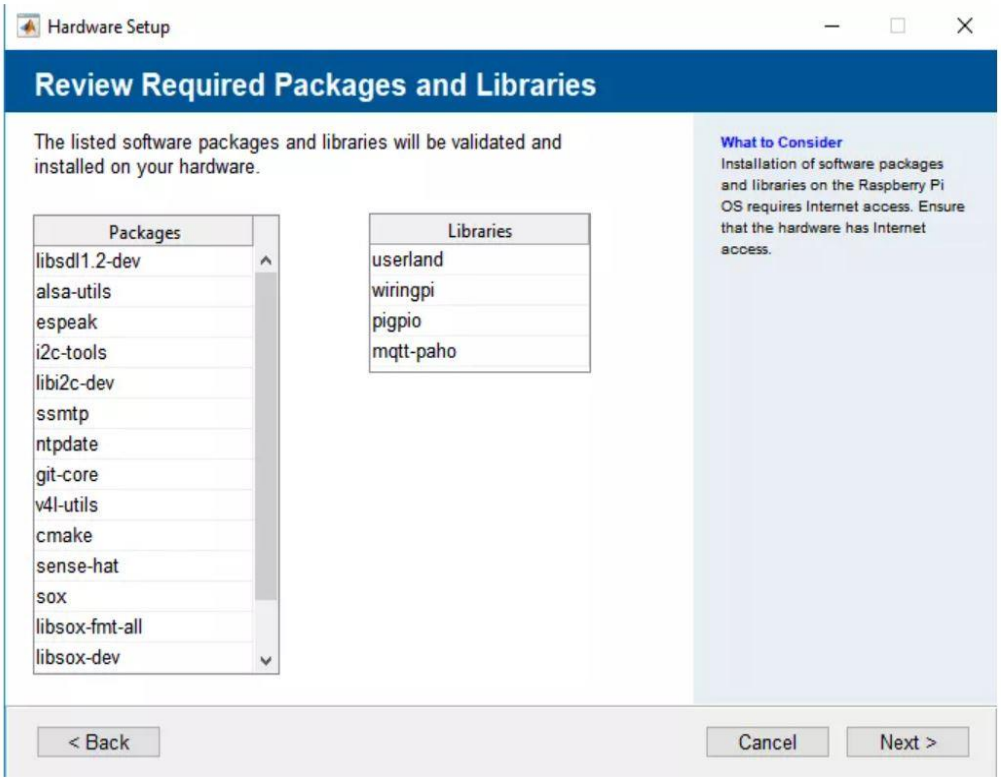
e. Then enter the IP address of your Raspberry Pi in “Device address”, user ID of Raspberry pi which is “Pi” by default in “device username” and password of Raspberry pi which is “raspberry” by default in “device password” and then click on Test connection button.



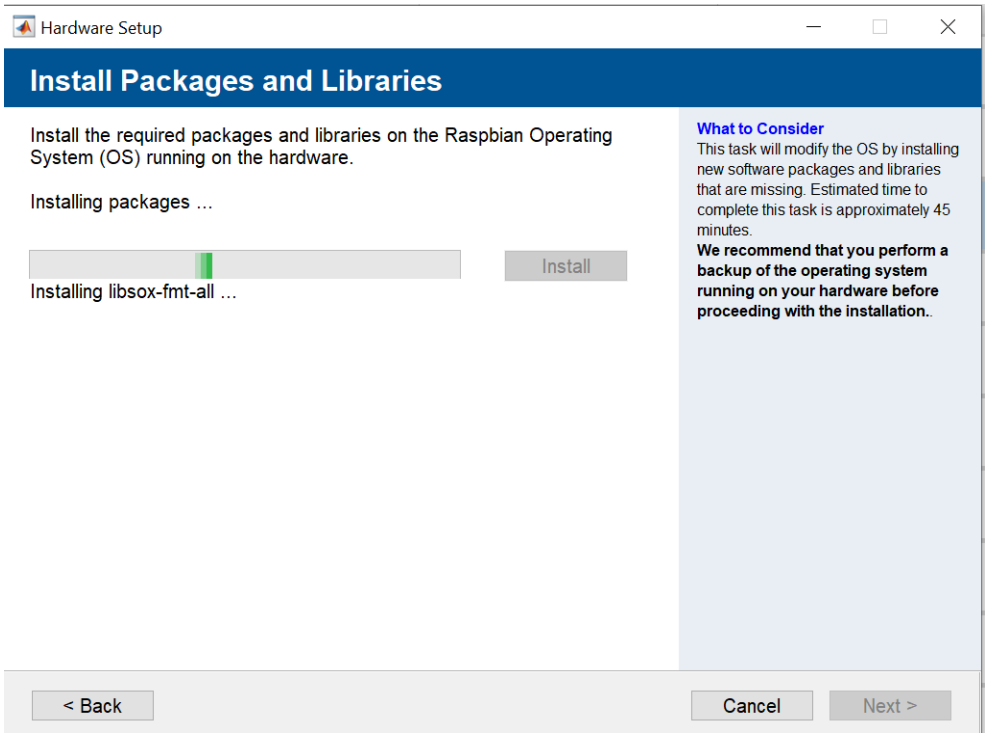
f. Then if MATLAB is able to connect to Raspberry Pi then below figure will be shown.



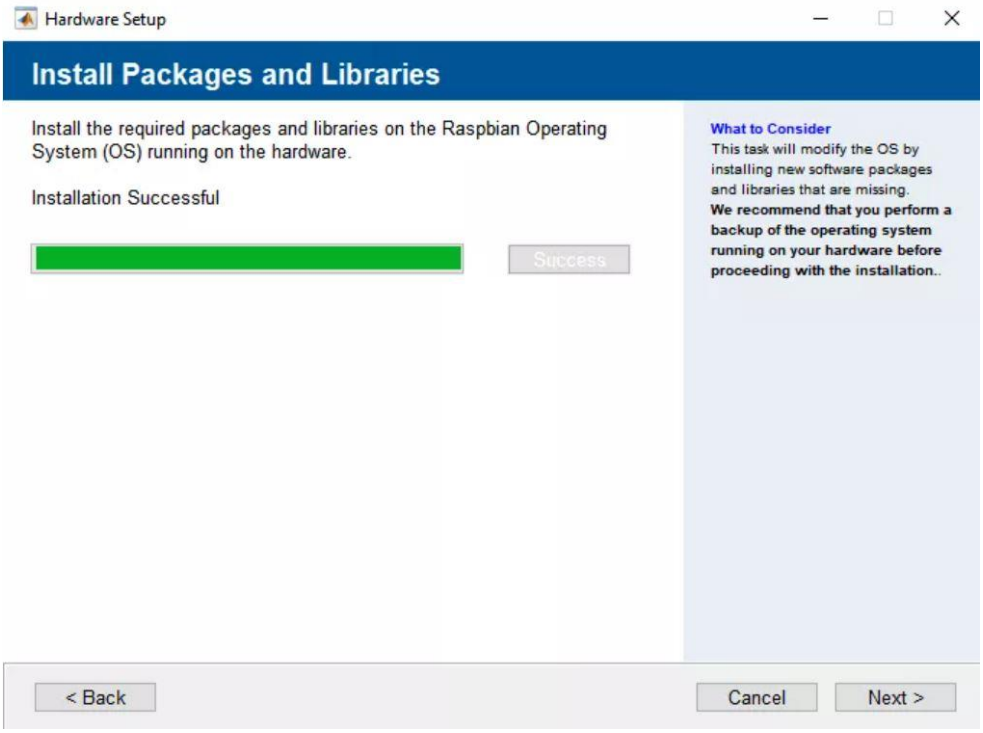
g. Click Next to proceed to the next window, where all the required software packages are listed in two different columns as shown below



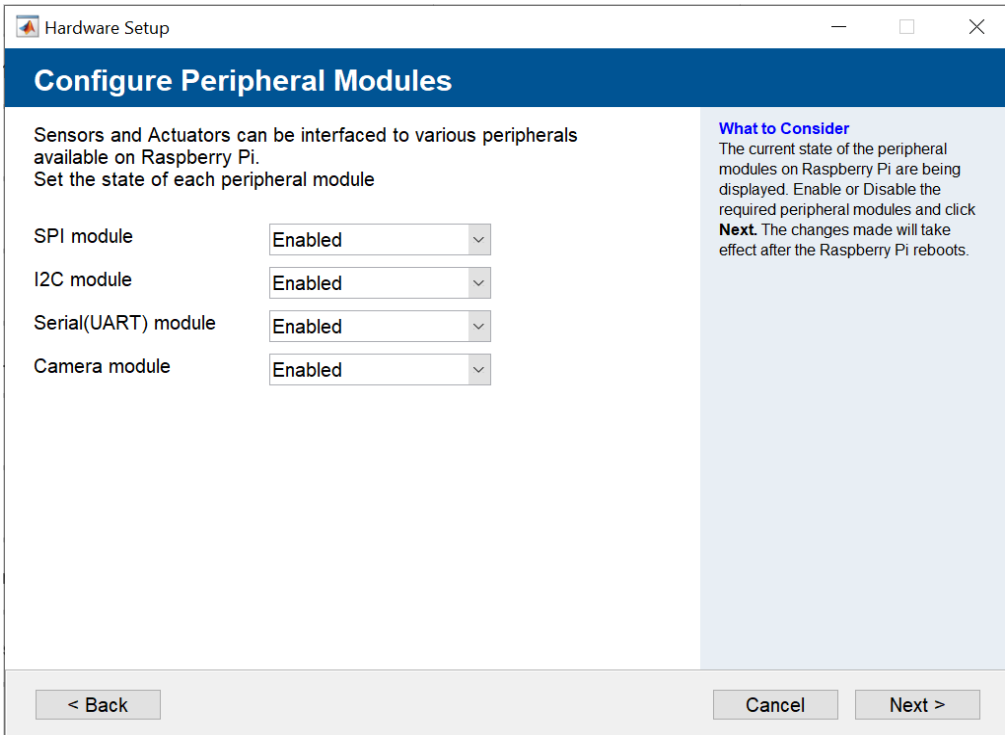
h. Click Next, then there will be an Install button in the window as shown below. Click on Install to install the software packages and libraries.



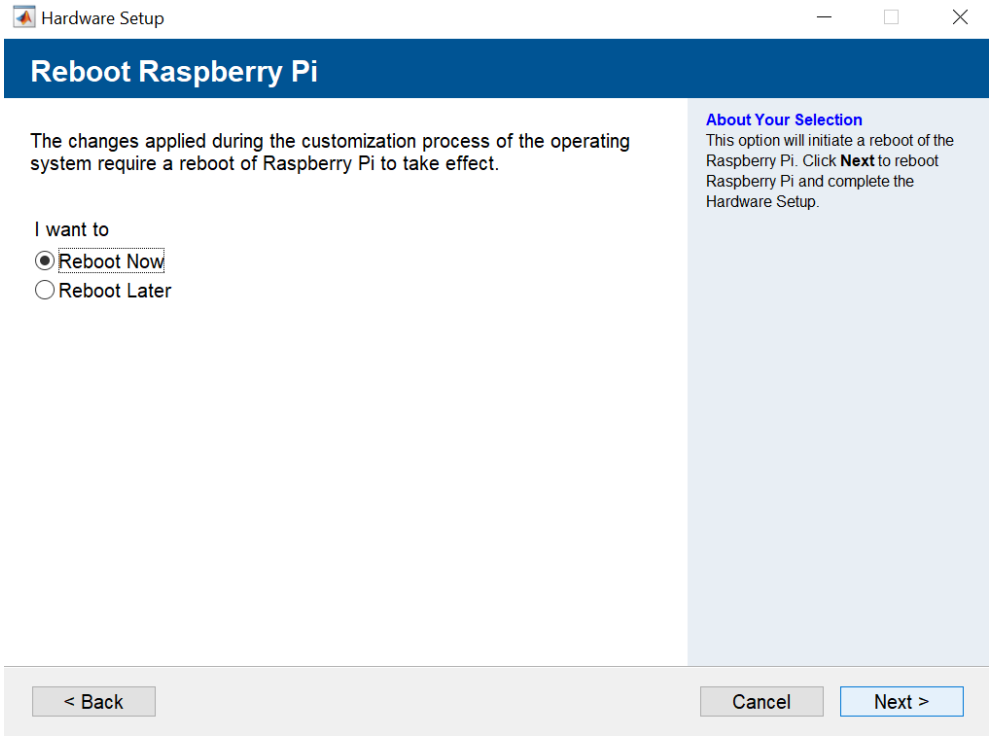
i. After successful installation, click on next



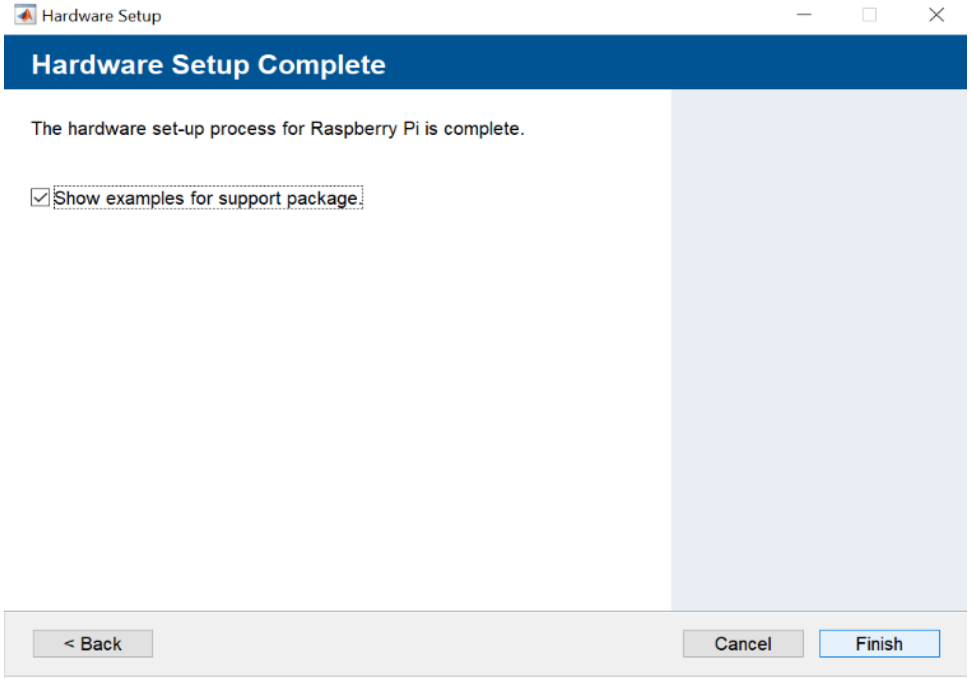
j. After clicking Next, we get “Configure Peripheral Modules’ . Select ‘Enabled’ for all the four options and click on Next.



k. Select ‘Reboot Now’ and click on Next.

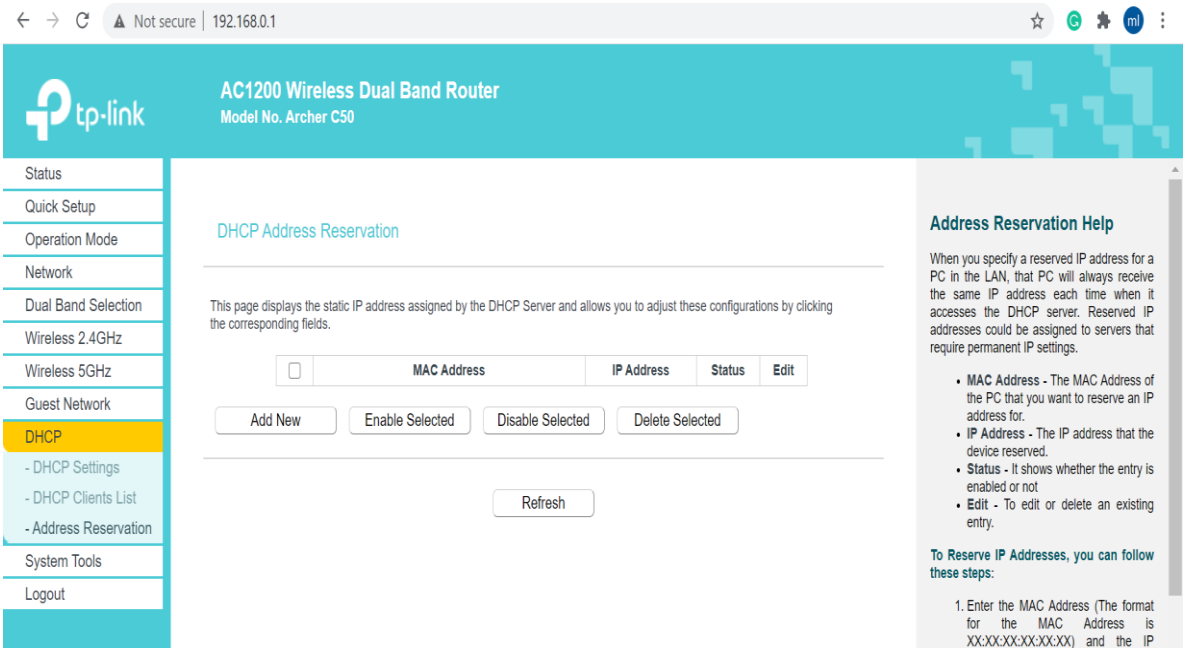


The ‘Hardware setup complete’ tab as shown below will appear at the end of the libraries setup process.

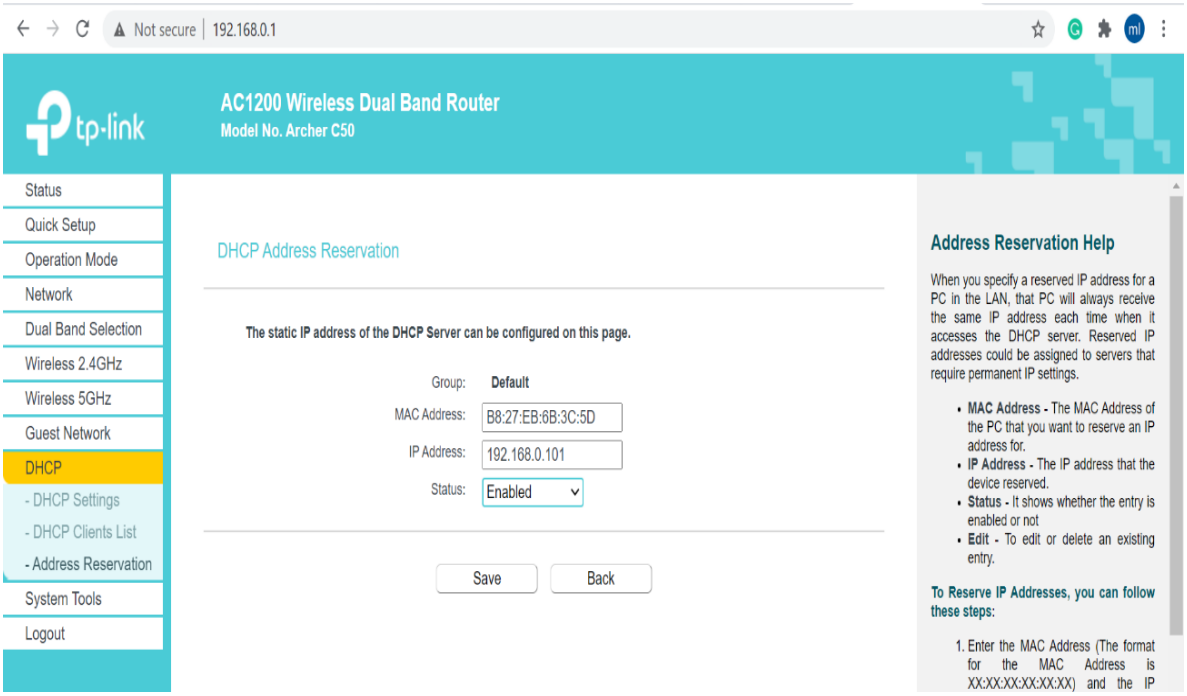


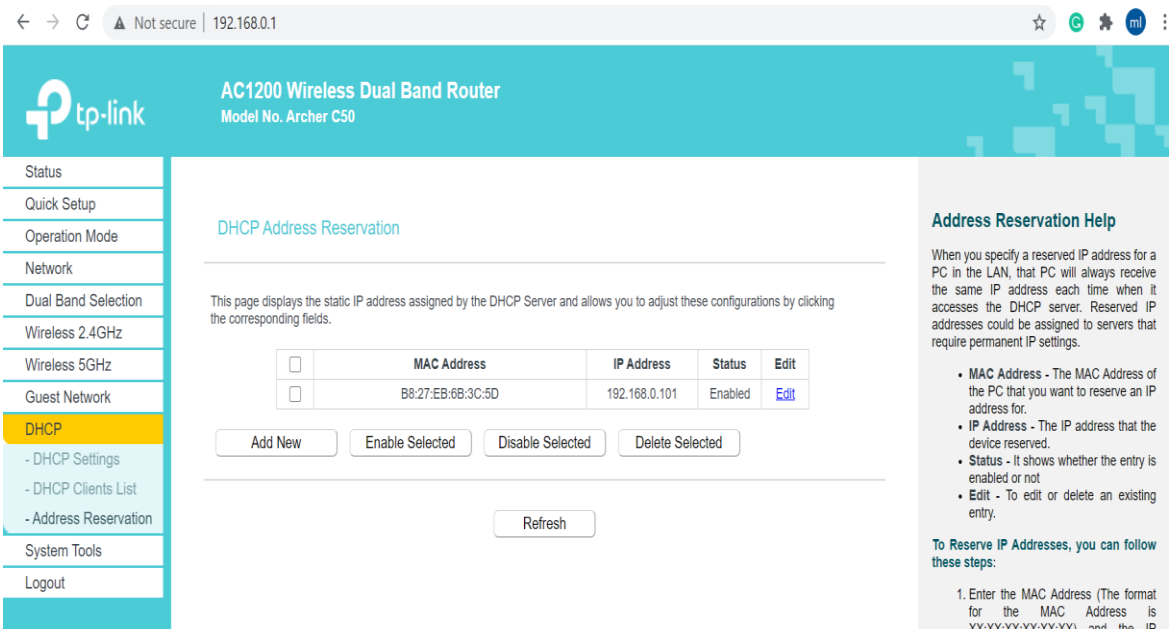
2 Assign static ip address to Raspberry Pi through ‘Address Reservation’ on access point’s setup page. Below steps are required to be followed:

a. Open Access point's (Router) setup page and go to DHCP >Address Reservation tab as shown below.

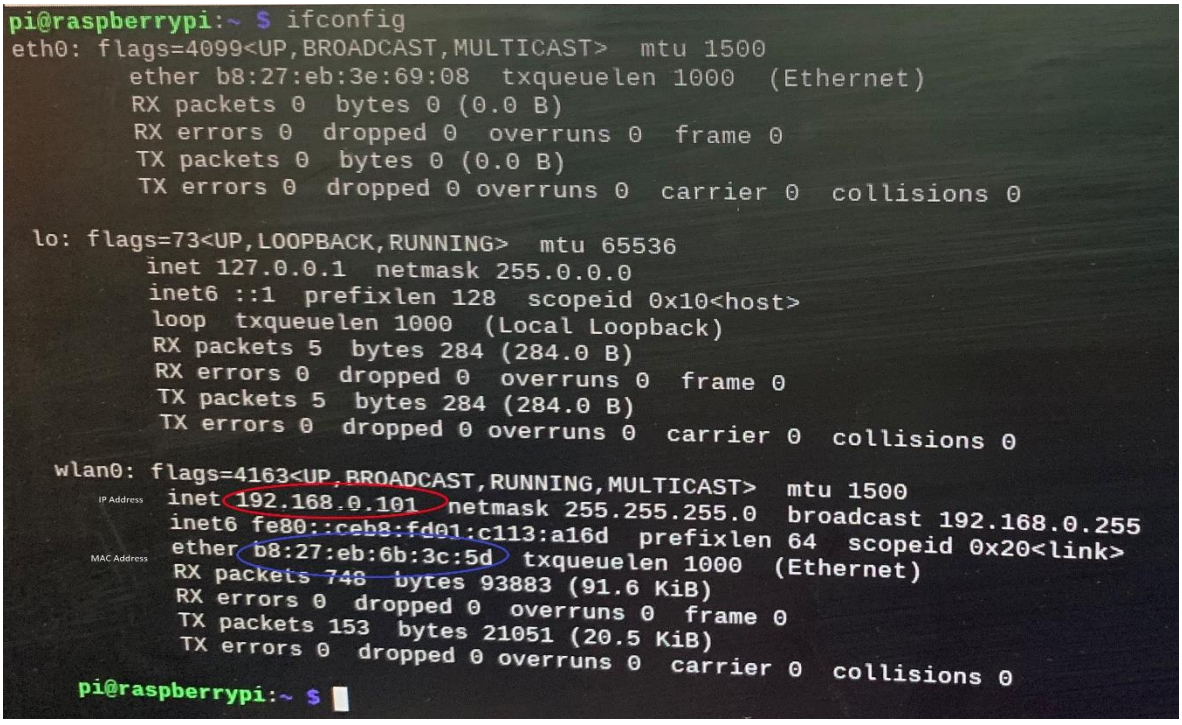


b. Click 'Add New' and add required details ('Desired IP address', 'MAC Address of RPI' : Check IP and MAC address through 'ifconfig' command on RPI) and select 'Enable selected' in status as shown below and then click on Save.



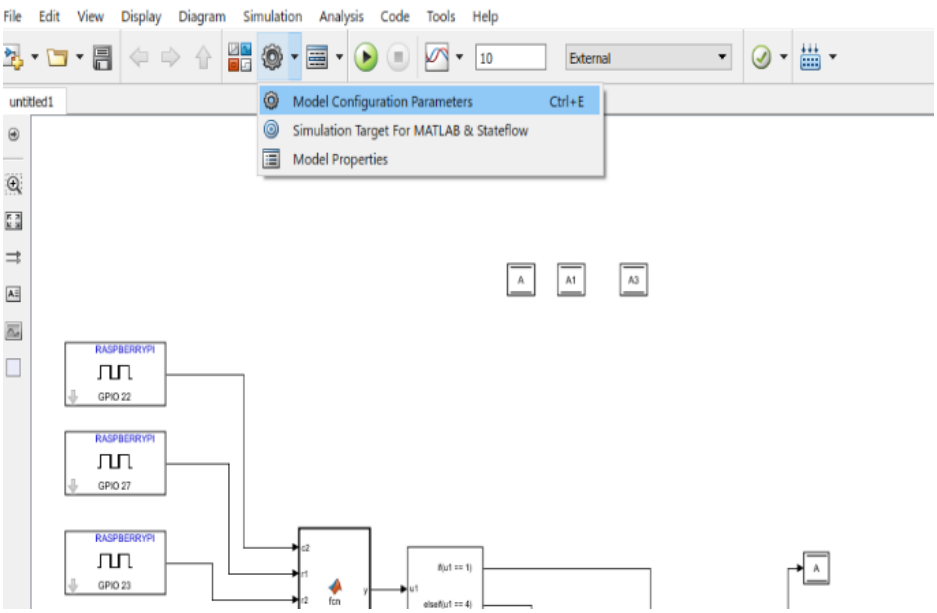


3 Check IP address again for confirmation on Raspberry Pi as shown below.



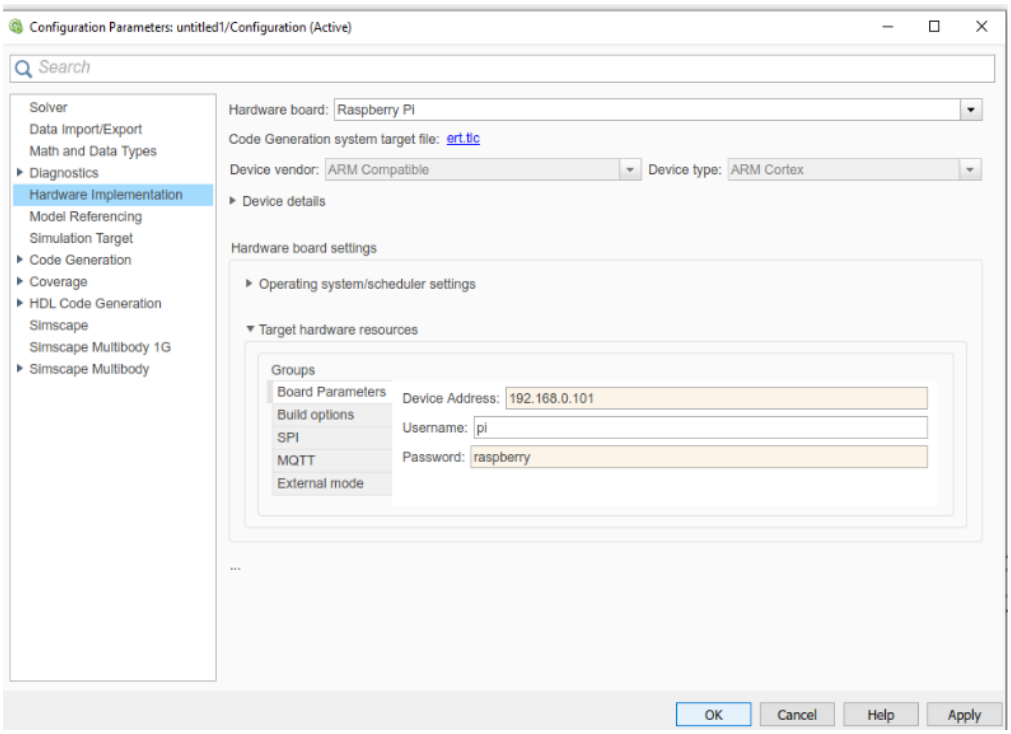
4 Open SIMULINK Model of the program to be installed on Raspberry Pi.

5 Open Model configuration parameters from settings tab as shown below.

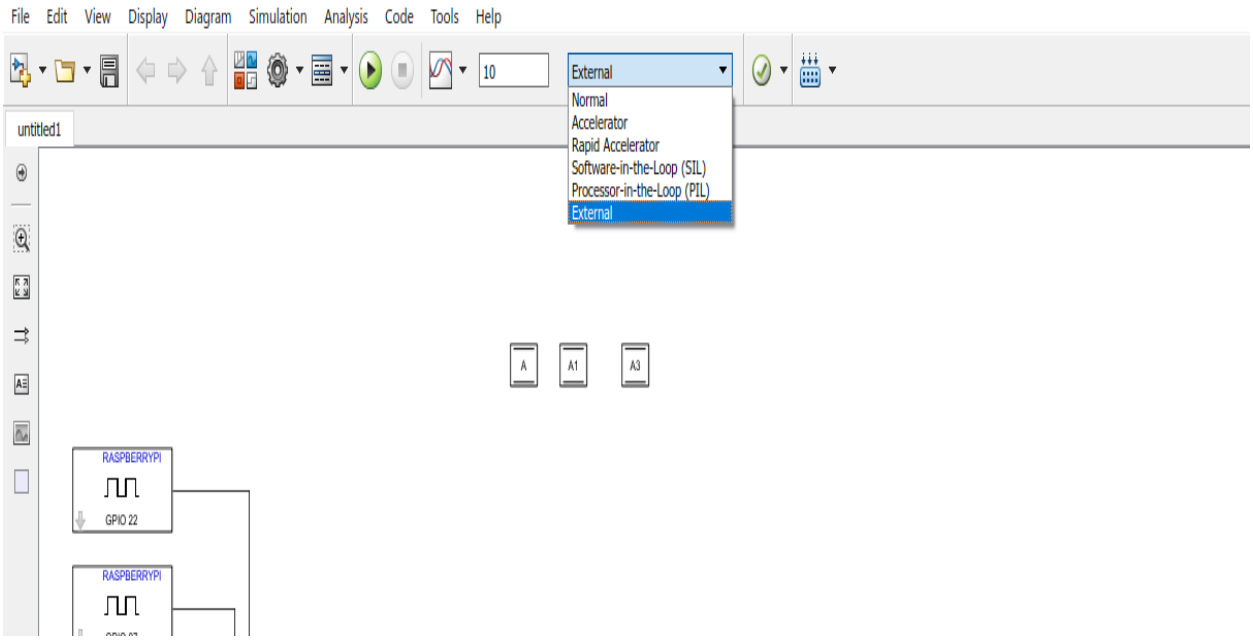


6 Go to ‘Hardware Implementation’ and select ‘Raspberry Pi’ in Hardware board.

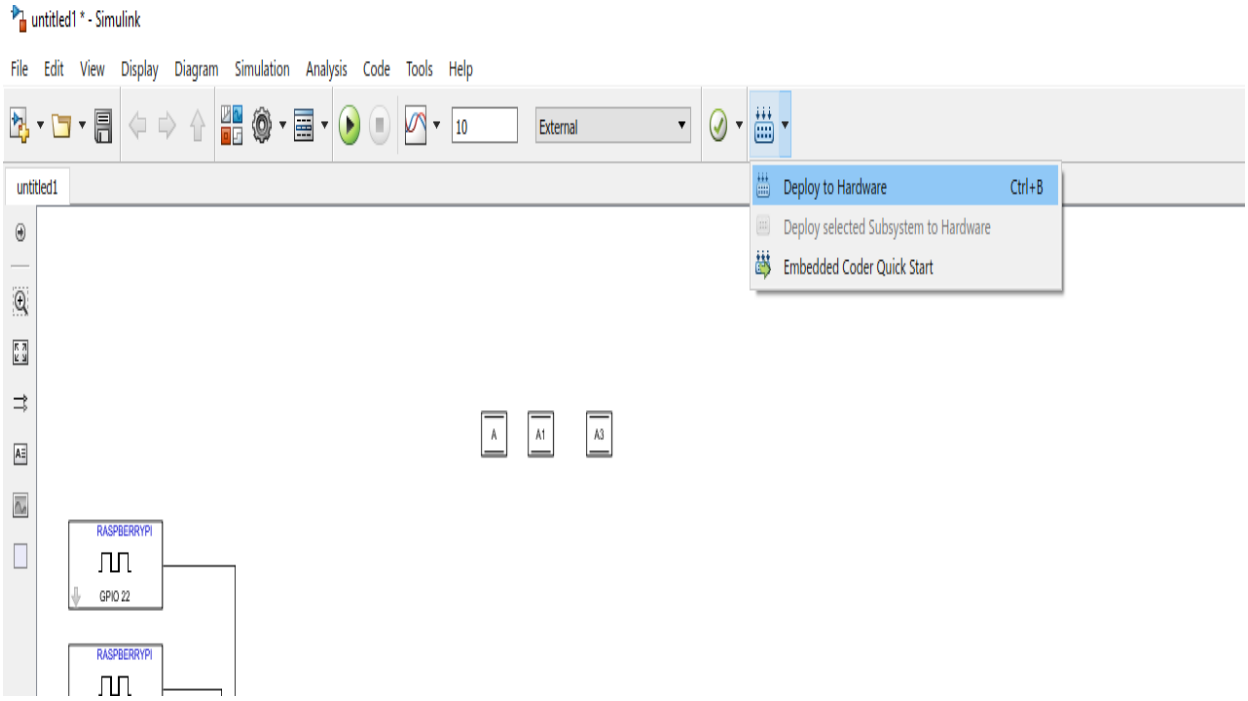
- Add IP address of RPI on which code has to be executed in Target hardware resources>Board parameters>Device Address
- Add ‘pi’ in username as this field requires Username of RPI on which code has to be executed. (Default username of RPI is ‘pi’)
- Add ‘raspberry’ in password as this field requires Password of RPI on which code has to be executed. (Default password of RPI is ‘raspberry’)



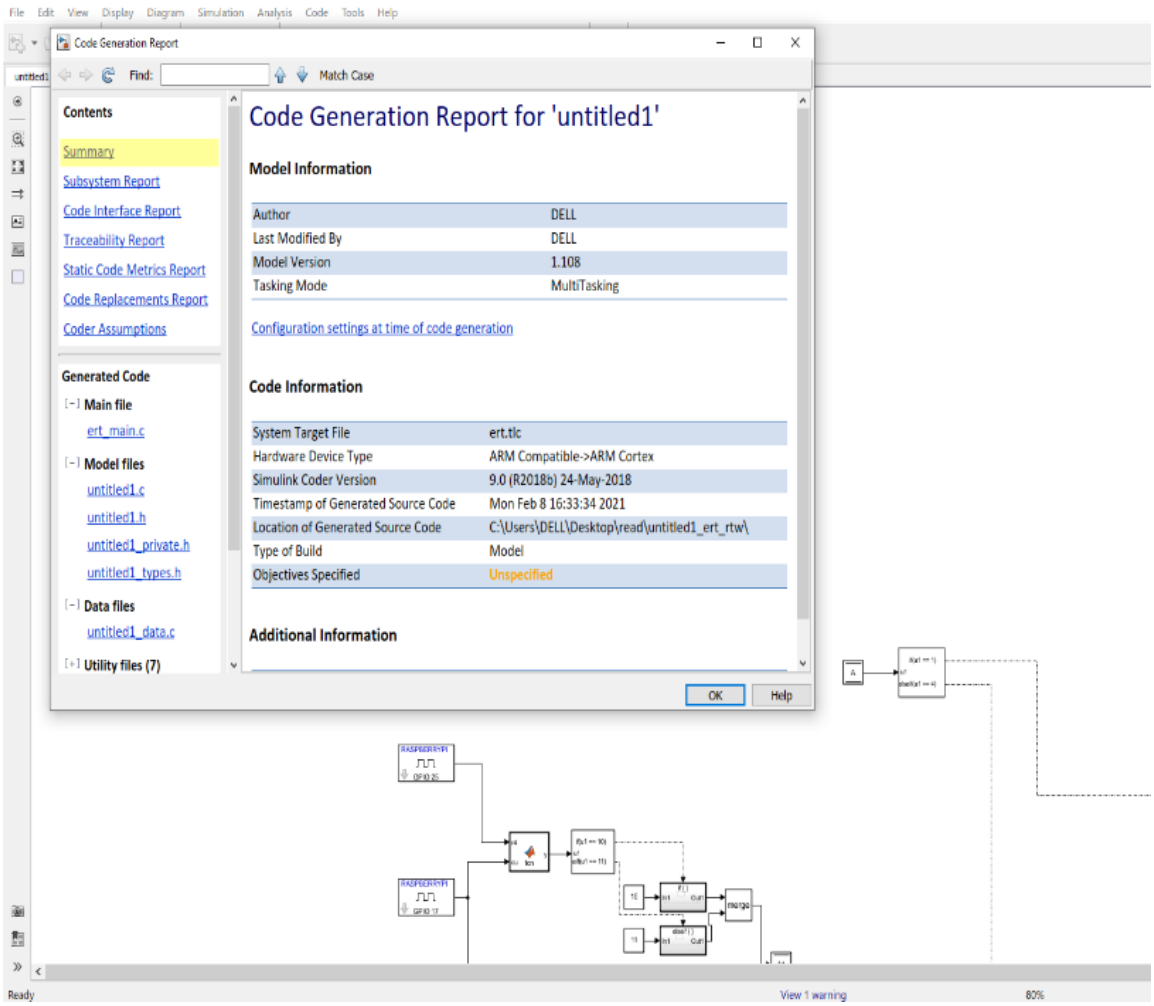
7 Select ‘External’ from drop down list as shown below.



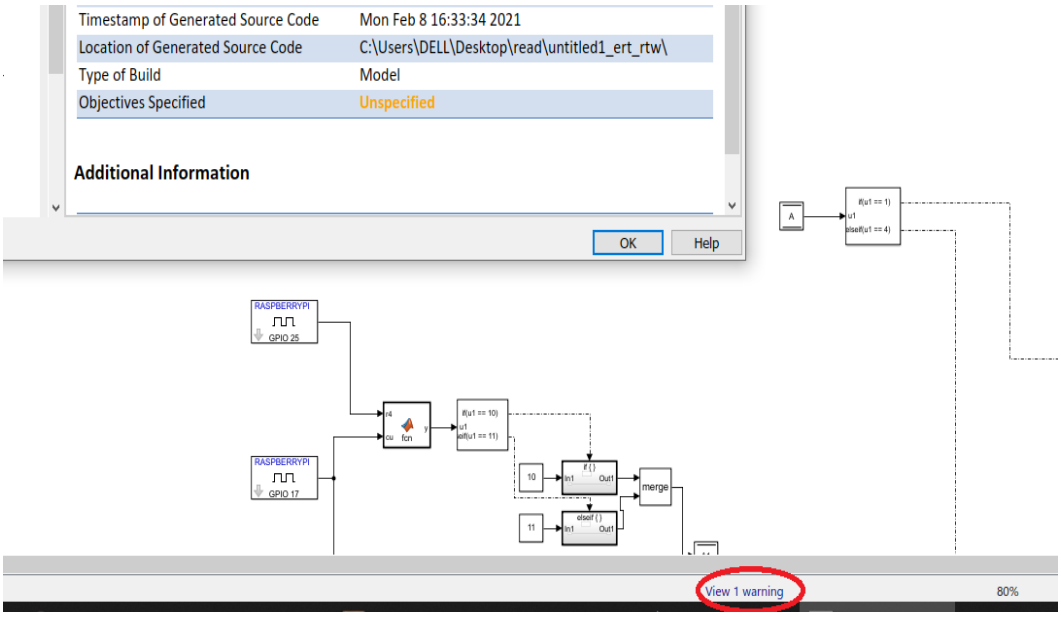
8 Select ‘Deploy to Hardware’ from drop down list as shown below.



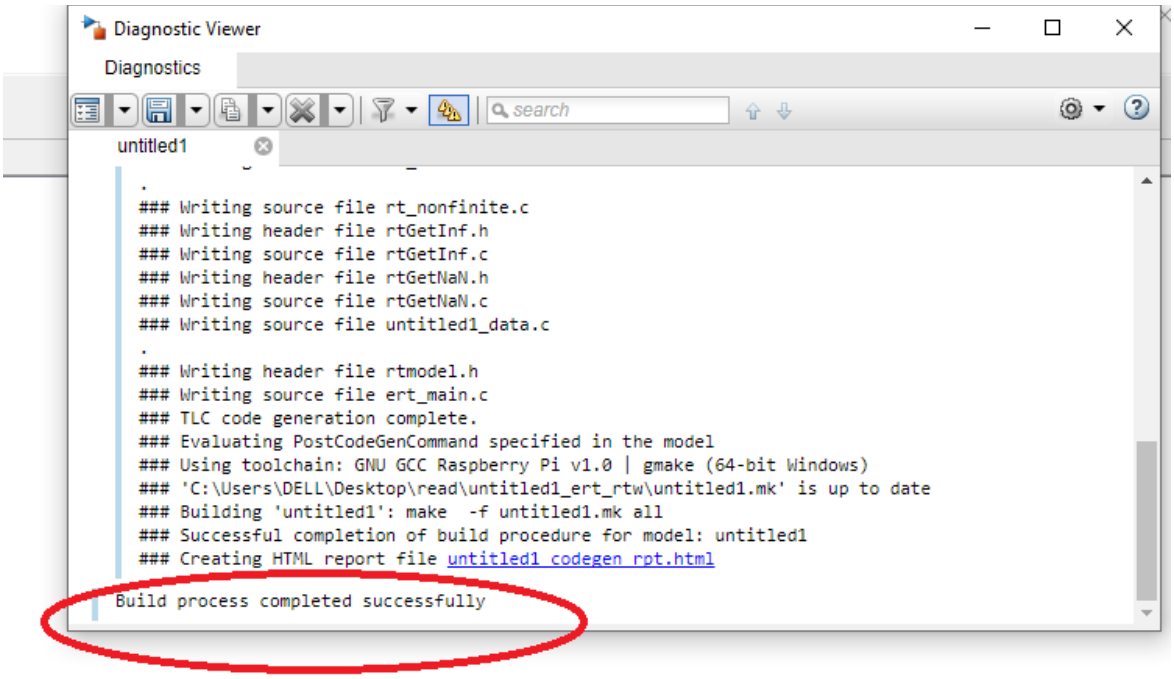
9 Once the program is deployed successfully on RPI, below tab will appear.



10 Click on ‘View 1 warning’ message present at the bottom of tab as shown below.



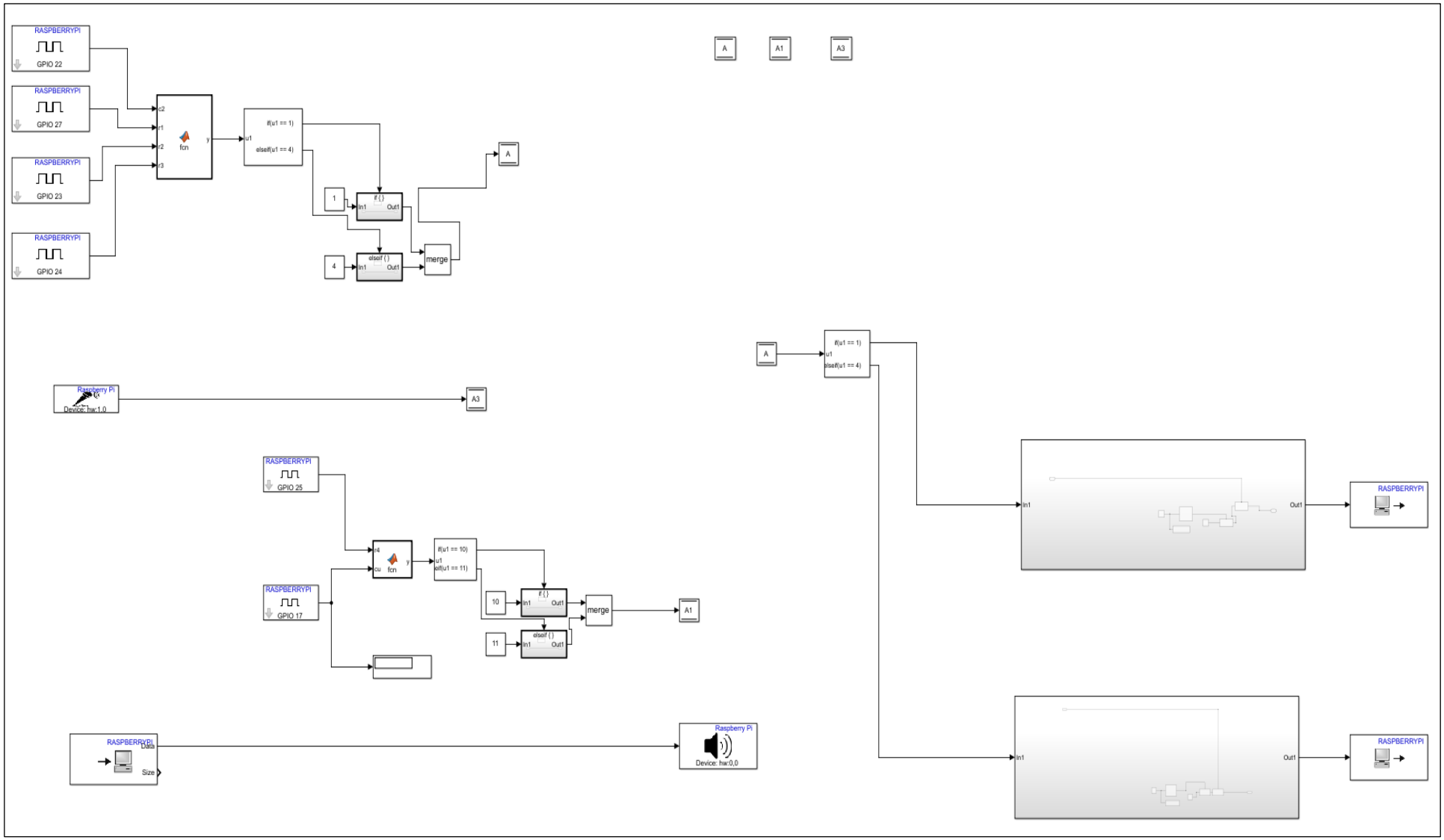
11 It will open ‘Diagnostic Viewer’ and in the diagnostic viewer, ‘Build process completed successfully’ shows the successful deployment of Simulink model to raspberry pi.



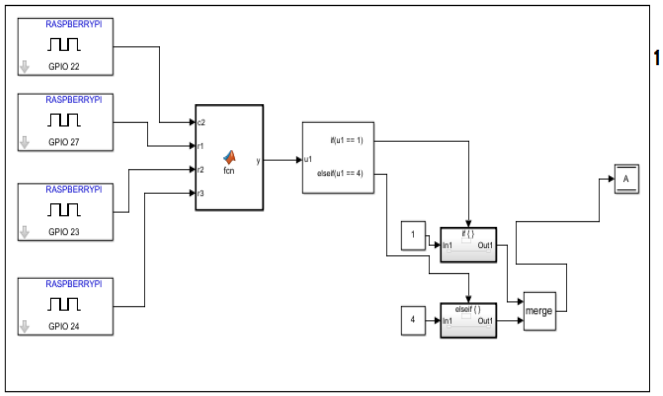
12 Restart the devices after uploading the code on each of them with mandatory changes discussed in further sections and start communication.

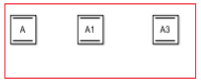
1.6 CODES FOR ONE-TO-ONE AND ONE-TO-MULTIPLE COMMUNICATION

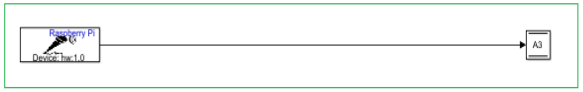
A. CODE FOR ONE-TO-ONE VOICE COMMUNICATION

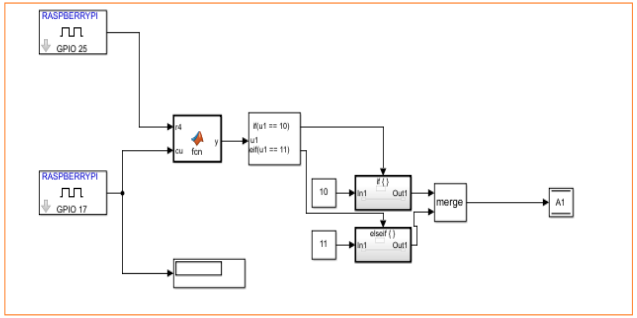


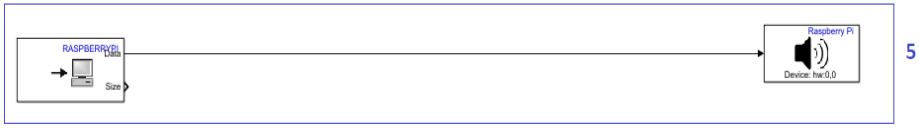
One-to-one voice communication Simulink model is divided into 6 sections as shown below. They are explained in further sections.

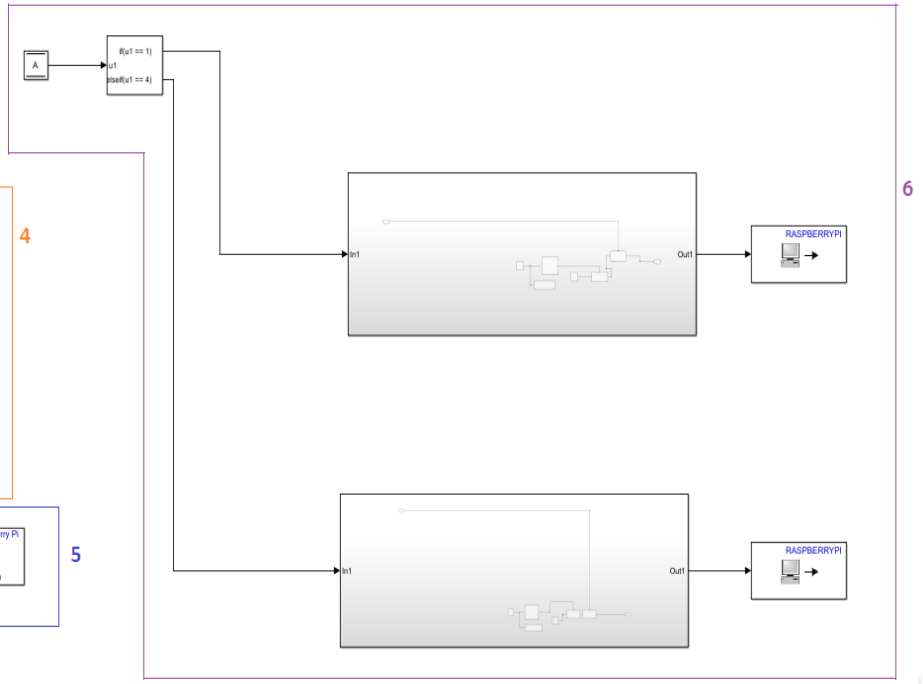




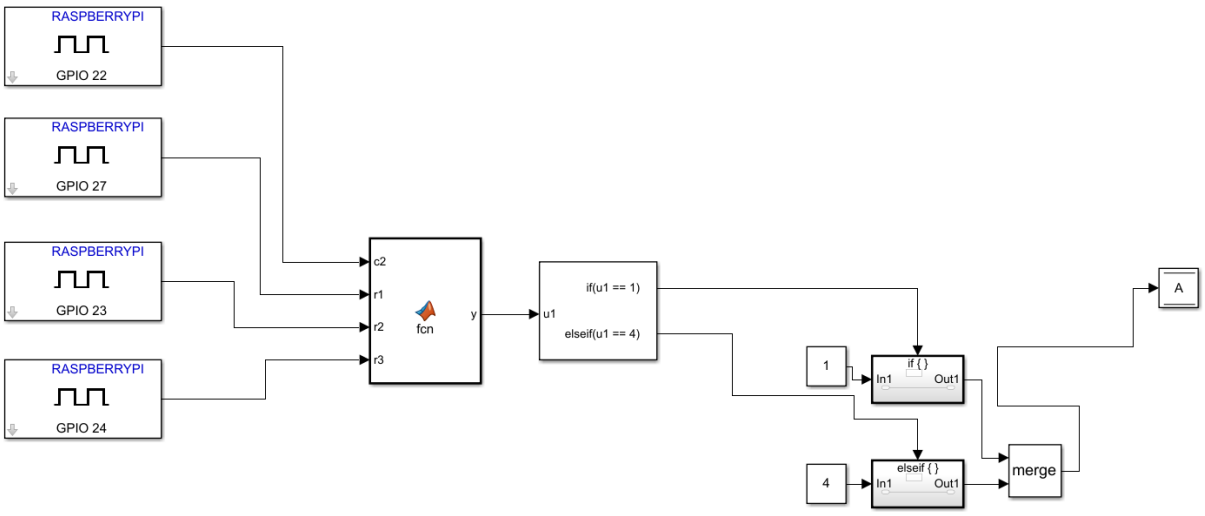








PART 1



This section consists of blocks for keypad connection with GPIO pins of RPI. The ‘fcn’ block consists of logic for identifying and storing the key value pressed on keypad.

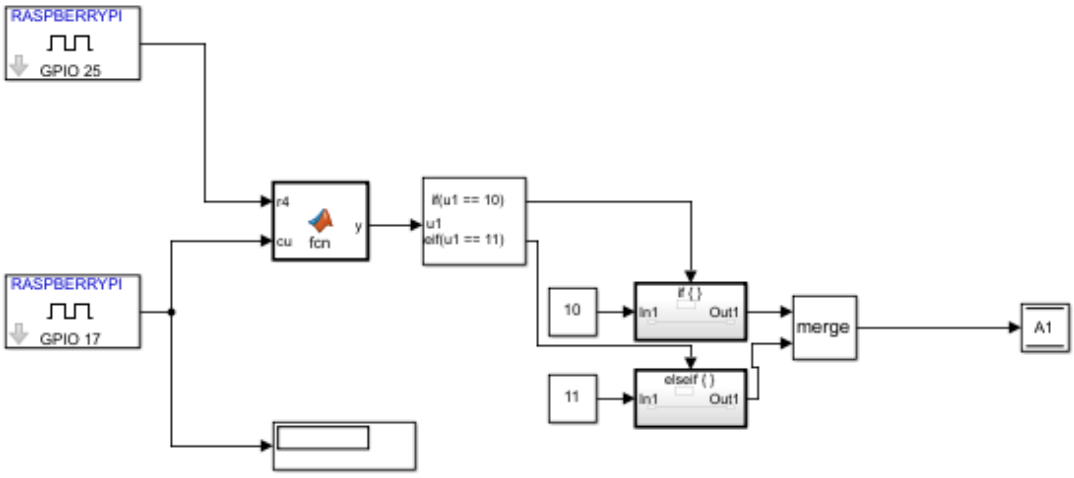
PART 2

This section consists of three Data Store Memory block which defines and initializes a named shared data store for three variables from code.

PART 3

This section consists of an ALSA Audio Mixer and a Data Store Memory block. The ALSA Audio Mixer block reads audio data from the audio input device connected to the RPI. The block uses the Advanced Linux Sound Architecture (ALSA) driver framework to read audio data. The audio data is stored in A3 audio block.

PART 4

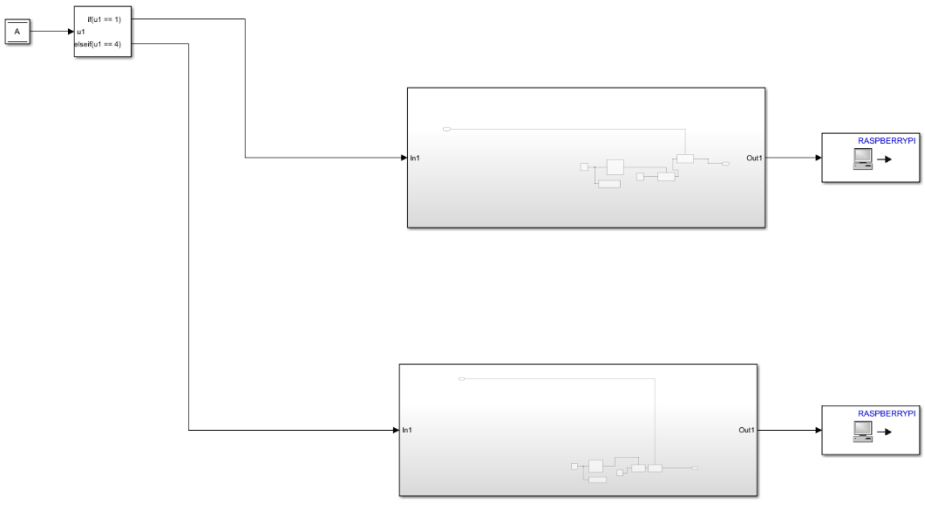


This section consists of blocks for keypad connection with GPIO pins of RPI for '*' key.

PART 5

This section consists of UDP Receive and an Audio Playback blocks. The UDP Receive block receives packet from another UDP hosts on a wireless network. Basically, this code segment is used to receive voice data sent from other devices and play it on speaker which connected to the RPI hardware.

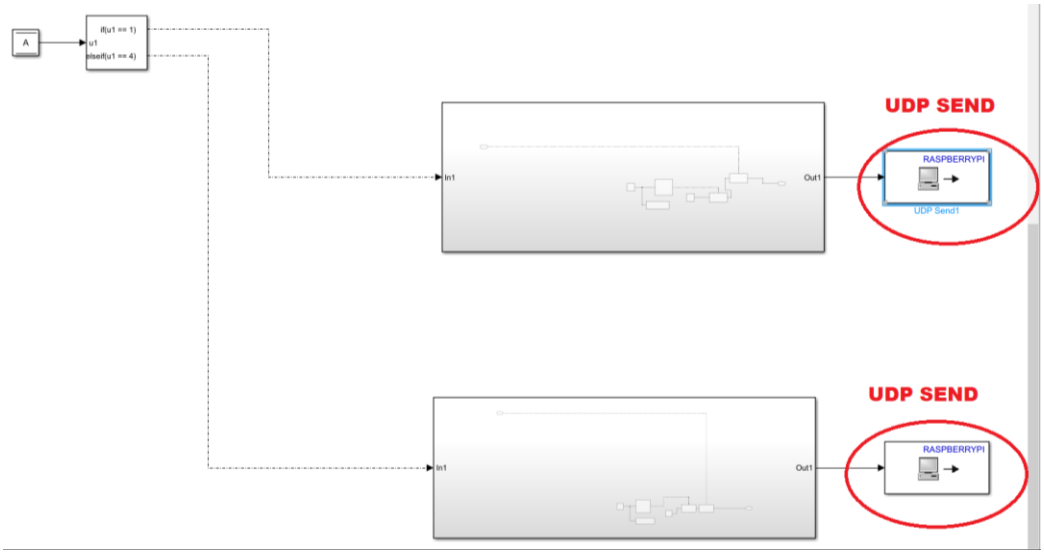
PART 6



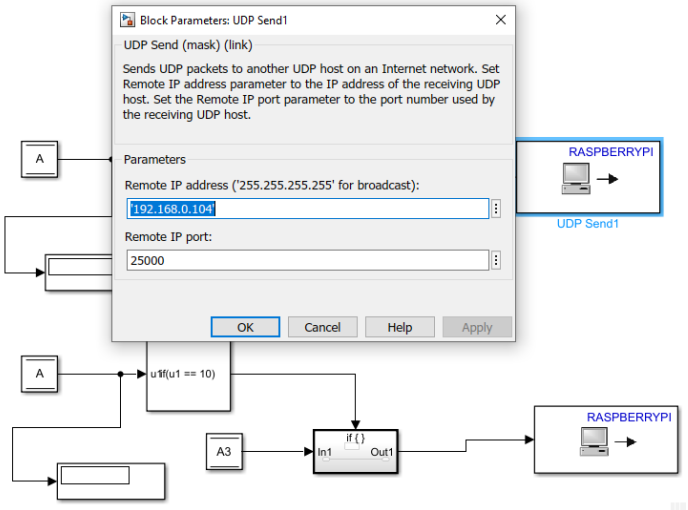
This section consists of logic to identify the device (IP address) at which data is to be sent. This block consists of ‘UDP Send’ blocks at end , which specifies the devices at which the data has to send through IP addresses. The UDP Send block sends UDP packets to a UDP host. The IP address and the port number of the receiving host are specified in the Remote IP address.

CHANGES REQUIRED IN ONE-TO-MANY COMMUNICATION CODE:

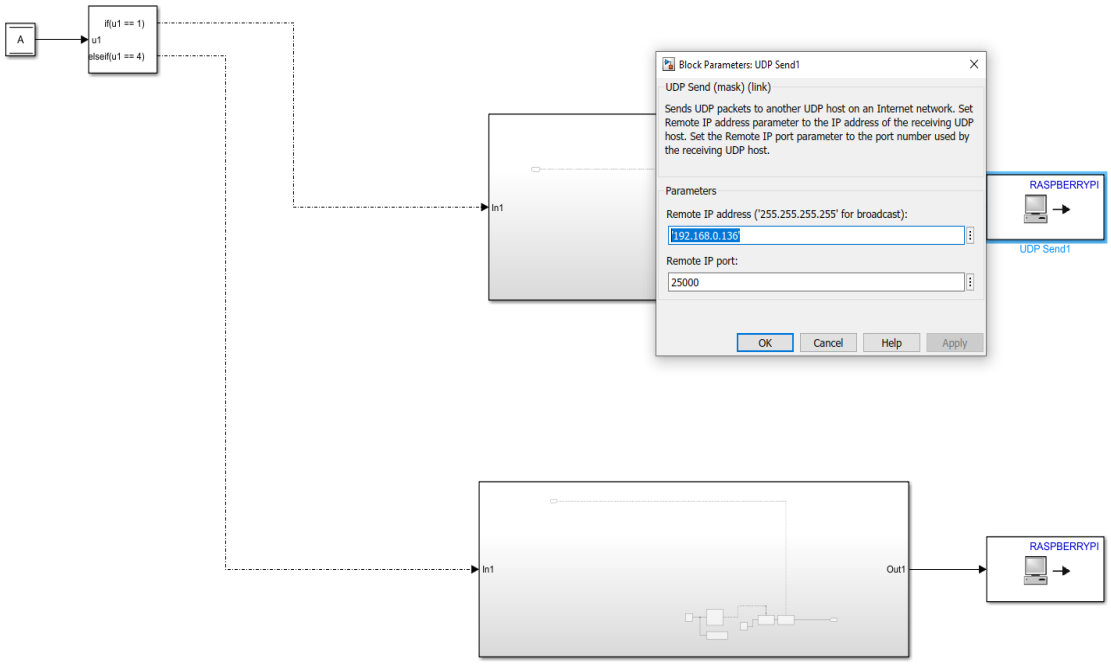
1. **PART 6:**
    - In part 6 of code, there are UDP send blocks encircled in the figure given below needs to be configured before installation to broadcasting RPI.



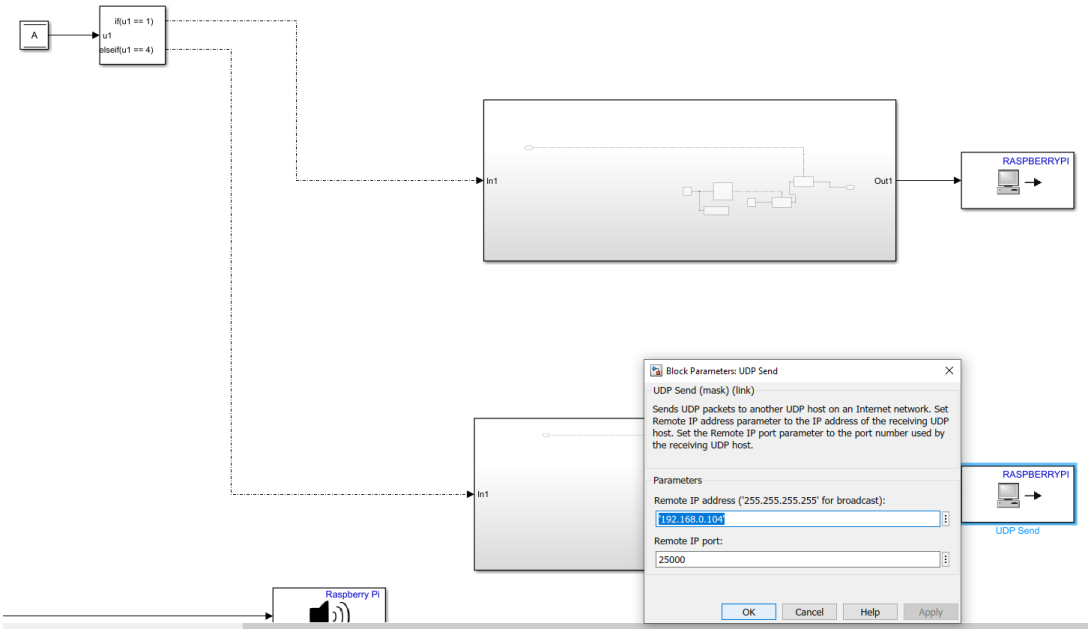
- Upon double clicking on UDP Send 1 block, block parameters tab appears as given below.



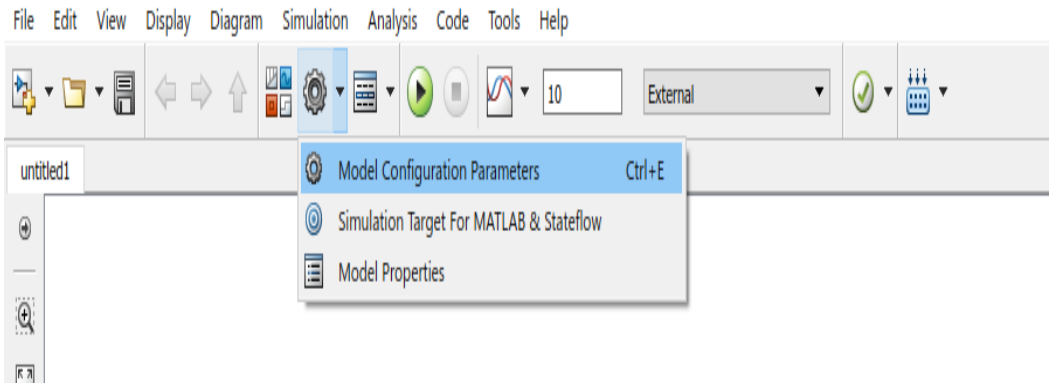
- Update ‘Remote IP address’ in UDP Send1 to IP address of device with which this raspberry pi is required to communicate. The devices should be connected to same wireless network.

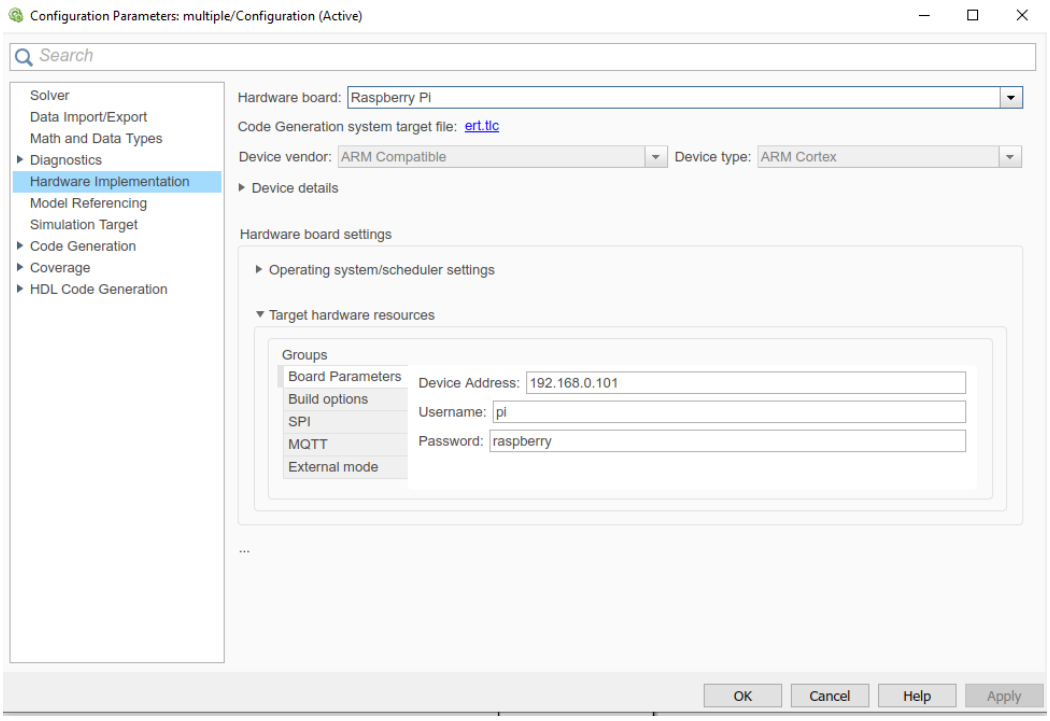


- Similarly, update ‘Remote IP address’ to IP address of other device with which this raspberry pi is required to communicate. The devices should be connected to same wireless network.



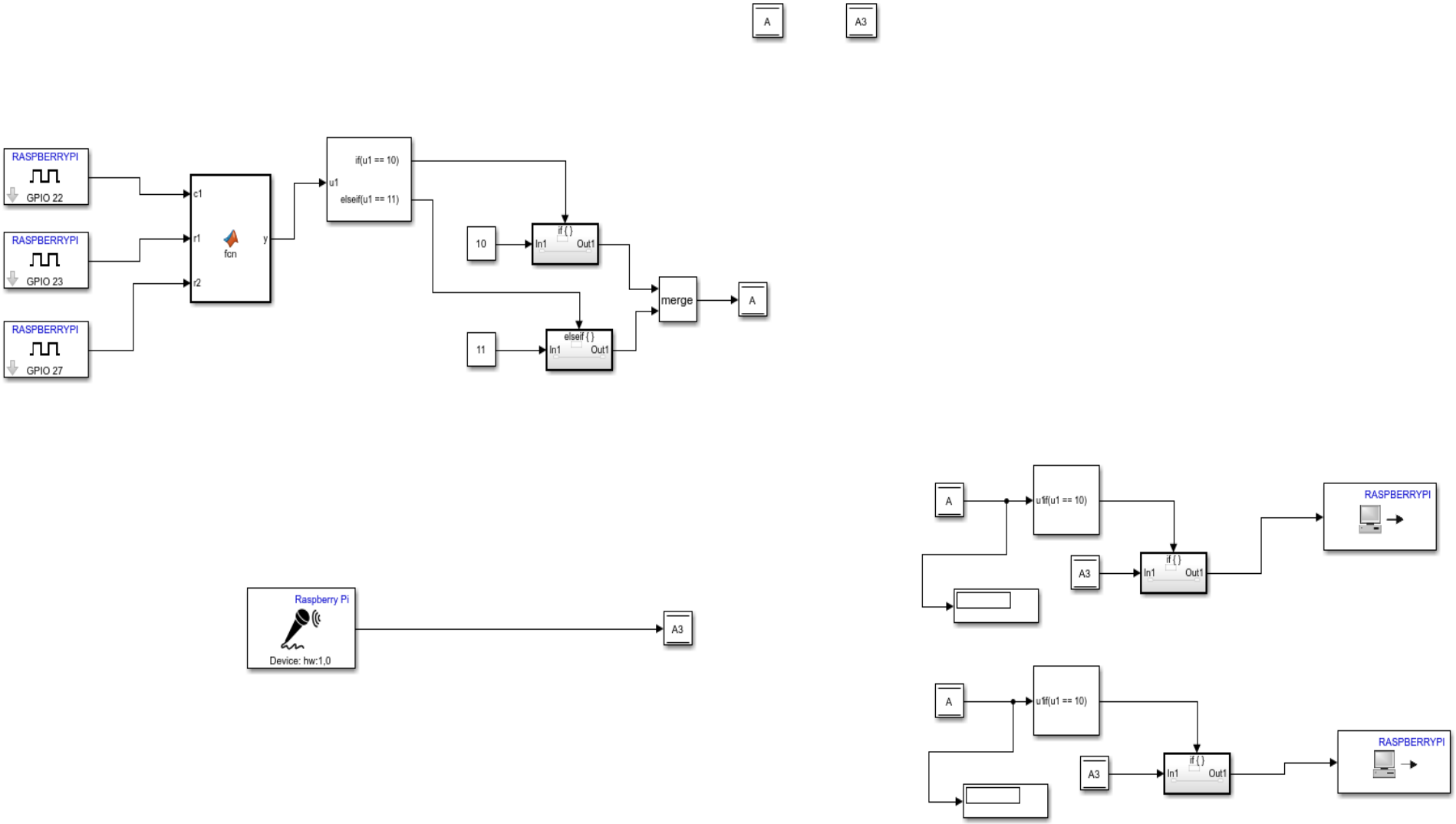
2. Update hardware details of Raspberry Pi at which the Simulink model will be deployed through . Go to Settings>Model configuration parameters>Hardware Implementation >Target hardware resources>Board parameters as shown below.





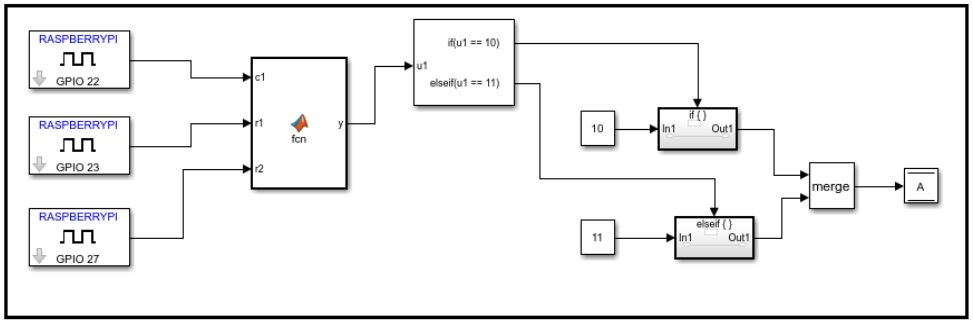
i. Add IP address of RPI on which code has to be executed in 'Device Address' field
ii. Add 'pi' in username as this field requires Username of RPI on which code has to be executed. (Default username of RPI is 'pi')
iii. Add 'raspberry' in password as this field requires Password of RPI on which code has to be executed. (Default password of RPI is 'raspberry')

B. CODE FOR ONE-TO-MANY VOICE COMMUNICATION



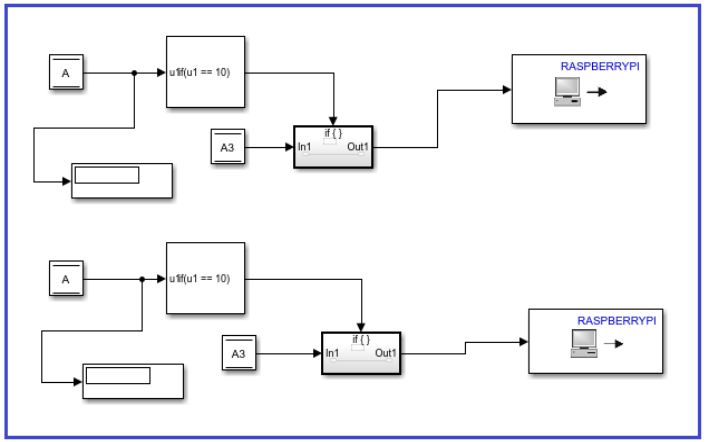
One-to-many voice communication Simulink model is divided into 4 sections as shown below. They are explained in further sections.

2



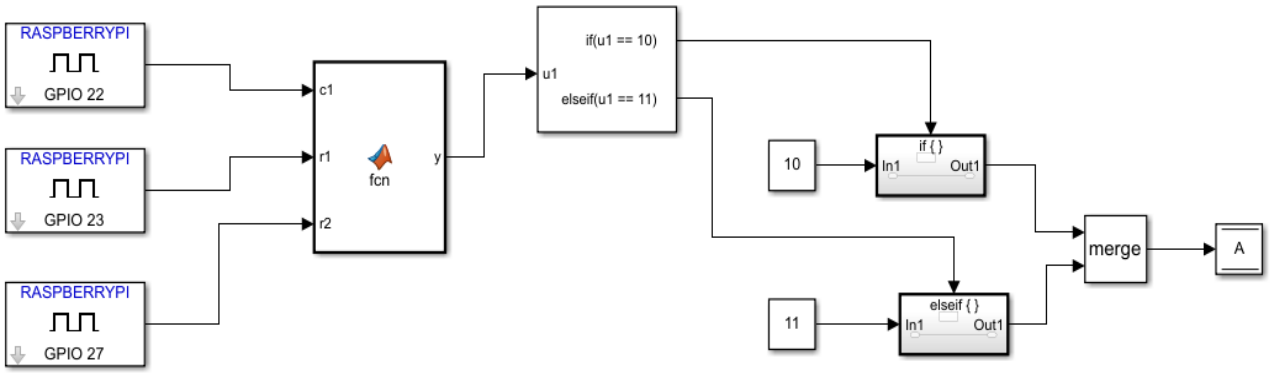
1

3



4

PART 1



This section consists of blocks for keypad connection with GPIO pins of RPI. The ‘fcn’ block consists of logic for identifying and storing the key value pressed on keypad.

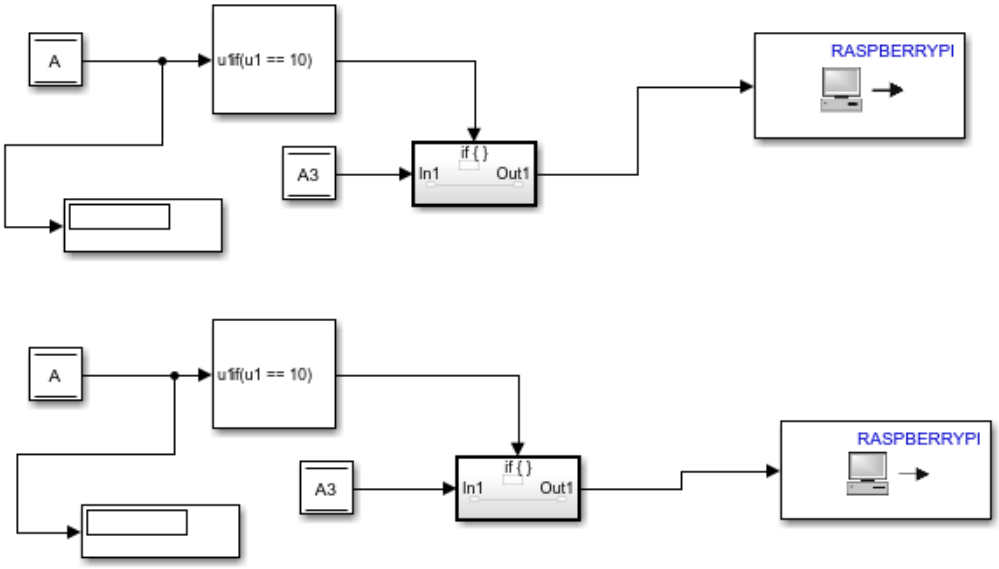
PART 2

This section consists of two Data Store Memory block which defines and initializes a named shared data store for two variables from code.

PART 3

This section consists of an ALSA Audio Mixer and a Data Store Memory block. The ALSA Audio Mixer block reads audio data from the audio input device connected to the RPI. The block uses the Advanced Linux Sound Architecture (ALSA) driver framework to read audio data. The audio data is stored in A3 audio block.

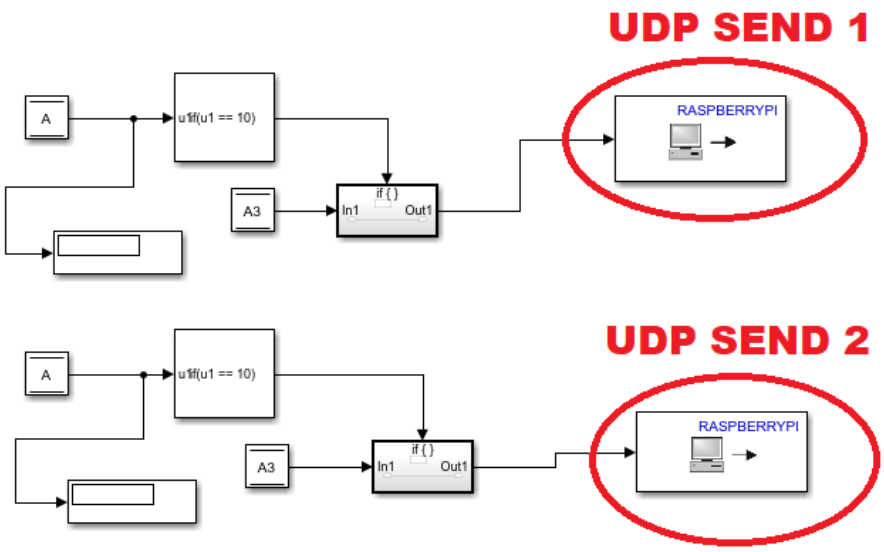
PART 4



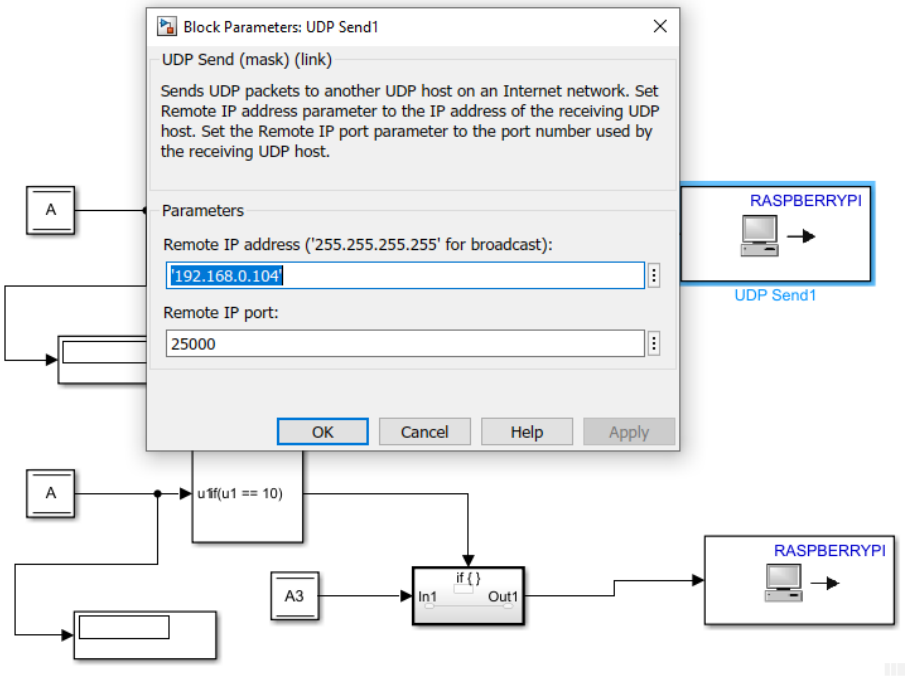
This section consists of logic to identify the device (IP address) at which data is to be sent. This block consists of 'UDP Send' blocks at end , which specifies the devices at which the data has to send through IP addresses. The UDP Send block sends UDP packets to a UDP host. The IP address and the port number of the receiving host are specified in the Remote IP address.

CHANGES REQUIRED IN ONE-TO-MANY COMMUNICATION CODE:

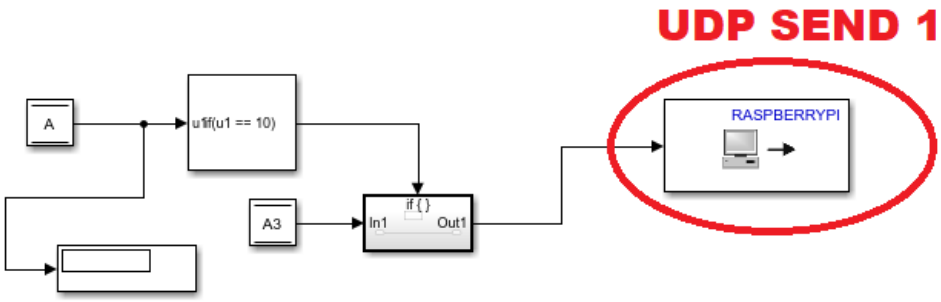
3. **PART 6:**
    - In part 6 of code, there are UDP send blocks encircled in the figure given below needs to be configured before installation to broadcasting RPI.



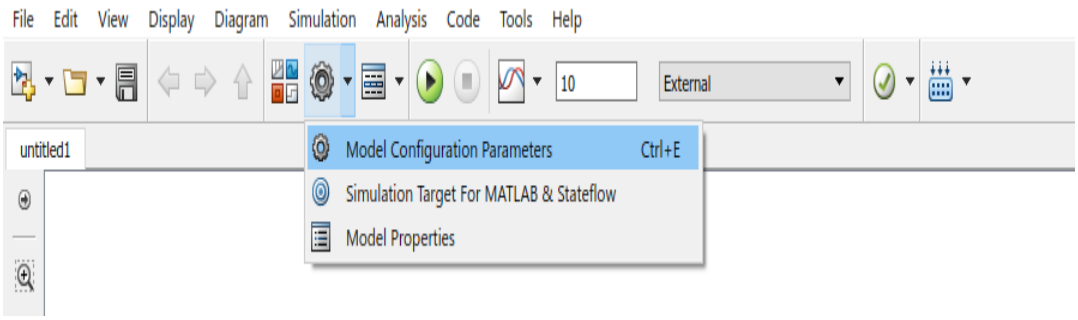
- Upon double clicking on UDP Send 1 block, block parameters tab appears as given below.

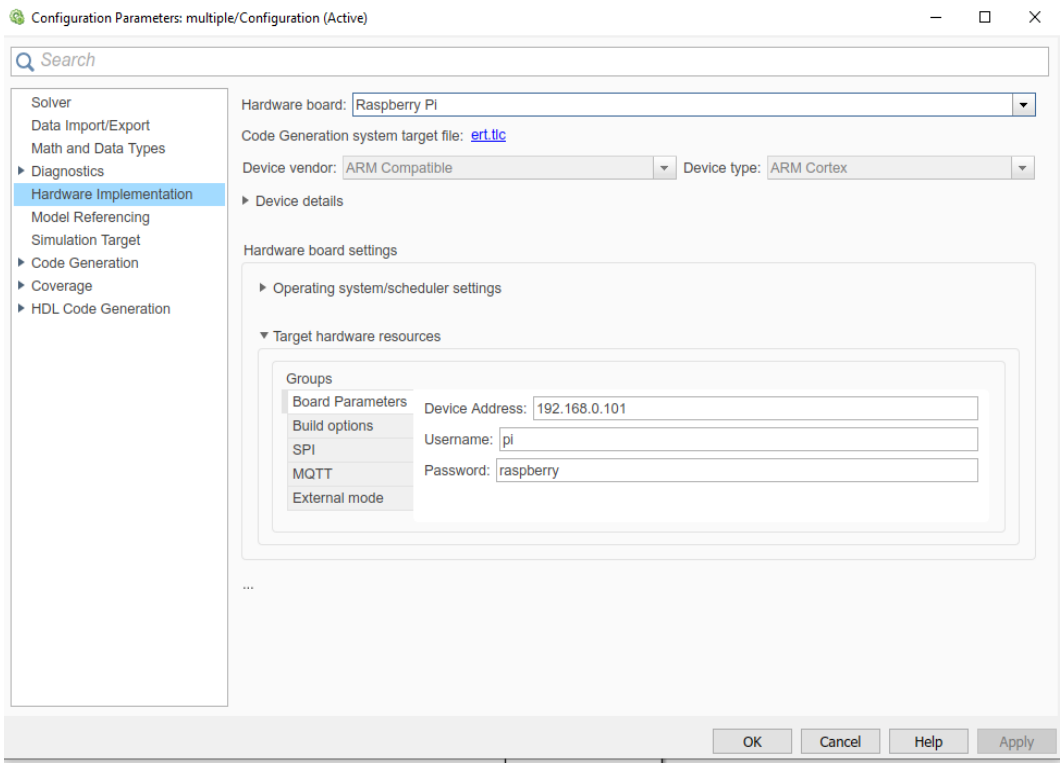


- Update ‘Remote IP address’ to IP address of device to which this raspberry pi is required to send voice message to be broadcasted. Similarly, ‘n’ number of UDP Send loops as shown below can be added with each UDP Send updated with different IP addresses of ‘n’ number of voice communication (RPI) devices should be added in this Simulink model. The devices should be connected to same wireless network.



4. Update hardware details of Raspberry Pi at which the Simulink model will be deployed through . Go to Settings>Model configuration parameters>Hardware Implementation >Target hardware resources>Board parameters as shown below.





iv. Add IP address of RPI on which code has to be executed in ‘Device Address’ field

v. Add ‘pi’ in username as this field requires Username of RPI on which code has to be executed. (Default username of RPI is ‘pi’)

vi. Add ‘raspberry’ in password as this field requires Password of RPI on which code has to be executed. (Default password of RPI is ‘raspberry’)