CPM Lab Documentation

An Open-Source Testbed for Connected and Automated Vehicles

Informatik 11
Embedded Software

RWTHAACHEN UNIVERSITY

CPM Lab

GitHub

Remote Access

Publications

This documentation provides technical information in detail of the Cyber-Physical Mobility (CPM) Lab at the Chair of Embedded Software at RWTH Aachen University. In our Lab it is possible to test and evaluate strategies for connected and automated vehicles.

In this documentation we show you how to setup your environment for simulation only as well as for a experiments. We provide tutorials to get you started with the CPM Lab. You can read more to get a deeper look into special topics.

As you can see in the picture the Lab consists of the Main PC which provides the access to the infrastructure. It processes information from the Indoor Positioning System (IPS) and information from our user interface called the Lab Control Center (LCC). You can upload your program in the LCC which will send the commands to the vehicles Data Distribution Service (DDS) of the RTI Connext which works similar to a cloud system by providing all information to all members of the service. All vehicles and NUCs participate in the DDS as well. The NUCs process the commands and execute your program. In each NUC a local DDS provides the information with the middleware. Once all data is evaluated it is shared with the central DDS and sent to the vehicle.

Software architecture

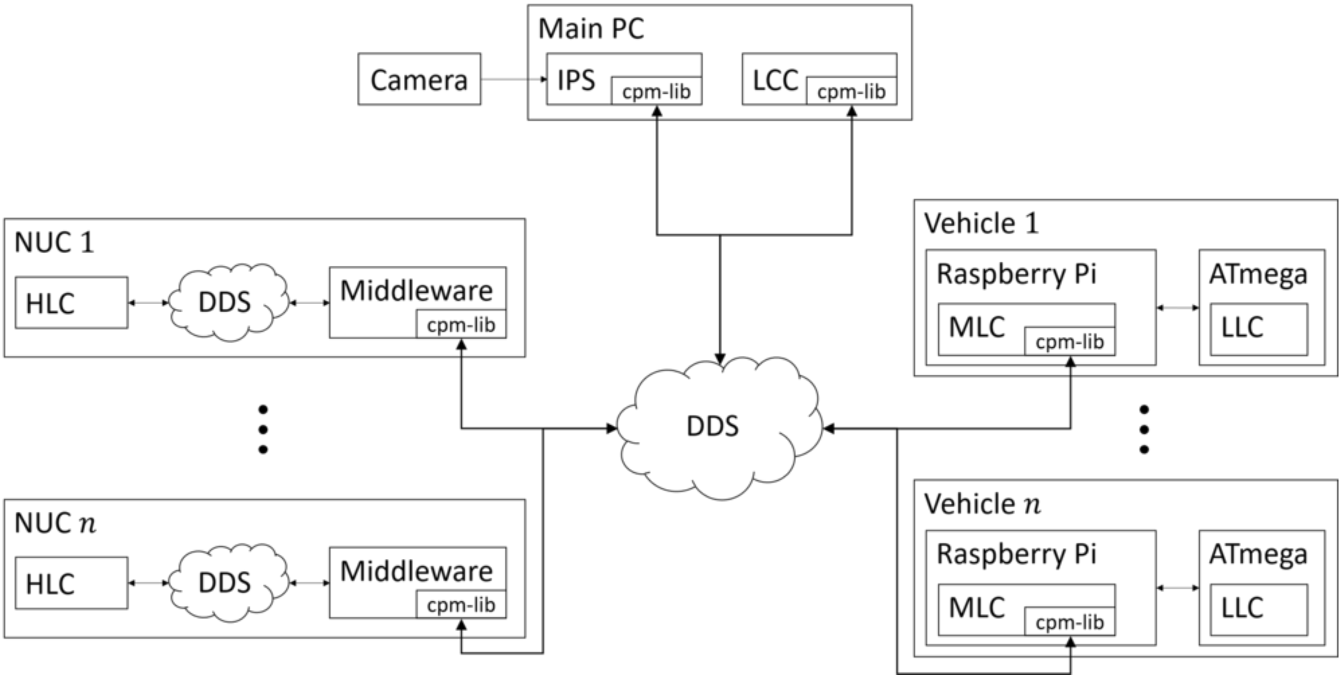
The software for each vehicle is distributed between three major components:

- The High Level Controller runs on an Intel NUC PC, which is logically associated with a particular vehicle, but located in a shelf next to the track. The HLC is responsible for decision making and coordination with other HLCs.
- The Mid Level Controller runs onboard the vehicle on a Raspberry Pi Zero W, which is connected via WiFi. The MLC is responsible for accurately following the reference trajectory given by the HLC.
- The Low Level Controller runs onboard on an ATmega2560. It is responsible for the vehicle hardware interaction and abstraction.

Two supporting software components run on a separate "Master PC":

- The Indoor Positioning System measures the vehicle poses using a camera.
- The Lab Control Center gives the user an overview of the current operational state of the lab.

All software components use our CPM library (cpm-lib). The CPM-lib provides functions to communicate with the vehicle and the IPS system. You can find functions for localization and vehicle states.


Main PC
Camera
IPS
cpm-lib
LCC
cpm-lib
NUC 1
HLC
DDS
Middleware
cpm-lib
NUC n
HLC
DDS
Middleware
cpm-lib
DDS
Vehicle 1
Raspberry Pi
MLC
cpm-lib
ATmega
LLC
Vehicle n
Raspberry Pi
MLC
cpm-lib
ATmega
LLC