**Automated stock/level monitoring**

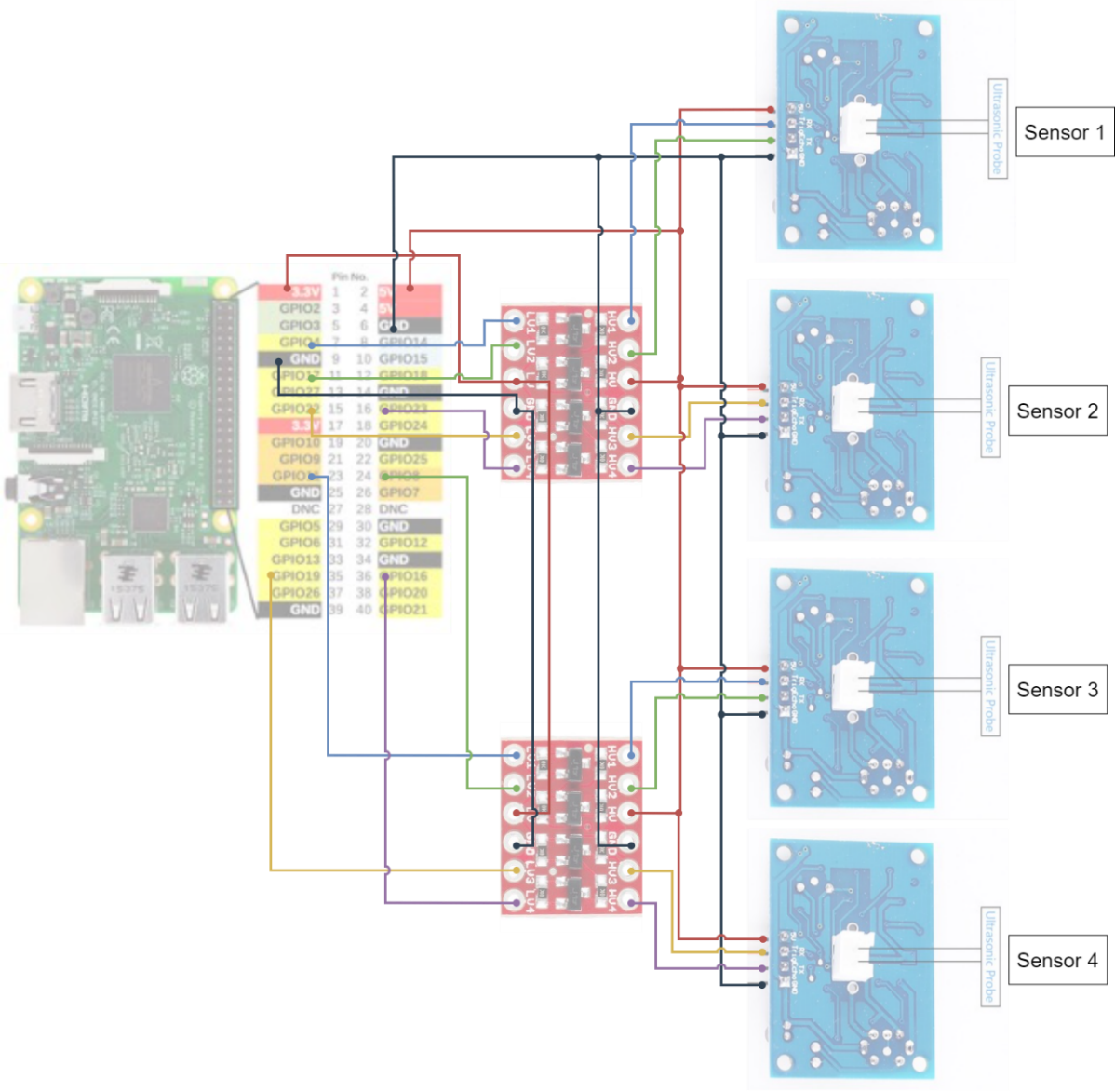
**Hardware:**

1. JSN-SR04T Waterproof Ultrasonic Distance Sensor (depends on the number of sensors to be installed)
2. Bi-Directional Logic Level Converter (depends on the number of sensors to be installed)
3. Pin header (optional)
4. Raspberry Pi 4 + microSD card
5. Jumper wires
6. Soldering tools

Installation suggestion:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Raspberry Pi | Bi-Directional Logic Level Converter | JSN-SR04T Waterproof Ultrasonic Distance Sensor |
| Quantity | 1 | 1 | 1 |
| 2 |
| 2 | 3 |
| 4 |

**Circuit diagram:**

****

**Software:**

1. Clone the code from GitHub repository: Open a Terminal (Click the ‘*Terminal*’ icon in the taskbar) and type in the following. Wait until the cloning process completes.

|  |
| --- |
| git clone https://github.com/jyl12/level-monitoring.git |

Or unzip the folder to the following path (Click the ‘*File Manager*’ icon in the taskbar)

|  |
| --- |
| /home/pi |

1. Install necessary software libraries: Type the following in the Terminal (continue from Step 1 or start a new Terminal). Wait until the installation process completes. Type *Y* for yes and continue if prompted.

|  |
| --- |
| cd level-monitoring |
| sh install.sh |

1. Run the code: Type the following in the Terminal to start.

|  |
| --- |
| python3 /userInterface/app.py |

If it is a new Terminal, type the following.

|  |
| --- |
| python3 /level-monitoring/userInterface/app.py |

Once it starts, the web interface should be accessible at <http://localhost:5000> or from another device at http://<raspberrypi-IP>:5000 (replaces <raspberrypi-IP> with the IP address of the Raspberry Pi in the network). All you need to find out the IP address of your Raspberry Pi is to run either one of the following commands in the terminal.

|  |
| --- |
| hostname -I |
| ifconfig |

**Email (optional):**

In *sendemail.py*, please modify *#Email Variables* accordingly to enable the email function.

|  |
| --- |
| #Email Variables  SMTP\_SERVER = 'smtp.gmail.com' #Email Server (don't change!)  SMTP\_PORT = 587 #Server Port (don't change!)  GMAIL\_USERNAME = 'shoestringtestzy@gmail.com' #change this to match your gmail account  GMAIL\_PASSWORD = 'Pass1234!' #change this to match your gmail password |

In *analysis.py*, set *ENABLE\_EMAIL = 1* to enable the email and change *USER\_EMAIL* to relevant emails.

|  |
| --- |
| ENABLE\_EMAIL = 0 #1 to enable email sending  USER\_EMAIL = ['jyl49@cam.ac.uk'] # ['abc@xyz.com','efg@xyz.com'] |

**Run Program on Startup (Optional):**

1. Create a .desktop File. Open a Terminal and execute the following commands to create an *autostart* directory (if one does not already exist) and edit a .desktop file.

|  |
| --- |
| mkdir /home/pi/.config/autostart |
| nano /home/pi/.config/autostart/startapp.desktop |

1. Copy in the following text into the *startapp.desktop* file. Feel free to change the *Name* and *Exec* variables to your particular application.

|  |
| --- |
| [Desktop Entry]  Type=Application  Name=App  Exec=@/usr/bin/python3 /home/pi/level-monitoring/userInterface/app.py |

1. Save and exit with *ctrl + x*, followed by *y* when prompted to save, and then *enter*. Reboot with the following command.

|  |
| --- |
| sudo reboot |

Please refer Method 2:autostart of the following hyperlink for detailed explanation.

<https://learn.sparkfun.com/tutorials/how-to-run-a-raspberry-pi-program-on-startup/all#method-2-autostart>

**Troubleshoot:**

|  |
| --- |
| 1. OPC UA port: 4840. When connecting UaExpert to Raspberry Pi, the following logs might appear in the terminal console of Raspberry Pi and UaExpert.     msg uaexpert |
| Solution: Can be ignored. Error message is caused by missing PublishResult.Results acknowledgements on freeopcua. Please refer the following links for more information.  <https://github.com/FreeOpcUa/opcua-asyncio/issues/438>  <https://github.com/FreeOpcUa/python-opcua/issues/803> |