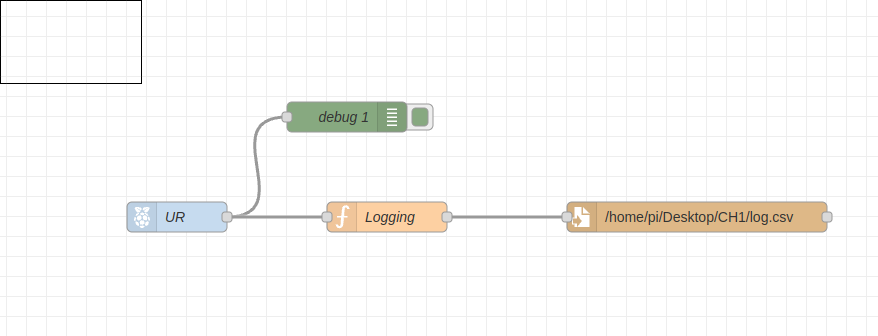
**Challenging Task-1 IOT Domain Analyst**

21MIS1128

T.Akash Reddy

1) An IoT system is implemented using an **ultrasonic sensor** connected to a Raspberry Pi to monitor object distances (e.g., vehicle parking, obstacle detection). The system uses **Node-RED** to process the sensor data and generate a log file for every 2 seconds of interval containing distance measurements with vehicle parking status. This log file can be used for analysis or reporting.

Node-red flow:



**function node code to save timestamp and sensor data**

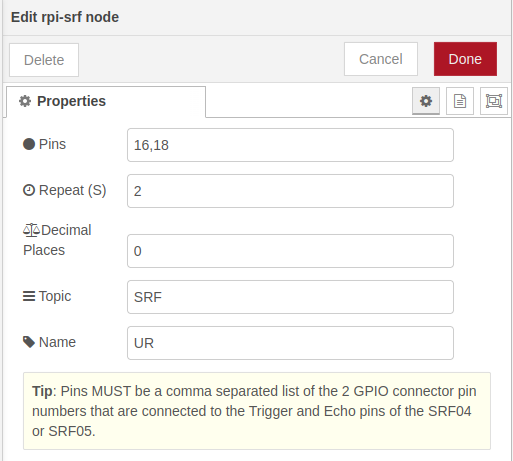
const d = new Date();

let time = d.getTime();

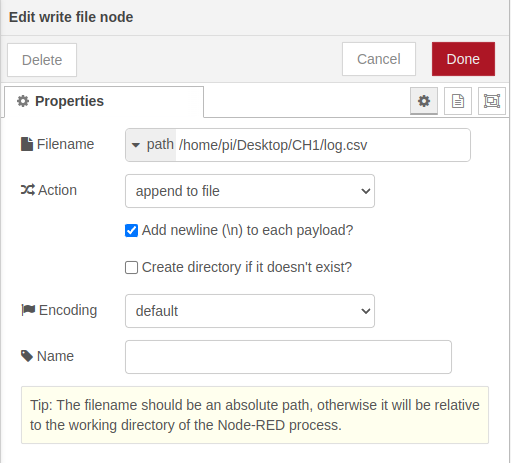
var msg=msg.payload;

var m = {'Time':time,'distance':msg};

return {payload:m};

here we

have to specify the pin numbers of trigger and echo pins of UR sensor.



This is csv node in node-red here we have to specify the file path to store sensor data.

This is the csv file whic contains sensor timestamp and sensor values.

Figure 1: This is the csv file with sesnor timestamp and sensor value

Here bigger number indicates that there is no motion near sensor but if it smaller reading that means there is some motion.

Json code:

[

{

"id": "9b21ef47070cbf87",

"type": "tab",

"label": "Flow 1",

"disabled": false,

"info": "",

"env": []

},

{

"id": "745526592b9c499b",

"type": "file",

"z": "9b21ef47070cbf87",

"name": "",

"filename": "/home/pi/Desktop/CH1/log.csv",

"filenameType": "str",

"appendNewline": true,

"createDir": false,

"overwriteFile": "false",

"encoding": "none",

"x": 850,

"y": 320,

"wires": [

[]

]

},

{

"id": "e6340243a48acc67",

"type": "debug",

"z": "9b21ef47070cbf87",

"name": "debug 1",

"active": true,

"tosidebar": true,

"console": false,

"tostatus": false,

"complete": "payload",

"targetType": "msg",

"statusVal": "",

"statusType": "auto",

"x": 500,

"y": 220,

"wires": []

},

{

"id": "c856d6077e072f9b",

"type": "rpi-srf",

"z": "9b21ef47070cbf87",

"name": "UR",

"topic": "SRF",

"pulse": "2",

"pins": "16,18",

"precision": "0",

"x": 330,

"y": 320,

"wires": [

[

"e6340243a48acc67",

"2c037e5de86d7067"

]

]

},

{

"id": "2c037e5de86d7067",

"type": "function",

"z": "9b21ef47070cbf87",

"name": "Logging",

"func": "\nconst d = new Date();\nlet time = d.getTime();\nvar msg=msg.payload;\nvar m = {'Time':time,'distance':msg};\nreturn {payload:m};",

"outputs": 1,

"timeout": 0,

"noerr": 0,

"initialize": "",

"finalize": "",

"libs": [],

"x": 540,

"y": 320,

"wires": [

[

"745526592b9c499b"

]

]

}

]