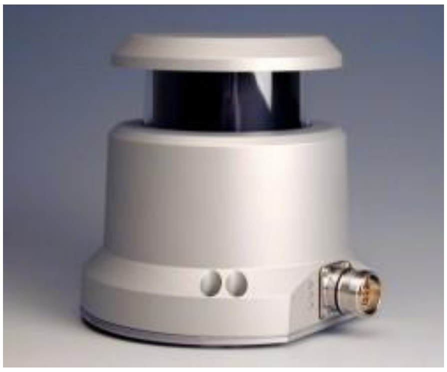
The Navigation Controller of Fraunhofer IML

A Controller Module for Free Range Driving
Automatic Guided Vehicles
(Here: Laser Based Navigation)



Author: Thomas Albrecht

Date: August 2019

1 General

For wireless guidance of an AGV the following three functions are needed that cannot be found at an AGV controller for „normal“ (optical/magnetic/inductive) line guidance procedure:

1. Orientation	meaning the determination of the current position (more precise: “pose” = X, Y, yaw angle) of the vehicle
2. Path Description	meaning the definition of a virtual guiding line and calculation of pose and speed set values
3. Path Control	meaning the control of steering and driving motors so the vehicle follows the calculated (planned) virtual path

These three functions are implemented in one controller software module, the so-called ***“Navigation Controller“*** (Nav.-Ctrl., NC). The functions are used in the same way for any kind of virtually guided AGVs and they are independent on the used method to determine the current vehicle position (e.g. Odometry/Dead Reckoning, Differential GPS, Laser Triangulation, Contour Navigation, Navigation by Floor Features/Patterns, Magnets/Transponders etc.). Furthermore, the three functions are also independent from each other, i.e. for example, the path description does not depend on the type of orientation method.

However, the Fraunhofer Nav.-Ctrl. is not only a powerful software library – it obviously also requires hardware (a controller / IPC) on which it is running. We decided to choose components of German company Beckhoff (CX5130 + I/O-terminals + TwinCAT 3). This modular and extendable hardware platform comes with different interfaces to be connected to sensors, actors and the already existing AGV controller (most probable a PLC). There is a reasonable number of different interfaces available so it is possible to connect it very fast and easily to any AGV control system.

The Nav.-Ctrl. software is intended to work as a supplement to an existing vehicle controller (this usually is a PLC). This already existing vehicle controller has to execute the following tasks (which are basic functions of any kind of AGV):

- Communication with fleet manager (for traffic control & order management)
- Communication with user interface (HMI: small text display, graphic touch display etc.)
- Evaluation of data of manual control unit
- Control of load handling device (LHD)
- Safety functions (emergency stop, bumper-stop, etc.)
- Management of orders, actual state of the AGV, etc.

Additionally there has to be done the communication with the Navigation Controller via fieldbus interface (ProfiBus, ProfiNet, CANbus). As ProfiBus-DP (with 12MBaud) is the fastest one, this one is recommended to be used.

Latest technology also enables a communication solution as follows: By using powerful Embedded IPC CX5130 of company Beckhoff, it is possible to run both Nav,-Ctrl. software as well as AGV controller software on one single hardware / on the same Embedded PC. In this case, communication between AGV controller and Nav.-Ctrl. is no longer done via fieldbus but will take place as an “internal” communication (via shared memory mechanism) within the Embedded IPC.

2 Vehicle Kinematics

By user definable parameters, it is possible to adapt all three basic functions mentioned above to common AGV types. At this moment, the following types of vehicle kinematics can be used / are being supported:

- **3-wheeler** with 1 driven & steered wheel + two fixed wheels; steer-drive-unit may be mounted centred or off-centred
- **3-wheeler** with 2 driven & steered wheels (one unit centred at the front + one unit centred at the rear side; steering angles identical, but with opposite sign) + two fixed wheels at the right and at the left side in the middle of the vehicle's body
- **3-wheeler** with 2 driven & steered wheels at the front right and front left side + two fixed wheels at the rear
- **3-wheeler** with steering by speed difference: 2 driven wheels, fixed to the vehicle's body at the right and left side + one or two idle wheels
- **3-wheeler** with steering by a pivoting sub-frame: 2 driven wheels, fixed to the sub-frame, steering by speed difference + two fixed wheels at the rear
- **4-wheeler** with 2 (independently) driven & steered wheels + 2 idle wheels
- **4-wheeler** with 2 (independently) driven & steered wheels and 2 additional steered (non-driven) wheels
- **4-wheeler** with 4 (independently) driven & steered wheels

3 Interfaces

3.1 Determination of Relative Position

For any type of AGV, dead reckoning algorithms are used to determine the current pose of the vehicle. The pose will be calculated relatively to a known starting position. This method is quite easy to implement and its result is fairly precise at the beginning of trip, but the precision will significantly decrease while driving for a longer distance. The following data / sensor data has to be read in:

- *Covered Distance* by measuring the revolutions of the driven wheel(s) or of additional measuring wheel(s) with *incremental encoder(s)*; signals shall be 5V push-pull according to RS422 standard or 24V signals, frequency shall not be higher than 500 kHz (Hint: there will be a Quadrature-Counter-Interface at the Navigation Controller!); signals may also be generated by motor controller (so called encoder emulation) instead of real encoder(s); resolution of distance measurement shall be at least 0.5 mm/incr
- *Steering Angle(s)* by using absolute angle decoder (output: 12 Bit parallel with Gray-Code or serial interface with SSI) or analogous value (potentiometer) or incremental resolver; measurement precision of 1/10° is required

3.2 Determination of Absolute Position

To improve the lack of precision of odometry/dead reckoning, an absolute reference system shall additionally be used. Laser triangulation with laser scanners of company Götting (HG43600-Z) or company Pepperl+Fuchs (R2000) is currently supported, as are the "Jupiter" and "Triton" sensors from Dutch company Accerion for localization by means of floor features. For outdoor vehicles, a differential GPS receiver from company Götting can be used (other sensors/methods on request). Depending on the type of sensor system used, data processing (= position calculation) takes place either inside of the sensor or it is done by Nav.-Ctrl. software. Accordingly, the Nav.-Ctrl. will provide a variety of different interfaces (hardware as well as software) to sensors from different manufacturers.

3.3 Interfaces to Line Guidance Sensors

For the sake of completeness, it shall be mentioned that it is possible to also use the Nav.-Ctrl. in vehicles equipped with sensors for physical guidance (i.e. to follow an optical/magnetic/inductive line). Usually these kind of sensors deliver data as analogous output voltage (-10V .. +10V) or by fieldbus signal – therefore Nav.-Ctrl. can be equipped with hardware and software interface to read in and evaluate such data.
This might especially be interesting for so-called “Hybrid Navigation” which means a combination of physical and virtual guidance, each of it to be used in a situation it fits best to. For instance, an AGV might be line guided within small aisles and outside of the aisles – e.g. for change-aisle-maneuvers or to reach/serve manual order picking working stations – laser navigation will be used for flexible driving path arrangement.

3.4 Interfaces to Actuators / Drives

Above-mentioned software *modules path planning* and *path control* generate speed set values for the vehicle’s driving motor(s) as well as for the steering motor(s) (if existent). To achieve a nice/smooth behaviour and a high positioning accuracy of the AGV it is required to use motor controllers with a built-in speed controller (closed loop speed control). To make it clear: Nav.-Ctrl. does NOT provide any speed control loop! Speed set values can be transmitted to motor controllers as analogous values or via (fast) fieldbus communication. When using analogous speed set values it is recommended to use voltage from -10V … +10V, representing -100% ... +100% of max. rotation speed of driving/steering motor.
The speed set values can be directly sent from Nav.-Ctrl. to motor controllers or via AGV controller – assumed that the communication between these two is fast enough (also refer to block diagrams in chapter 6).

4 Path Description

The absence of a physical guidance line or similar objects that determine the path of the vehicle, requires an abstract form of path description. This shall also lead to determined vehicle behaviour, which is equivalent to that of a line-guided vehicle. The path description will be predetermined directly by the user in a fixed, mathematical form. This procedure corresponds to physical reference lines and it will thus be called „virtual reference line“.

A complete path consists of at least two decision points (DPs). The first DP is the starting point and the last DP is the destination point. Between these two DPs there can be several additional DPs, for example at a branch. Obviously, it is also required to add the desired speed of the AGV, this might vary along the path, e.g. according to driving in a curve, passing a door, approaching a machine etc.

Normally all possible paths are split up into several block-sections to control more than one AGV without any risk of collision (within one block section normally only one single AGV is allowed to stay at the same time).

For Fraunhofer IML's Navigation Controller there is available a PC-based tool named *CadCourse* (for Windows 7, 8, 10) which allows the configuration of DPs, block-sections etc. The paths between the DPs are generated automatically by this software by taking into consideration the kinematic principle of the AGV and all important parameters as wheel distance, track gauge etc.

A complete transportation order consists of a list of partial orders that precisely describe driving manoeuvres, for example „drive to DP 81". Additionally so called signal-functions allow the user to enable several functions dependent on the AGV's position, for example „switch horn on", „call elevator",„open door", „wait for sensor signal" etc. For these purposes, 48 "remote" bits are available in both, CadCourse as well as Nav.-Ctrl.. The meaning of all of these bits can be defined by the user.

5 Operation modes

5.1 Manual Mode

In manual mode, the user can control the steering angle and speed with the help of a manual control unit. In case of a 4-wheeler AGV the user can additionally select cornering or traversing steering mode or "turning on the spot" (change heading of the vehicle with smallest envelope).

5.2 Semi-automatic Mode

In semi-automatic mode, the behaviour of the AGV is similar. Additionally, when the AGV comes close to a virtual guiding line steering will be taken over by the navigation controller so that the AGV automatically moves towards the path and can be set up (prepared for automatic mode) at an DP.

5.3 Automatic Mode

In automatic mode the manual control unit is disabled, Navigation Controller takes over control of driving and steering.

6 Data Exchange

In order to achieve a high-speed communication it is recommended to use a serial link with ProfiBus-DP protocol for data exchange between AGV controller and Navigation Controller. A specific and detailed description of this communication interface is available as a separate document ("Navigation Controller with ProfiBus Communication, Specification of the Interface Protocol").

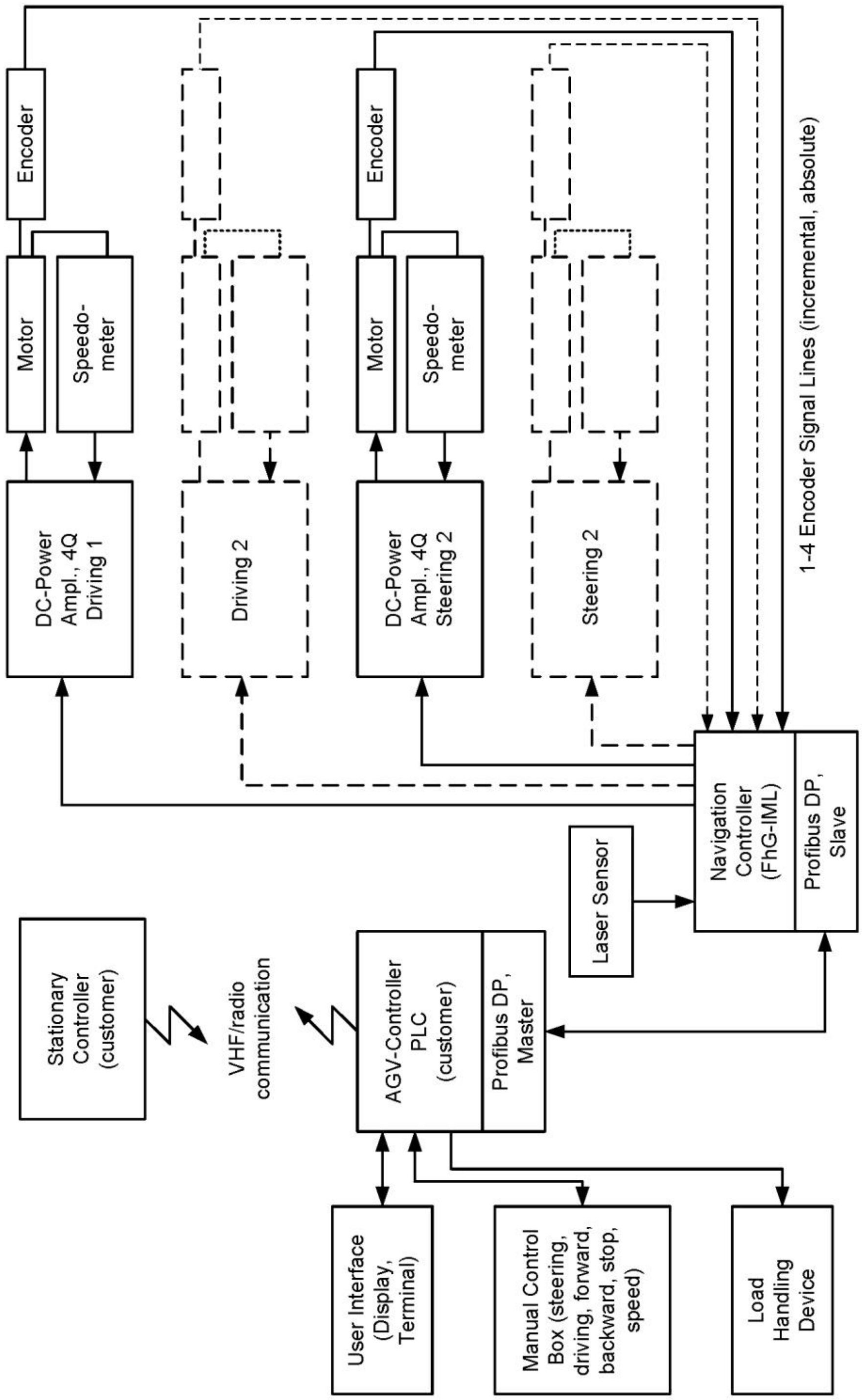
In case the "integrated solution" of AGV controller + Navigation Controller on one single Embedded Controller (Beckhoff CX5130) is used, there will be "internal" communication via shared memory data exchange mechanisms according to details specified in another document.

As only one controller hardware is being used with the integrated solution, this obviously is smaller (space saving), cheaper (cost saving) and faster (fast data exchange) than any solution based on two separated controllers – and therefore this is the recommended solution.

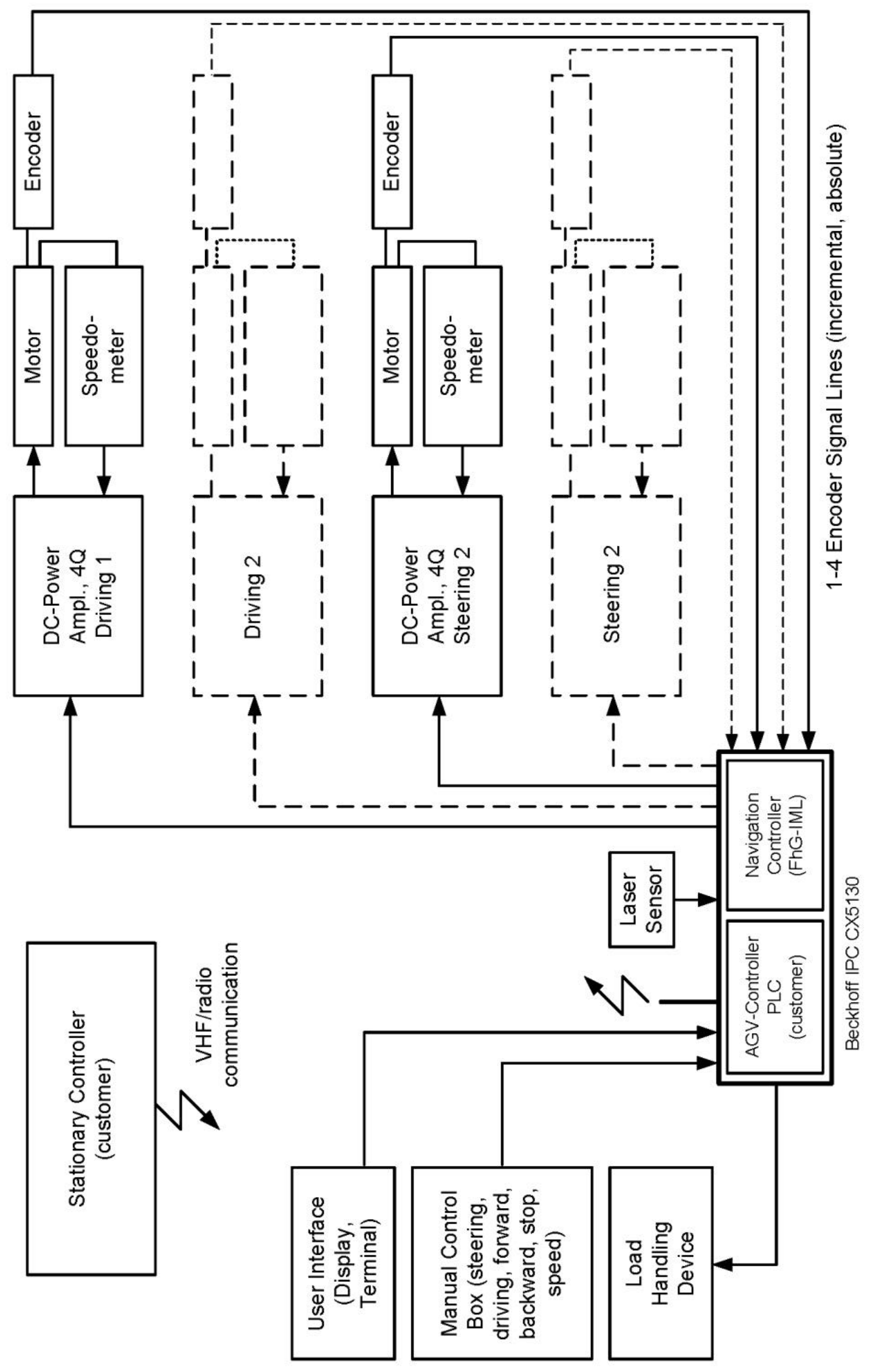
The integrated solution is also faster in this meaning: we provide quite a lot of PLC code free of charge that, among other things, already comes with the complete data exchange interface. Therefore this is almost a plug & play solution – assumed that the client is already familiar with Beckhoff hardware and software (TwinCAT 3, Microsoft Visual Studio).

However, there are several clients with many AGVs built, who use a PLC (e.g. Siemens S7-315DP) and ProfibusDP communication to NC without any drawbacks.

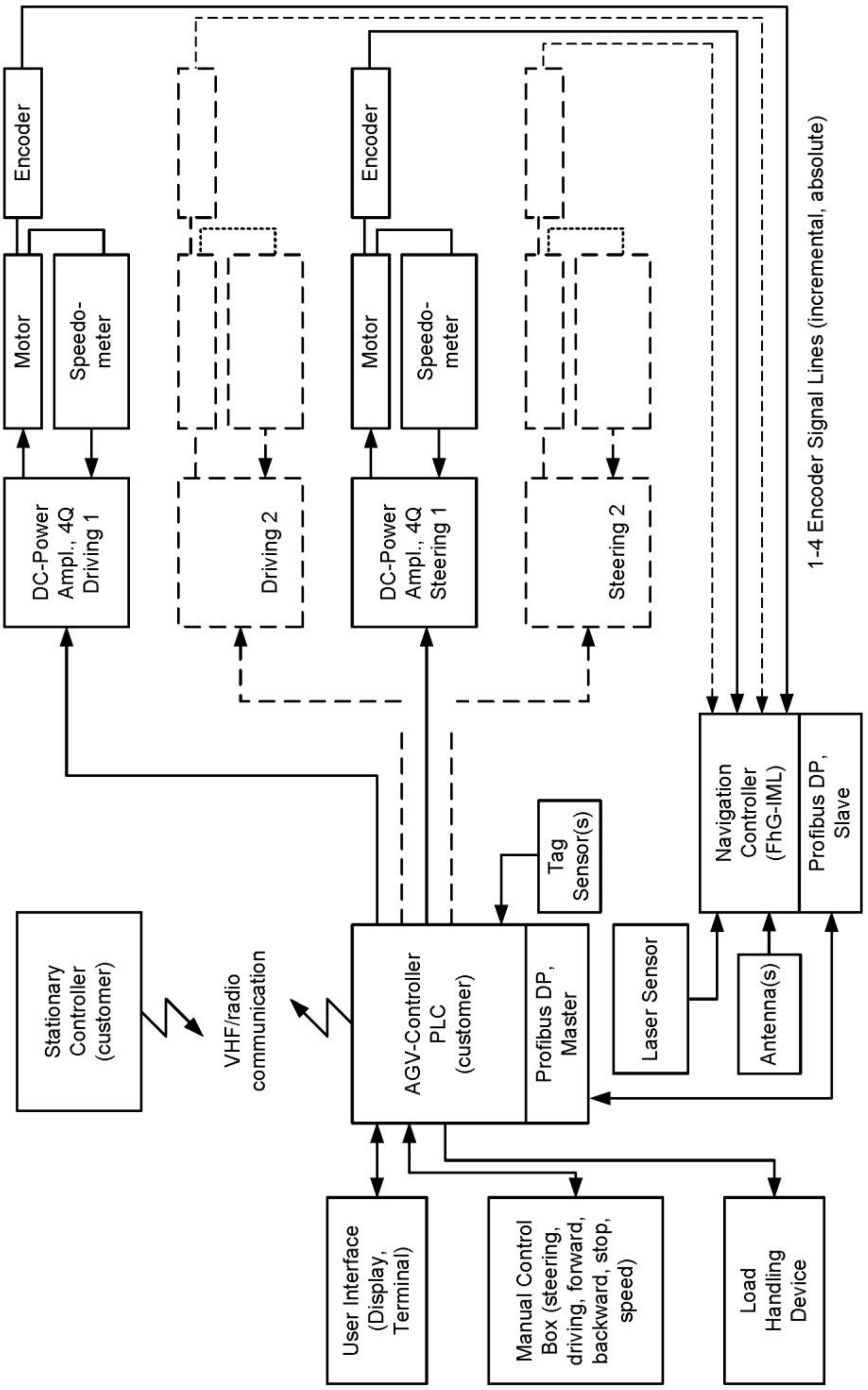
7 Block Diagrams



1.) Two separate controllers, data exchange via ProfiBusDP; Navigation Controller with Laser Triangulation



2.) One controller, data exchange via internal/shared memory; Navigation Controller with Laser Triangulation



3.) Two separate controllers, data exchange via ProfiBusDP; Navigation Controller with Hybrid Navigation (i.e. Laser Triangulation + physical guidance)