

A picture containing food

Description automatically generated

**Decentralized Cross Robot Control Safety features**

Report of the Project DCRC at minor Adaptive robotics, Fontys engineering

Author : J. van Wolfswinkel, Hessel van Gemert

Student number : 3497941, 3604985

Minor : Adaptive Robotics

Project : Decentralized Cross Robot Control

Teacher Safety : Esmaeil Najafi

Date : 04/07/2021  
Version : 0003

General information

**Project:**

Project name : Decentralized Robot Control Safety features   
Part of : Minor Adaptive Robotics, DCRC, Fontys Engineering

**Information student:**

Name : Joost van Wolfswinkel

Study : Fontys Engineering Mechatronics

Student number : 3497941

Email : j.vanwolfswinkel@student.fontys.nl

**Information student:**

Name : Hessel van Gemert

Study : Fontys Engineering Mechatronics

Student number : 3604985

Email : h.vangemert@student.fontys.nl

Version management

Please update this table when changing or adding something to this report.

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Name | Description |
| 0001 | 02/07/2021 | Joost van Wolfswinkel | First draft |
| 0002 | 04/07/2021 | Joost van Wolfswinkel | Added video description |
| 0003 | 04/07/2021 | Hessel van Gemert | Final revision |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Index

[List of figures 3](#_Toc76334883)

[List of tables 3](#_Toc76334884)

[1. Introduction 4](#_Toc76334885)

[2. Assignment overview 5](#_Toc76334886)

[3. Possibilities of group reactions to hazards 6](#_Toc76334887)

[3.1. Type of hazards 6](#_Toc76334888)

[3.2. Solutions to hazards 7](#_Toc76334889)

[3.3. Connecting hazard with solutions 8](#_Toc76334890)

[4. Implementing features in to CoLAB 10](#_Toc76334891)

[4.1. Choice fire sensor 10](#_Toc76334892)

[4.2. Connection to the CoLAB 11](#_Toc76334893)

[4.3. The program 12](#_Toc76334894)

[5. Testing features 13](#_Toc76334895)

[6. Recommendation 14](#_Toc76334896)

[7. Attachments 0](#_Toc76334897)

[I. Attachment pinout sensor and raspberry pi 0](#_Toc76334898)

[II. Attachment flowchart CoLAB 1](#_Toc76334899)

[III. Attachment flowchart Robot 3](#_Toc76334900)

# List of figures

[Figure 1 Car with UWB tags and CoLAB 4](#_Toc76329450)

[Figure 2 Smoke and gas sensor on CoLAB 11](#_Toc76329451)

# List of tables

[Table 1 Development hazard reactions 7](#_Toc76329461)

[Table 2 fire reactions 8](#_Toc76329462)

[Table 3 Ambulance / Police emergency reactions 8](#_Toc76329463)

[Table 4 Bad weather reactions 8](#_Toc76329464)

[Table 5 Building power failure reactions 9](#_Toc76329465)

[Table 6 Obstructed paths reactions 9](#_Toc76329466)

[Table 7 Gas leakage reactions 9](#_Toc76329467)

[Table 8 Earthquake reactions 9](#_Toc76329468)

[Table 9 Fire detection sensors 10](#_Toc76329469)

# Introduction

This report is written to document the safety assignment for the ‘Decentralized Cross Robot Control’ project, also called DCRC, of the minor adaptive robotics. The goal of the DCRC project is to develop a device that is called ‘CoLAB’. The CoLAB should be a box that can connect to every autonomous driving robot and give and take information from them. With that information the CoLAB is used to tell the robot where to go and what to do. The CoLAB is also capable to communicate in a bigger network of other CoLAB’s, to divide information and tasks given by a main server.

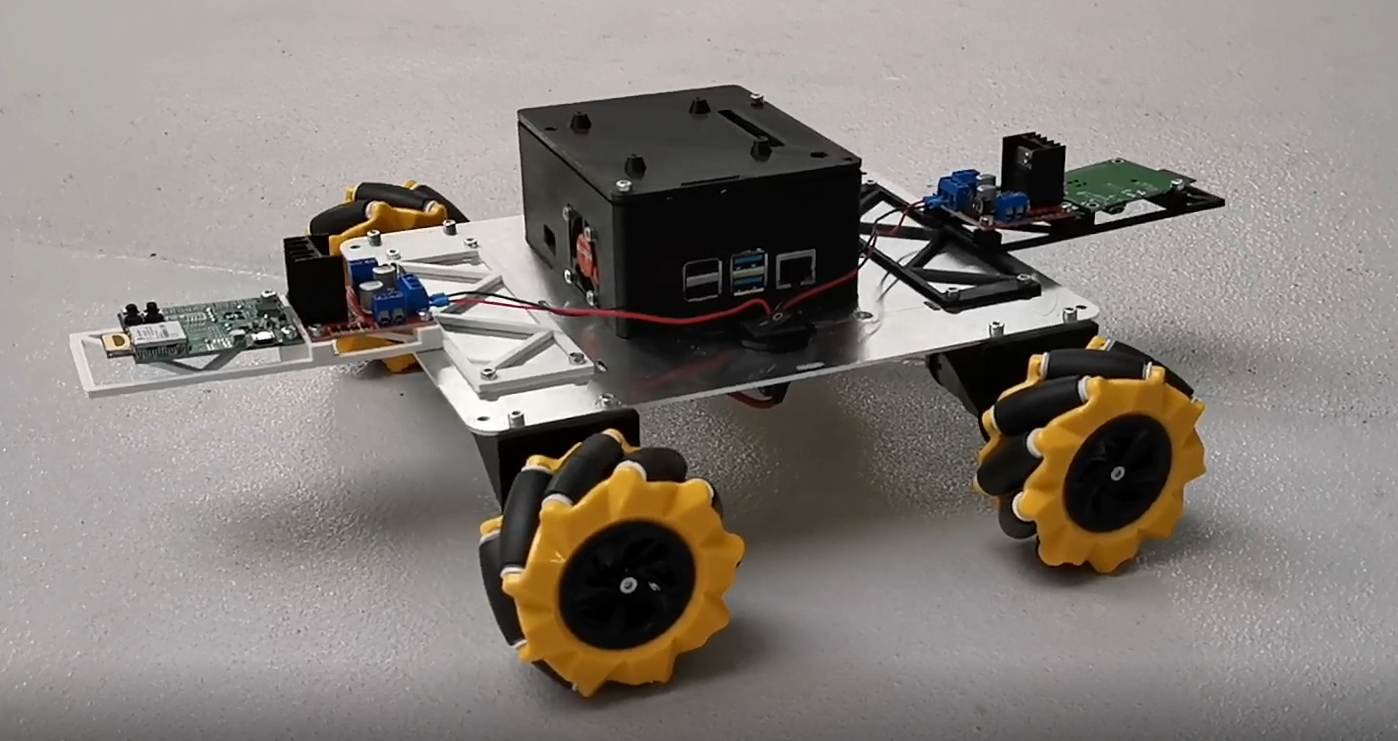


Figure 1 Car with UWB tags and CoLAB

With a standard communication by CoLAB between all driving robots, it is possible to create a group reaction to hazards. This can improve the safety of a robotic environment dramatically. With these features you can not only make a safer environment but also prevent any extra cost in damage. To start building those features this report investigated the possibilities of the group reaction for all the kinds of hazards. The reaction to fire was then chosen to be the first feature to be implemented in to the CoLAB.

In chapter 2 an overview of the safety assignment can be found, chapter 3 consists of an analysation of the different hazards and possible reactions to those hazards. After that in chapter 4 it is explained how these reactions are implemented in the CoLAB and in chapter 5 the results from the tests of those implementations are found. Finally, in chapter 6 there are some recommendations for possible future projects.

# Assignment overview

The goal of this assignment is to Implement new safety features into the robots of the DCRC project. In this assignment there will be focussed on the reaction of the robots when a fire is located. This is done with the following steps:

* Sensor on the robot for detection
* Communication to other robots in the system, by using the CoLAB Ultra-wide band communication device (UWB).
* Multiple ways of reaction to the fire, so the client can choose what is the best stagey for their application.
* Implementing the program
* Testing the features

All these steps will be written down in this safety report to accomplish level 4 of safety in the minor. This report will also help further DCRC groups for implementing more features to different emergencies.

# Possibilities of group reactions to hazards

This chapter first looks at all the possible hazards in the CoLAB environment. And will secondly describe how to react as a group in these situations.

## Type of hazards

There are many environments that a CoLAB can be used in like a warehouse, an airport or in a factory. These places are most likely inside, but if we look to the future, it is possible that the CoLAB will also be used outside in places like farming areas or outside storage. All these environments have mostly the same hazards but there can be some exceptions. Listed below all hazards for the robot are explained:

* **Fire**

There is fire located in the building or one of the robots is on fire.

* **Ambulance / Police emergency**

The emergency services are coming.

* **Bad weather**

Too much wind, rain or even lighting can make the robots take damage or lose control.

* **Building power failure**

All systems in the building are down due to power failure, except the self-powering robots.

* **obstructed paths**

Paths in the environment are damaged and robots are getting damage or getting stuck.

* **Gas leakage**

A gas leak is detected in the environment.

* **Earthquake**

An earthquake can make the robots take damage or lose control.

## Solutions to hazards

In previous paragraph all possible hazards where placed. Those hazards can have a safety solution with the group of robots. A brainstorm session was done to find all these possible solutions. These solutions were then marked with a label, this label shows how hard it is to develop this solution. In the Table 1 Development hazard reactions you can find all the solutions and their label. Note that the solutions are only for group reactions not for a single robot reaction.

Table 1 Development hazard reactions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr. | Description | Good to develop | Hard to develop | Barely  possible |
| 1 | Stand still | X |  |  |
| 2 | Clear walking path | X |  |  |
| 3 | Drive to a save location | X |  |  |
| 4 | Only work inside | X |  |  |
| 5 | Avoid specific area | X |  |  |
| 6 | Guide people to exit |  | X |  |
| 7 | Light up the environment | X |  |  |
| 8 | Guide emergency service to location where help is needed |  | X |  |
| 9 | Uses robots as battery | X |  |  |
| 10 | Clear specific area | X |  |  |
| 11 | Drive hospital beds to safety |  | X |  |
| 12 | Guide animals away from fire |  |  | X |
| 13 | Open shed doors in case of fire |  | X |  |
| 14 | Avoid a certain robot | X |  |  |

## Connecting hazard with solutions

Most hazards can have multiple solutions. This also depends on environment of the client. For instance, an automated warehouse without humans, can have a different reaction to fire then a warehouse with humans. This is because the safety of to humans comes first, and the safety of the robot comes second. In the following tables all the connections between hazards and solutions are made and explained.

* **Fire**

Table 2 fire reactions

|  |  |  |
| --- | --- | --- |
| Nr. | Solutions | Explanations |
| 1 | Stand still | Do nothing to prevent driving in the way of people |
| 2 | Clear walking path | Make it easier for people to go to the exit |
| 3 | Drive to a save location | When not on fire drive to a save location to save the robot |
| 5 | Avoid specific area | In combination with Nr. 3, avoid fire |
| 6 | Guide people to exit | Help people to find the exit trough smoke with indication or driving with them |
| 7 | Light up the environment | Give extra vision to people when there is smoke |
| 11 | Drive hospital beds to safety | Help people that can’t walk by driving their beds away from fire |
| 12 | Guide animals away from fire | Help animals to find the right way out by forming a wall for instance |
| 13 | Open shed doors in case of fire | Let animals out when there is fire |
| 14 | Avoid a certain robot | If a robot is on fire, avoid this robot |

* **Ambulance / Police emergency**

Table 3 Ambulance / Police emergency reactions

|  |  |  |
| --- | --- | --- |
| Nr. | Solutions | Explanations |
| 1 | Stand still | Do nothing to prevent driving in the way of people |
| 2 | Clear walking path | Make it easier for people to go trough |
| 8 | Guide emergency service to location where help is needed | Indicate or drive where to go |
| 10 | Clear specific area | Make space in a certain area |

* **Bad weather**

Table 4 Bad weather reactions

|  |  |  |
| --- | --- | --- |
| Nr. | Solutions | Explanations |
| 1 | Stand still | Shut down till weather is over |
| 3 | Drive to a save location | Drive to a shelter |
| 4 | Only work inside | Only work inside till weather is over |

* **Building power failure**

Table 5 Building power failure reactions

|  |  |  |
| --- | --- | --- |
| Nr. | Solutions | Explanations |
| 1 | Stand still | Do nothing to prevent driving in the way of people |
| 2 | Clear walking path | Make it easier for people to go trough |
| 7 | Light up the environment | Give light for people to see in the dark |
| 9 | Uses robots as battery | Give power to places that need it, like hospital equipment |

* **Obstructed paths**

Table 6 Obstructed paths reactions

|  |  |  |
| --- | --- | --- |
| Nr. | Solutions | Explanations |
| 5 | Avoid specific area | Al robots avoid the obstructed area till path is cleared, to prevent damage. |

* **Gas leakage**

Table 7 Gas leakage reactions

|  |  |  |
| --- | --- | --- |
| Nr. | Solutions | Explanations |
| 1 | Stand still | Shut down to prevent making a spark |
| 2 | Clear walking path | Make it easier for people to go to the exit |
| 3 | Drive to a save location | drive to a save location to save the robot from possible explosions |
| 11 | Drive hospital beds to safety | Help people that can’t walk by driving their beds away from gas leakage |

* **Earthquake**

Table 8 Earthquake reactions

|  |  |  |
| --- | --- | --- |
| Nr. | Solutions | Explanations |
| 1 | Stand still | Prevent damage by standing still |
| 3 | Drive to a save location | Drive to safer area if possible |

# Implementing features in to CoLAB

The first features that will be implemented in the CoLAB will be standard reactions to Fire, stated in the Assignment overview. For these features a sensor will be added in 4.1 Choice fire sensor. With the sensor on the robot a program will be written and explained in 4.2 Connection to the CoLAB.

## Choice fire sensor

For detecting a fire, we chose for a ‘smoke and gas sensor’. There are a couple of reasons to choose for this sensor. At first there is the difference between detecting the fire itself or the smoke of the fire. When detecting the fire, you will only be able to detect it if you really see the fire. This will give the possibility that the robot is near a fire but doesn’t see it. Detecting the gasses that are released can be the solution to this problem.

In the following Table 9 Fire detection sensors all sorts of fire detection sensors are explained. Number 3 displays the sensor that is chosen for on the CoLAB.

Table 9 Fire detection sensors

|  |  |  |  |
| --- | --- | --- | --- |
| Nr. | Figure | Name | Pros and cons |
|  |  | Infrared Flame detection | - Short distance  - Can detect other light sources  - Looks only one direction |
|  |  | Heat sensor  (PT100) | + Will detect fast if the robot is on fire  + Can be used to measure other temperature related things like the chip of the Raspberry pi  - It will take long for the sensor to get hot  - Short distance |
|  |  | Gas & smoke sensor | + Will detect more than smoke also gasses  + Will have a good range because air will spread the gasses trough the building |
|  |  | Vision camera | - Looks only in one direction  + can be used for different functions on the robot as well  + Precise localization of the fire  - expensive |
|  |  | Building fire system | + Humans can tell when they see a fire  + Big system for the whole building so much range  - Hard to connect to the CoLAB to the amount of different systems there are. |

## Connection to the CoLAB

The smoke sensor will have a direct connection to the Raspberry pi of the CoLAB. This will be done with the connection below. The Gas sensor MQ-2 has 3 used connections: a plus, minus and a digital data pin. For the pinout of both boards see I Attachment pinout sensor and raspberry pi.

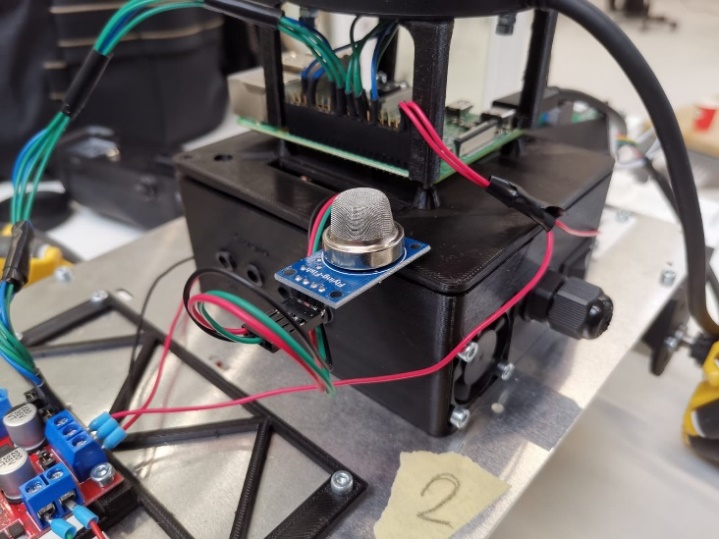
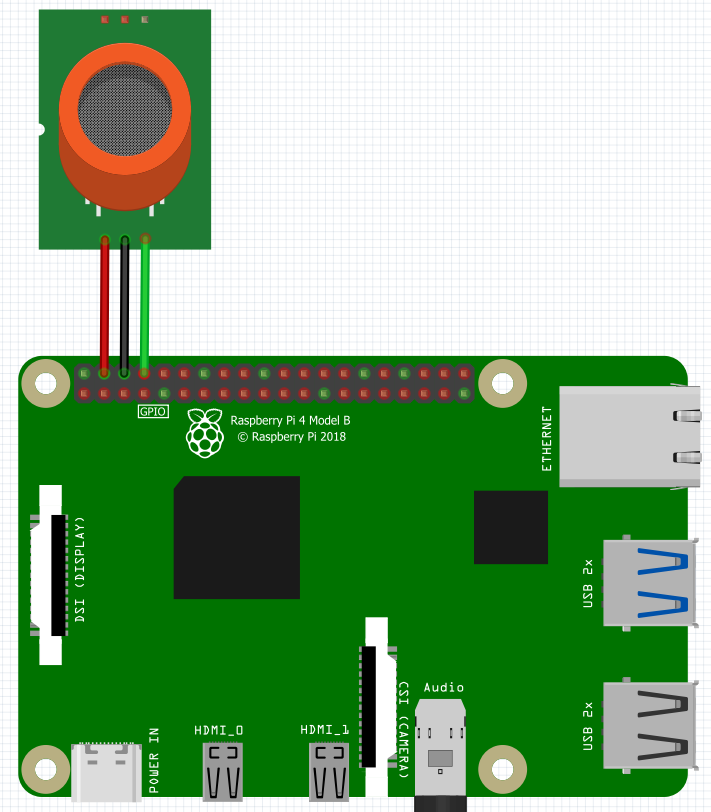


Figure 2 Smoke and gas sensor on CoLAB

The program on the raspberry pi will be programmed in Python, for Python there are a lot of libraries which make programming easier. One of those libraries is the GPIOZero library. This library makes it possible to read from and write to the IO pins on a raspberry pi. In the case for the CoLAB and the fire detection it will be used to read the output from the smoke and gas sensor.

The UWB tags, which are used to determine the location of the robot are also connected to the CoLAB and use Python to get the location data. So, when a fire is detected, it is relatively easy to get the location data and send it out to other CoLAB’s, the other CoLAB’s can use this data in their response to the fire.

## The program

The program that would be made is based on the DCRC project program. Unfortunately, this program had a lot of setbacks when the team of DCRC wanted to implement the code. The problem was between two different raspberry pi’s. One controlled the robot when the other was controlling the CoLAB (the decentralized cross robot control). The communication between those two pi’s wasn’t working because of a couple of issues that are not important for this report.

The test program for the safety features is split in two to still being able to test the program. This is done by splitting the program at the part of the communication. The programs will simulate the other raspberry pi for themselves. This will make the programs look as follow:

1. CoLAB pi
   * Read the sensor.
   * When detecting fire spread a signal.
   * Note where the fire is so other robots can avoid it.
   * \*Shut down after above points
2. Robot pi
   * \*\*When there is a fire, it starts a fire safety feature:
     + Do nothing
     + Drive to a location while avoiding the fire.
     + Clear the path

\* = To be sure a robot is not spreading the fire, it won’t be able to do anything when it detected it. Other robots will be able to move.

\*\* = depending on what environment the robots have the client will when installing decide which feature to take.

for having a better look how the programs work, it is possible to look at the following attachments:

* Flowchart CoLAB: II Attachment flowchart CoLAB
* Flowchart Robot pi: III Attachment flowchart Robot

# Testing features

In video “Firefeature direct drive to location” you can see the following events:

* Robot drives to a job (to left back) without any fire.
* A switch is clicked to simulate the fire input from the CoLAB pi.
* The car starts driving to the location where he is save (right front)

In video “Firefeature drive to location with avoidants” you can see the following events:

* Robot drives to a job (to left back) without any fire.
* A switch is clicked to simulate the fire input from the CoLAB pi.
* Hessel (in green) simulates fire, this location came from the CoLAB pi and is hardcoded, so the robot knows the location.
* The car starts driving to the location where he is save (same location as previous video)
* While driving the robot avoids the area of located fire.

In video “Firefeature clear path” you can see the following events:

* Robot drives to a job (to right back) without any fire.
* A switch is clicked to simulate the fire input from the CoLAB pi.
* The robot starts driving to the wall for 10 second.

\*This can later be improved using a lidar. This couldn’t be implemented because of the communication problem.

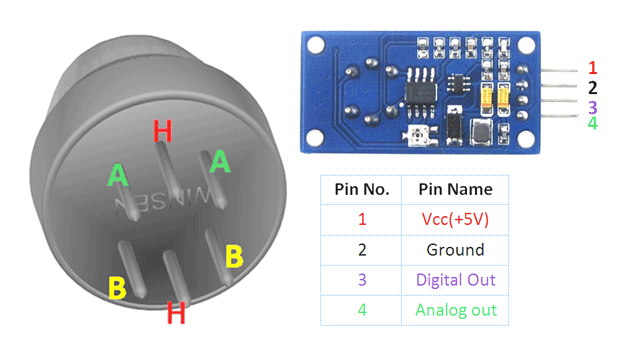
# Recommendation

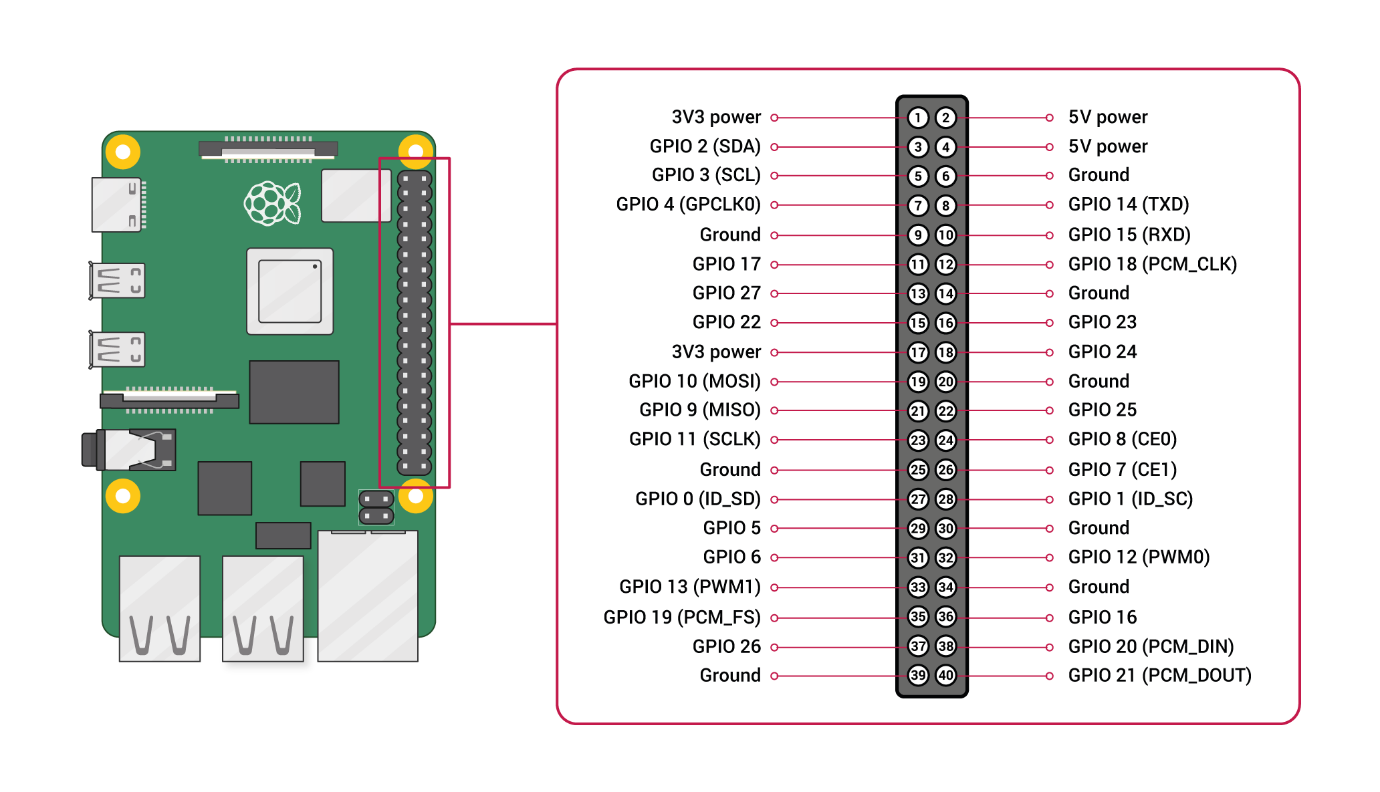
It is recommended to wait for the DCRC project to get a better working program/system. After that the existing programs made in this project can be further implemented and tested. There can be more improvements done on the path finding around doors and objects. At last other features can be added as well.

Right now, the CoLAB does not use a map of the environment for the reaction to the fire. Maybe in future projects a map could be implemented so a robot can drive to a safe location more directly and avoid dead ends. This could also be used to improve the clear the walking path reaction, because then the robot would know where the walking paths and the walls are.

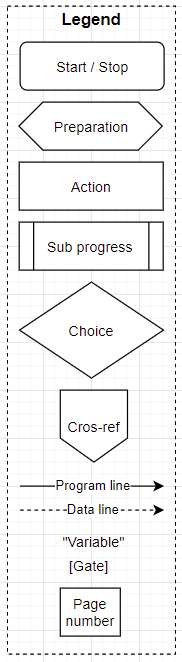
# Attachments

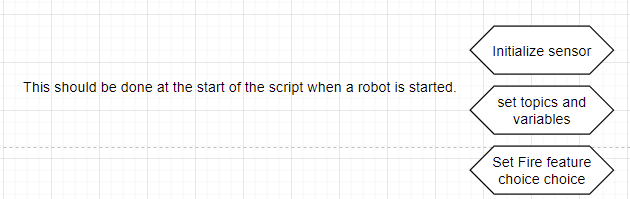
## Attachment pinout sensor and raspberry pi

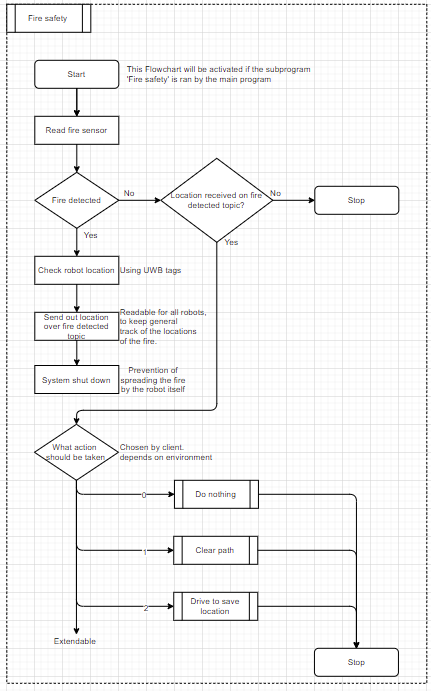




## Attachment flowchart CoLAB







## Attachment flowchart Robot

