Towards a Multi-Agent Platform for Cyber-physical Systems based on Low-power Microcontroller for Automated Intralogistics

A Minimized Embedded Solution for the Internet of Things in Intralogistical Environments

Arne Stasch[1] and Axel Hahn[2]

[1] *OFFIS Institute for Information Technology, Oldenburg, Germany*

[2] *Department of Computing Science, Carl von Ossietzky University, Oldenburg, Germany*

Keywords: Cyber-physical System, Multi-Agent System, Material Flow, Logistics.

Abstract: Today the fluctuating market requires more flexibility from central controlled material flow systems. A new approach such as the Internet of Things is able to turn the common structure into a cognitive decentralized system. This paper addresses a modularized solution for material flow systems in intralogistical environment. The promising concept of Internet of Things leads to an idea of a cyber-physical system with a Multi-agent platform which is presented in this paper. In order to develop a system which is capable of meeting industrial needs, a low-power microcontroller is chosen as basis for the Multi-agent system. The combination has advantages but also restrictions which are discussed. A demonstrator for the future implementation of the system and evaluation is introduced.

1 INTRODUCTION

Flexibility is an increasing key requirement in intralogistical environments. The material flow systems (MFS) used in this field have to bear alternations in the flow rate and flow directions. Common material flow systems consist of centralized programmable logic controllers (PLC) and are designed for a certain capacity. By the time requirements of the MFS change so that a retrofitting is necessary or reasonable, then a conflict occurs: Changes of the MFS are time- and cost-intensive for logistic providers.

Modular material flow systems are one solution of this problem (ten Hompel, 2011). Among the approaches to transfer promising technologies to a modular MFS, the concept of “Internet of Things” sticks out (Günthner, 2010). In this concept, the intelligence is distributed over a decentralized control system. This results in engineering advantages. The reusability of soft- and hardware solutions will increase and the systems diversity will be reduced for example (ten Hompel, 2011). This effect will reduce development and working costs.

The MFS control architecture was basically stacked in three layers so far (Günthner, 2010). These layers are the physical, the PLC and the coordination layer (see Figure 1). The physical layer represents mechanical and electrical parts. These are controlled by the PLC layer which accesses sensors and actuators. The other layers are coordinated by the coordination layer which consists of an IT-infrastructure.

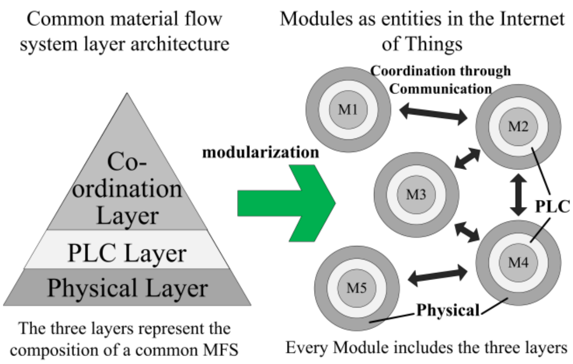

Figure 1: Transferring of common MFS layer into the concept Internet of Things.

In order to modularize the common MFS, every module must implement this structure. Figure 1 visualizes the change to a decentralized concept.

Stasch A. and Hahn A..
Towards a Multi-Agent Platform for Cyber-physical Systems based on Low-power Microcontroller for Automated Intralogistics - A Minimized Embedded Solution for the Internet of Things in Intralogistical Environments.
DOI: 10.5220/0004583004290433
In *Proceedings of the 10th International Conference on Informatics in Control, Automation and Robotics* (ICINCO-2013), pages 429-433
ISBN: 978-989-8565-71-6

The concept presented in this paper takes this modularized structure and transfers it into an embedded solution. Every module has its own physical sensors and actuators (Physical Layer). They are controlled by a microcontroller with a Multi-agent system (MAS) platform (PLC Layer). Agents coordinate their tasks through a fieldbus (Coordination Layer).

The "Internet of Things" can be implemented in different ways such as Near Field Communication (NFC), Wireless Sensors and Actuators Networks (WSAN) or IP for Smart Objects (IPSO) (Atzori, 2010). Two appealing ways are described in the following. Radio Frequency Identification (RFID) tags allows packages to carry their own information or code which can be altered. Multi-agent systems (MAS) are systems which consist of entities which can collectively solve problems (Wooldridge, 2002). Those technologies are solutions to distribute the intelligence.

The RFID tags have to be attached on packages and need an intelligent infrastructure. Electromagnetic interference (EMI) and certain materials (metals, liquids) can restrict the use of it in industrial environments (White et al., 2007).

A MAS needs a modularized infrastructure to be meaningful. The task, the MAS has to fulfil, has to be divisible so that different agents work on the same task. Further a platform that gives multiple agents the ability to communicate and interact with its environment is needed. To approve the reliability of a system which coordinates through communication, a pervasive testing is necessary.

Taking the idea of RFID with adaptable identities (ID) where the packages can represent themselves in the software and transfer this to an agent who represents the package in a MAS is on a par with the idea of ten Hompel (ten Hompel, 2011). Creating a MAS which can be used in an industrial, intralogistical environment on a modularized MFS is the next step.

MAS are used in different applications like autonomous driving (Beeson, 2008) or decentralized control of Automatic Guided Vehicles (AGV) (Weyns, 2008). These approaches use computer systems to realize the MAS infrastructure. The approaches of geo-references swarm agents (Barbera, 2010) or a hardware platform for analogue power simulator as a test environment for smart grids (Spencer, 2010) appeal more suited for a decentralized control system. They use embedded controllers as basis but one controller represents only one agent. This is not enough for a platform that has to handle more than one agent. With the above mentioned MAS for MFS such an embedded software platform is needed and is presented in section 3.2.

Modules which communicate with each other and interact with the environment physically are called cyber-physical systems (CPS) (Lee, 2008). Therefore the aim will be to create a CPS which can replace common MFS. Microcontrollers can meet all industrial standards and are widely used. Although they are not powerful, they meet the requirements for networked systems which can divide their tasks. The decision of taking microcontrollers as basis is discussed in the next section.

The paper is arranged in the following order: The concept of a new cyber-physical system is presented in section 2. In section 2.1 MAS frameworks and their benefits are discussed in the described scenario. The own idea of a MAS platform is presented in section 2.2. In the following section 2.3 the demonstrator which will be used to implement the developed control system is described and is followed by the conclusion in section 3.

2 A CONCEPT FOR A NOVEL CYBER-PHYSICAL SYSTEM

The goal is to create a CPS which can substitute the common MFS and deliver new features as loose coupling. It should be possible to couple or decouple hardware parts and the software completes the reconfiguration autonomously. If the MFS will be split in fine-meshed modules then there are more options to rearrange the MFS to be on a par with the changing requirements. Every storage shelf, transport-vehicle and reasonable smallest part of a conveyor can be a capsuled module. This means that in every module an electronic control unit (ECU) has to control the module and communicate with other modules.

Communication between these modules is the key for smooth control and coordination. Every entity has to fulfil the functions of all layers, mentioned in section 1, but only between the borders of the module. The modules must communicate to achieve the tasks of the whole MFS.

By creating a modular system, the system structure has to be changed (ten Hompel, 2011). Figure 2 shows the previous horizontal view and the modularized vertical view of the systems structure. In every development step the vertical view must be considered.

These separated modules only need the resources for a small contingent of the whole system. An embedded solution sticks out against PCs and PLCs because a module does not need the processing power of these. Small, optimised, real-time capable systems which can be mass-produced for low cost could be enough for the task.

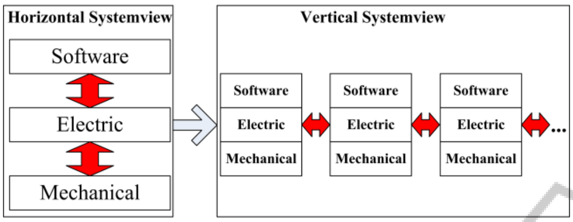

Figure 2: Changing horizontal to vertical system view.

Hardware-requirements for an ECU are:

1. Enough processing power to control the mechatronic parts and handle the communication to other modules.
2. Low-power consumption for modules which are not connected to a static power supply.
3. Reliable for safety critical applications
4. Interfaces are compatible to industrial standards.

These requirements are not very specific. However they give a direction to small microcontrollers as MAS devices. After defining the hardware, suitable software is needed. This is discussed in the next section.

2.1 MAS Framework for embedded Solutions

A multi-agent system (MAS) delivers handy functionality for MFS. Agents represent the modules and packages in the software. This distributes the responsibility for the package transport to the modules. Other functionalities like routing can be implemented as agents, too.

Requirements, like real-time operations, industrial standards and small footprints sort out most of the existing MAS frameworks. JADE (Bellifemine, 1999) is a very common used MAS framework but doesn't meet the requirements stated above. Mobile C uses C/C++ as programming language and C meets industrial standards (Chou et al, 2010). However Mobile C needs a General Purpose Operating System (GPOS) and even the smallest embedded versions, e.g. Embedded Linux, need more footprint than 259 KB Flash Memory[1] and this does not account RAM usage.

[1] http://www.lynuxworks.com/products/whitepapers/xp-vs-linux.php3: conclusion last visited 02.04.2013

In the search of agent friendly architectures AUTOSAR (AUTomotive Open System ARchitecture) stands out (Heinecke, 2004). It is a standardized system which solves software development issues for automotive ECUs. The architecture is layered and separated between applications and basic software which is dependable on hardware. The basic idea is to reuse written applications on any device independent from hardware. This architecture allows the applications to communicate to other applications on the same ECU and on other ECUs with the same interface. The interface is provided by the AUTOSAR runtime environment (RTE). A patent from the DAIMLER AG[2] describes the use of agents with AUTOSAR for ECU Diagnostics.

Because of the specifications from the automotive sector, AUTOSAR is not lean. It has a footprint of 256KB on the leanest AUTOSAR ECU (Bunzel, 2011). It is not optimised for agents and their interactions. Exemplary, only one application has access rights to a specific part of the memory supplied by the RTE. This application has to manage and deliver it again to other applications with ports through the RTE. Therefore the implementation of the agent communication which needs access to memory is more complicated.

The listed memory resources do not look much in the embedded world today. This approach is to aim for a minimal, reliable and cheap solution for industrial use. The MAS framework should be lean as possible because the embedded environment should be chosen due to the application and not on the framework. A suitable MAS platform is presented in the next section.

2.2 Concept of MAS Platform for CPS

Using the software architecture of AUTOSAR gives a basis for a MAS framework which suits for this CPS. Figure 3 depicts an adapted AUTOSAR architecture. On the ground level, the ECU hardware is represented. The background level (in AUTOSAR language basic software) represents the Real-Time Operating System (RTOS), the hardware drivers, and the driver's service interfaces which are unified APIs. Agents reside in the Agent Level were an Agent RTE connects them with each other and the Background Level.

Creating the Agent RTE will be the key task in the future work. It has to support the agents with

[2] Patent Daimler AG: (WO2008095518) USE OF A DISTRIBUTED DIAGNOSTIC ARCHITECTURE IN AUTOSAR

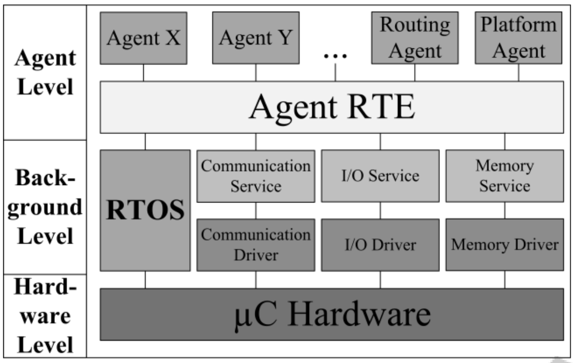

Figure 3: Adapted AUTOSAR Architecture.

all necessary information and access to fulfil their role autonomously. Every Agent needs to run in a capsuled task so it can act independently. The Agent RTE has to manage all communication between the agents and agents on other modules. In Figure 4 the

Routing Agent communicates with a Package Agent on the same module and with a Routing Agent on another module. The Agent RTE and the services below handle the communication. However, the interface the agents use is independent from their location.

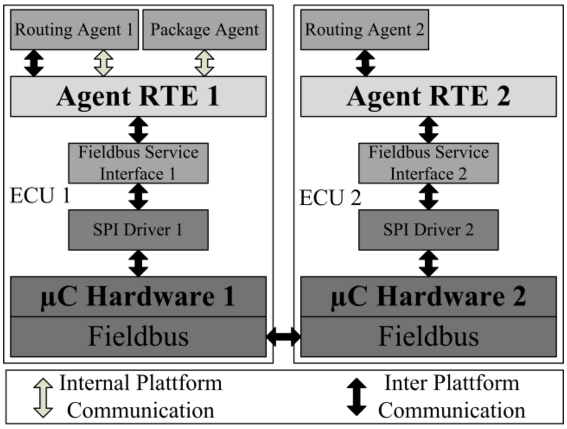

Figure 4: Agent-to-Agent Communication.

The functionality the Agent RTE has to provide the agents has to be altered by the knowledge about the functionality of other MAS frameworks like Mobile C. Because the limitations of the microcontroller can hinder the development of some features, it has to be analysed whether it is a fully-fledged MAS platform or it has reduced functionality.

In the next section the environment where this MAS framework will be implemented is introduced.

2.3 Implementation in a Material Flow System

The research project CogniLog (Overmeyer, 2012) deals with automated, cognitive logistic-networks. A demonstrator composed of conveyor modules and forklifts will be built. It will show the capabilities of modularized conveyor systems.

The setup consists of conveyers, Industrial PCs (IPC) with Soft PLCs, frequency convertors, electrical engines, and sensor barriers. A Profibus fieldbus is responsible for the communication between the electronic parts.

Per every “capsuled platform” a microcontroller will be implemented. A “capsuled platform” is defined as a conveyor which actuators control solely itself and no other platforms. The intersections must have sensor barriers to detect incoming or outgoing goods. The microcontroller completes the capsuled platform to a module.

In a full modularized environment the microcontrollers replace the PLCs. To minimize development time the PLCs remain as a middleware for actuators, sensors and bus communication. The microcontrollers represent the distributed intelligence. They will do the planning, coordination, and monitoring.

The microcontrollers which are used are Texas Instruments (TI) MSP430F6638 with a 16-Bit Architecture, 256 KB flash memory, 16 KB RAM, SPI and I²C Interfaces. An EEPROM of 128 KB non-volatile memory and a Proficonn Profibusadapter is connected with the SPI Interface.

In figure 5, a draft of the connections between the different electronic parts is shown. The modules represent the separated conveyor parts.

Although this setup does not correspond with the targeted decentralization, it shows most of the functionality which a fully decentralized system will have.

The previously described demonstrator setup uses only established technologies. The MFS can be logical modularized but not real physical. The conveyor parts are heavy and not exchangeable. Heavy electrical cabinets and many electrical cables do not allow fast reconstruction of the whole setup. But a conveyor system which can be rearranged is not the aim of this research. The purpose is to create a technology that enables the creation of such a system. In this setup the MAS platform can be tested on the usability of coordination on communication. The loose coupling can be shown by connecting and disconnecting modules to simulate a retrofitting. Additionally the system has to fulfil the same role as a common MFS setup. No information should be lost in the communication and no package routing problems should occur.

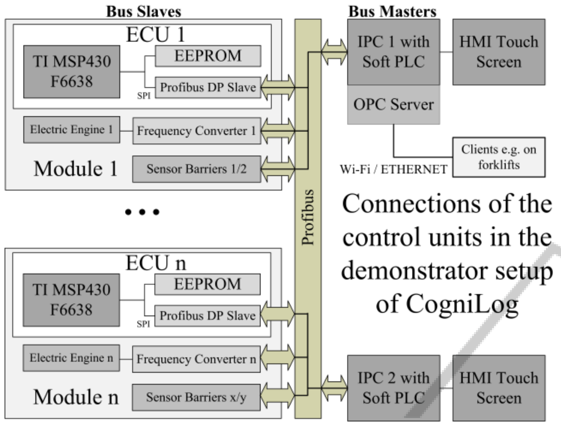

Figure 5: CogniLog demonstrator setup.

This should demonstrate that small embedded controllers with a proper multi-agent system framework can be a flexible way to bring the ideas of "Internet of Things" to material flow systems.

3 CONCLUSIONS

This paper describes the ideas of a modularized intralogistical control system. After a summary about suitable technologies a new approach is proposed. Capsuled modules, which consist of their own sensor barriers, actuators, intelligence and communication abilities, form a new MFS. The intelligence is distributed through a MAS platform for small embedded controllers. This platform will be constructed for this purpose. This paper also describes the implementation environment inside the CogniLog project. This will be an authentic test environment of the usefulness of this approach of a CPS.

After the implementation phase the Agent RTE has to be compared with other MAS frameworks to evaluate the different features and cuts. When this is done, the possible features can forecast a profit in other domains. If a mechatronic task can be split in many parts which do not need many processing power, the new MAS platform could bring an advantage.

REFERENCES

ten Hompel, M., Nettsträter, A., Feldhorst, S. & Schier, A. 2011. `Engineering of Modular Material Flow Systems in the Internet of Things`, *at – Automatisierungstechnik,* vol. 59, no. 4, pp. 248-256. Oldenbourg Wissenschaftsverlag.

Günthner, W. & ten Hompel, M. 2010. *Internet der Dinge in der Intralogistik – engl.: Internet of Things in intralogisitics*, Springer Verlag. Berlin Heidelberg.

Lee, E. 2008. `Cyber Physical Systems: Design Challenges`, *11th IEEE Symposium on Object Oriented Real-Time Distributed Computing (ISORC)*, pp. 363-369.

Chou, Y., Ko, D. & Cheng, H. 2010. `An embeddable mobile agent platform supporting runtime code mobility, interaction and coordination of mobile agents and host systems`, *Information and Software Technology,* vol. 52, no. 2, pp. 185-196. Elsevier B.V..

Overmeyer, L., Krühn, T., Hahn, A. & Pinkowski, J. 2012. `CogniLog – Cognitive Logistics for Warehousing`, *6th International Scientific Symposium on Logistics*, pp. 104-122.

Wooldridge, M. 2002. *An Introduction to MultiAgent Systems,* John Wiley and Sons Ltd.

Heinecke, H., Schnelle, K. P., Fennel, H., Bortolazzi, J., Lundh, L., Leflour, J., & Scharnhorst, T. 2004. 'AUTomotive Open System ARchitecture-an industry-wide initiative to manage the complexity of emerging automotive E/E-architectures`, Paper presented at the *Convergence International Congress & Exposition On Transportation Electronics,* Detroit, Michigan, United States

White, G., Gardiner, G., Prabhakar, G. & Razak, A. 2007. `A Comparison of Barcoding and RFID Technologies in Practice`, in Journal of *Information, Information Technology, and Organizations,* vol. 2, pp. 119-132.

Bellifemine, F., Poggi, A., & Rimassa, G. 1999. `JADE–A FIPA-compliant agent framework`, In Proceedings of *The Practical Application of intelligent Agents and Multi-Agent Technology (PAAM),* vol. 99, pp. 97-108.

Atzori, L., Iera, A. & Morabito, G. 2010. `The Internet of Things: A survey', *Computer Networks,* vol. 54, no. 15, pp. 2787-2905.

Beeson, P., O`Quin, J., Gillian, B., Nimmagadda, T., Ristroph, M., Li, D. & Stone, P. 2008. `Multiagent Interaction in Urban Driving` in Journal of *Physical Agents: Multi-Robot Systems,* vol. 2, no. 1, pp. 15-29.

Weyns, D., Holvoet, T., Schelfthout, K. & Wielemans, J. 2008. `Decentralized Control of Automatic Guided Vehicles`, Proceedings of *the 23rd ACM SIGPLAN conference on Object-oriented programming systems languages and applications*, pp. 663-674.

Barbera, S., Stallo, C., Savarese, G., Ruggieri, M., Cacucci, S. & Fedi, F. 2010. `A Geo-Referenced Swarm Agents Enabling System for Hazardous Applications`, Proceedings of the *12th International Conference on Intelligent Systems, Modelling and Simulations,* pp. 598-603.

Spencer, M., Feliachi, A., Pertl, F., Pertl, E. & Smith, J. 2010. `Hardware Platform for Multi-Agents System Developement`, *International Journal of Latest Trends in Computing,* vol. 1, no. 2, pp. 47-52.