Low-cost Autonomic Rescue Robot (LARR)

Matthias Heni, Tobias Vonier
Supervisor: Lawrence Natusch
Student Research Center Baden-Württemberg, Germany

1. Scenario/Introduction

We built a low-cost autonomic robot which can enter dangerous areas people aren't able to access. When an accident happened in such an area, it is the task of the robot to locate missing people and to help them.
The robot must be able to carry out the rescue mission in a fully autonomous mode with no human assistance.
A possible area could look like the layout below. It could happen that people are locked in rooms the robot can't enter. The task is to remember the locations of the locked doors and harmed people. Furthermore the robot deposits rescue kits to support them. To start the procedure, the robot is placed near the red arrow. Having fullfilled the tasks, the robot comes back to the starting point with information that makes lifesaving much easier for the rescue team.

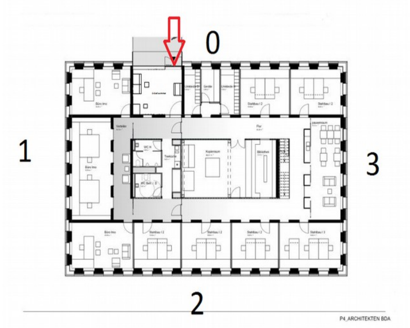

Fig. 1: Layout of a building, possible rescue area

2. Construction of the Robot

Based on the high complexity of the task on the one hand and the need of being small and flexible on the other, the robot has three levels. The different mechanical pieces of the robot are constructed with “Autodesk Inventor”, a CAD-program to reach maximum accuracy.
On the base level, two electrical engines drive four wheels. Each engine controls one side of the vehicle to have maximum flexibility. We use ball bearings to reduce friction.

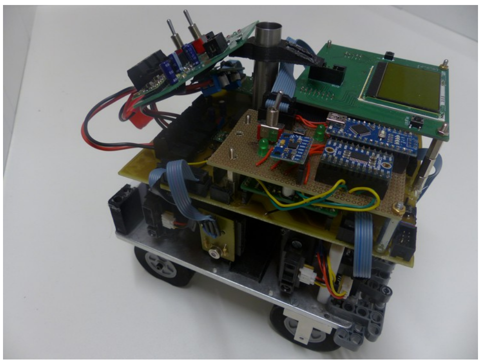

Fig. 2: Our self-designed robot

On the middle level there are the six infrared distance sensors, their control pcb and the motorized rescue kit thrower. On the upper level there's the robot's motherboard which is connected to the sensors on the middle level via ribbon cables. Above the motherboard there's the battery controller, a self-designed battery unit which automatically shuts down if there's not enough or too much voltage or if there's a short circuit. An LCD for communication completes the robot.

3. Electronics

The electronic part of the solution uses standard components as follows:

- ATMEGA1280 main processor
- Motor drivers with PWM
- Distance sensors with ADC
- Heating sensors with I²C

4. Communication/Mapping

We use Atmel Studio³ with the language C to develop the code. The solution and the map are stored on the ATMEGA1280.
Splitting up the area into little squares with 50 cm width each is the first step of the algorithm. These squares are represented by the elements of different $(length \times width \times level)$ matrices. The first one shows the path of the robot's tour through the area. Another one has marks where harmed people are and a third contains information about closed doors and other barriers.
This chart, as a part of the whole software solution, shows the driving part.

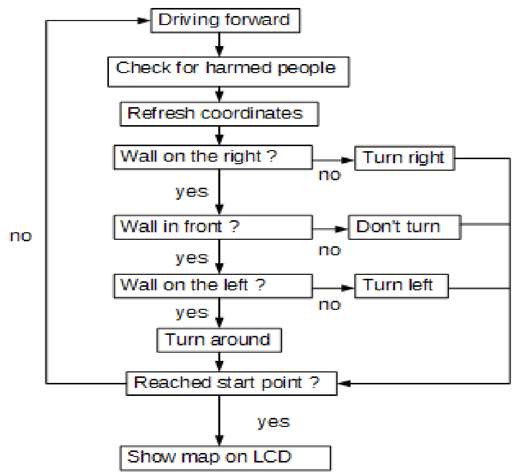

Fig. 3: Driving algorithm for one square

The part of the software dealing with the rescue kits has its own chart and is connected to this. It refers to the box “Check for harmed people”.

References

[1] https://www.heinze.de/architekturobjekt/zoom/
[2] https://www-m9.ma.tum.de/graph-algorithms/spp-a-star/index_de.html
[3] https://www.microchip.com/