Novel approach for wireless commissioning and assisted process development based on Bluetooth Low Energy

1st Pallasch Christoph
Laboratory for Machine Tools
RWTH Aachen University
Aachen, Germany
c.pallasch@wzl.rwth-aachen.de

2nd Peitz Alexander
Laboratory for Machine Tools
RWTH Aachen University
Aachen, Germany
alexander.constantin.peitz@rwth-aachen.de

3rd Herfs Werner
Laboratory for Machine Tools
RWTH Aachen University
Aachen, Germany
w.herfs@wzl.rwth-aachen.de

4th Schmeink Anke
ISEK Research Group
RWTH Aachen University
Aachen, Germany
anke.schmeink@rwth-aachen.de

5th Dartmann Guido
Environmental Campus
Trier University of Applied Sciences
Trier, Germany
g.dartmann@umwelt-campus.de

***Abstract*—The configuration and integration of automation devices into an existing communication infrastructure requires high efforts due to parameterization and commissioning processes. In addition, a linkage between device properties / values to logical identifiers used in process control code has to be adapted using a correct mapping of memory areas within the process memory acquired from the device. In order to exchange an automation device by a similar one, all steps above have to be adapted and checked until it can be used productively. To meet flexibility requirements of future automation scenarios, the integration and commissioning process of automation devices has to be made more flexible, less error-prone and less dependent of boundary conditions. This paper presents a concept for self-description of automation devices based on wireless communication using 802.15.1 (Bluetooth Low Energy) to facilitate commissioning processes of devices and assist engineers in process definition and device selection during development phase. A demonstrator is realized to validate this concept.**

***Index Terms*—Wireless communication, 802.15.1, Bluetooth Low Energy, commissioning, automation and control, self-description, process development, process control, plug and produce, online process development, offline process development, generic information model**

I. INTRODUCTION

Modern production is characterized by an interlocking of information and communication technology in production. In the field of industrial automation, this induces flexible use of production resources and rapid adaptation of production processes. Although digitization has increased, industrial automation faces the challenge of mastering this flexibility and coping with ever-increasing process complexity within a shorter time [1]. In industrial automation, cable-based transmission technologies are still mainly used in combination with proprietary field protocols or commonly-used Ethernet-based protocols. While latency and reliability requirements are met, the configuration and integration of automation devices into an existing communication infrastructure via field bus requires high efforts according to today's standards. In addition to the mechanical and electrical integration, initialization steps such as address assignment or the definition / division of the memory via a device information file etc. must be carried out for each automation device before it can receive information from a higher-level controller and operate as a valid bus device [2]. If an automation device is replaced by another one of a similar functionality, these steps must be carried out again when the system is shut down. In addition, the actual exchange of information, due to non-standardized data formats and representation of available measured or manipulated variables, is subject to considerable risks when adapting code from programmable logic controllers (PLC) in the event of a device exchange. Often, existing program code has to be adapted or rewritten in order to guarantee existing process functionality [3]. Wireless communication technologies offer spatial invariance and thus mobility as well as ubiquitous availability without prior infrastructure planning of the transmission medium (as done with cable-based approaches etc.). The use of wireless communication technologies in combination with semantic information models holds great potential for wireless commissioning and configuration. Information models enable a generic and comprehensive description of available measurement and control variables as well as existing device capabilities. The resulting advantages are flexible reconfiguration, reduced configuration effort and direct communication with field devices.

This paper provides concepts and methods for commissioning, configuration and control of field devices using wireless communication, enabling new development methodologies for automation processes. The outline of the paper is structured into five sections. Section II describes related work concerning wireless technologies, self-description of devices and generic device control. Section III introduces the concept and methods developed in this paper. Section IV presents new development methodologies based on presented concepts of Section III. Section V shows a demonstrator setup used for validation. Finally, Section VI summarizes this paper and gives an outlook for future work.

II. RELATED WORK

This section gives a basic overview of all related topics in the field of wireless technologies, self-description of devices and generic device control.

A. *Wireless technologies*

Wireless technologies can be broadly divided into licensed and unlicensed technologies. Licensed technologies use dedicated frequency bands allowing the explicit use of frequency components for individual applications as interferences of coexisting technologies do not have to be taken into account. Unlicensed technologies use free frequency bands, allowing an immediate use without planning and installing specialized infrastructure.

In the area of licensed technologies, 5G Mobile Radio (5G) and Narrow-Band IoT (NBIoT) are establishing themselves as transmission technologies for future use. Due to the increasing number of subscribers in mobile communication and the rapid increase in data traffic, it is becoming clear that the current 4G network will no longer be sufficient in the near future to meet requirements for the Industrial Internet of Things (IIoT) [4], [5]. 5G defines three different use cases to cope with these challenges: enhanced mobile broadband (eMBB), ultra-reliable and low latency communication (uRLLC) and massive machine-type communication (mMTC). In the context of future industrial automation scenarios, Ashraf et al. classifies industrial applications in three categories with varying latency and reliability requirements, stating that uRLLC and mMTC are of crucial importance [6]. NBIoT, another licensed technology, addresses mMTC and is a standard for low-cost devices with low data rates and high energy efficiency developed by the Third Generation Partnership Project [7]. NBIoT classifies as a Low Power Wide Area Network (LPWAN) that can transmit data over long distances, at low power, thus ensuring energy efficiency at a theoretical datarate up to 250 kbit/s with latencies of around 1-10s [8], [9].

In the area of unlicensed technologies, 802.11 and 802.15.1 are both used in consumer and industrial applications. Today's most used standard, 802.11n wireless local area network (WLAN), can operate in both 2.4Ghz ISM and 5Ghz ISM bands. In industrial applications, WLAN is used as industrial WLAN (IWLAN) in which a time-accurate, deterministic transmission as well as additional encryption is used.

Due to the rapid development and adoption of WLAN protocols, it is an interesting candidate for the IoT. However, the increased energy consumption compared to other technologies is still a limiting factor when used with embedded systems [10]. Bluetooth Low Energy (BLE) was developed from the Bluetooth specification 4.0 as an extension to the conventional Bluetooth (BT) Classic and introduced generic access profile (GAP) and generic attributes (GATT) as new concepts for data exchange and access. BLE targets small, energy-efficient devices only irregularly sending small data packets. BLE uses the 2.4 Ghz band, as does 802.11 WLAN. BLE achieves data rates of up to 2 Mbit/s and uses 40 channels with 2Mhz bandwidth of which 3 are advertising channels and 37 are data channels [11]. The adaptive frequency hopping (AFH) mechanism is used to avoid interference. The GAP concept of BLE uses connectionless advertising packets to broadcast information about device services to nearby devices with a packet length of up to 23 Bytes (255 Bytes for BLE 5). Furthermore, the GATT concept of BLE uses services, characteristics and descriptors as a basic information model that is standardized by the Bluetooth Special Interest Group (SIG). Fig. 1 gives an overview about the structure and the relations of services, characteristics and descriptors. In this structure, services are comprised of characteristics and thus group characteristics thematically. Descriptors are attached to characteristics as further annotations in order to provide additional meta-data information for the characteristic (e.g., physical units, data type presentation formats, valid ranges of values). In the consumer area, Bluetooth and WLAN are used equally often for applications. In the industrial sector, IWLAN is used more frequently than Bluetooth, but the general use of wireless transmission technologies is still low compared to the consumer sector.

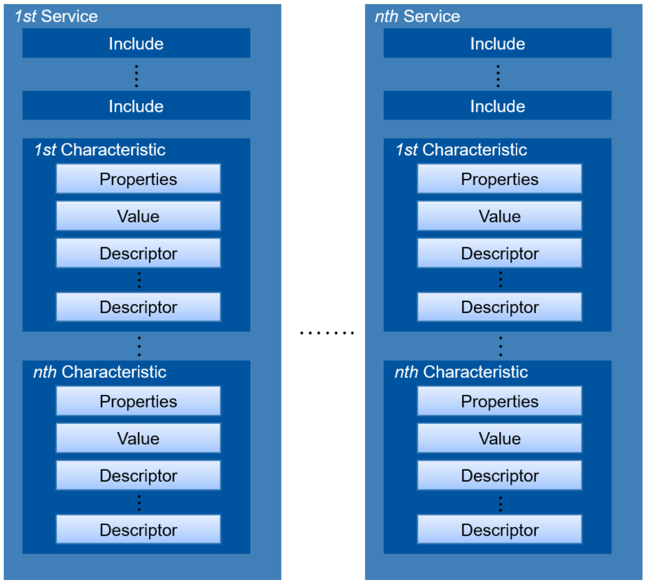


Fig. 1. Structure of BLE GATT services, characteristics and descriptors

However, the Industrial Internet of Things does not provide a wireless standard that can be used as basis. Moreover, there will be a combination of different wireless technologies that will be used in one system.

B. *Self-description of field devices*

For the integration of field devices, device description languages containing information about functionalities, communication interfaces and data addresses are used in automation technology. Device description languages have descriptive, assisting and integrative properties. Most common standards, like Field Device Tool and Device Type Manager (FDT/DTM), Electronic Device Description Language (EDDL), Open Platform Communication Unified Architecture (OPC UA) and Field Device Integration (FDI), use XML or proprietary textual representations as base for parsing and interpreting device specific information and are hence descriptive [12]–[17]. Some of these have assisting parts providing non-functional components like a representation for graphical user interfaces (GUI). Furthermore, only a smaller subset are fully integrative languages which provide frameworks and software development kits (SDK) in order to design, model, develop and execute device functionalities. OPC UA, as a representative, provides integrative capabilities and is based on different architectural layers. OPC UA servers and clients communicate through services. These forward the information model provided by the server to the client. The concept of services allows a client to access all data on the server without knowing the entire structure of the server. The technology is object-oriented, i.e. the address space consists of objects that have variables, methods and relationships to other objects called references. Through definitions of communication, address space and information model structure, services and encryption, OPC UA has grown into a very comprehensive framework. On the downside, compiled source files have a large footprint and are resource-intensive to run. Use in embedded systems is accompanied by a reduction in the overall functionality, since these systems often have low resources and low performance.

C. *Generic device control*

”Plug and Play” was introduced in 1994 as an international standard by the International Standards Association (ISA) to enable automated integration of drivers and direct use of connected devices. In analogy to this concept, ”Plug and Produce” represents the fully automated integration and configuration in industrial application scenarios [18]. The requirements for an industrial ”Plug and Produce” system can be derived from this proven technology. The following five rough steps can be identified: Discovery, Description, Control, Eventing and Presentation.

Discovery describes the behavior of devices or control points during initial or re-integration into a network infrastructure. Devices can become familiar to the network and control points can search for devices. After a checkpoint has found a device, more detailed information about it must be exchanged with the checkpoint for full functionality. The **Description** step allows the checkpoint to read and save the detailed capabilities of the device. After the checkpoint is informed about the device, it can give instructions to the device in the **Control** step. In the **Eventing** step, the device reacts to the instructions, writes status variables or sends notifications about status changes back to the control point. In the **Presentation** step, the checkpoint can finally access a type of information representation of the device [19]. These steps can be transferred into a ”Plug and Produce” system, whereby the presentation must be adapted to an industrial area of application [20].

D. *Contribution and Methodology*

In summary, it can be stated that no generic access exists for automation devices and available device descriptions specify capabilities in form of memory mappings and protocol-depended descriptions. As of today, cable-based technologies are still primarily used, even though wireless technologies possess some significant advantages (spatial invariance, mobility, ubiquitous availability, reduced infrastructure planning).

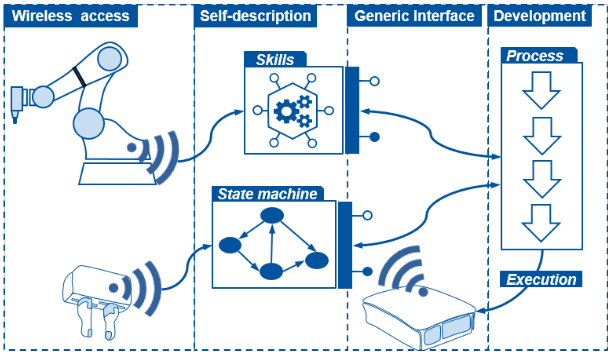


Fig. 2. Overview of approach and methodology

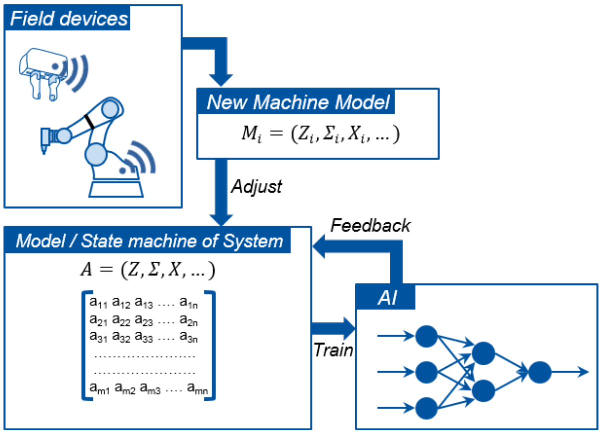
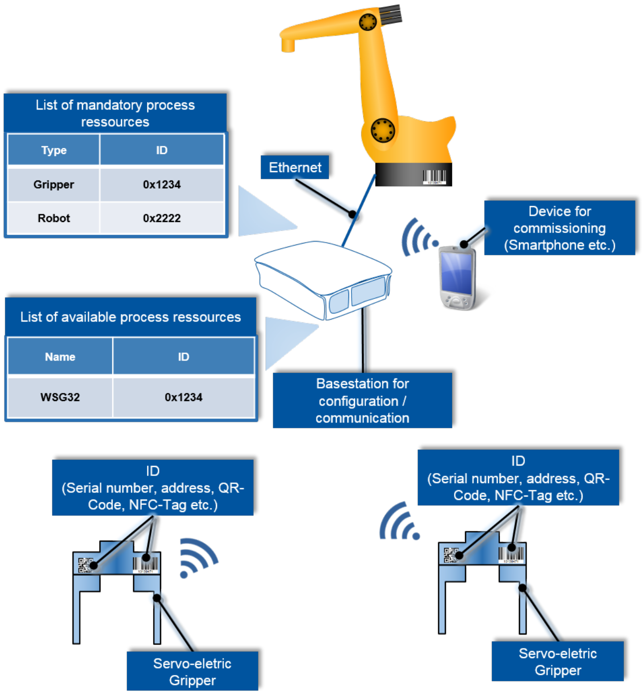


Fig. 3. Adjustment of automation systems in future production

Fig. 2 depicts the contribution and the methodology presented in this paper. The contribution consists of an approach which includes the combination of wireless communication, device self-description, generic interfaces for access and process development assistance. Automation devices expose their skills and current state using wireless communication. Using this information in combination with a suitable base station, an engineer is able to prototype automation processes by assigning skills from devices in range to process steps. Based on this, a training of a state model and an automated generation of test code of future automation systems is to be accomplished. A possible future concept is depicted in Fig. 3.

III. WIRELESS DEVICE INTEGRATION

This section introduces the concepts and methods for enabling wireless integration of automation devices. As discussed in the last section, ”Plug and Produce” is used to fully automate integration and configuration of devices to achieve immediate process execution in the industrial sector. In addition, wireless technologies allow the detection and identification of devices in the immediate environment of a controller. This allows to scan devices, check their capabilities against process requirements and control them without having to perform commissioning tasks regarding the integration in a bus system. Fig. 4 depicts the basic goal of realizing Plug and Produce paradigms combined with wireless technologies.



Type	ID
Gripper	0x1234
Robot	0x2222

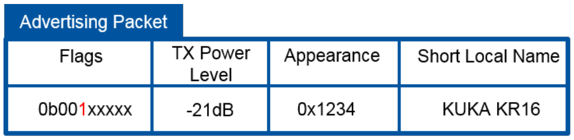
Name	ID
WSG32	0x1234

Fig. 4. Target picture for wireless ”Plug and Produce”

In the following, concepts for the self-description of device capabilities, the definition of device methods, parameters and device capabilities (e.g., is able to perform force-controlled gripping, maximum width etc.) as well as state machines are presented, which achieve this goal in interaction. Bluetooth Low Energy version 4.1 was chosen as wireless technology. Due to the adaptive frequency hopping spread spectrum method, it is more resistant to interference in the same frequency band than other common transmission methods. BLE is supported by many consumer devices and does not require a special infrastructure node to set up a network. Furthermore, the Bluetooth Special Interest Group (SIG) has standardized data types, properties, capabilities etc. in shape of services, characteristics and descriptors that are identifiable by 128-Bit long universally unique identifiers (UUID) which are used throughout many Bluetooth applications. This facilitates the definition and specification of device properties presented later.

A. Device discovery and Device skills

Enabling ”Plug and Produce” for devices requires an automatic device discovery and the use of a skill-based device description. Device discovery is achieved by broadcasting specific advertising packets. The structure of the advertising packet is shown in Fig. 5 and is comprised of a flags, a transmission power level identifier, an appearance and a short local name field. Within the flags field, the first least significant bits are already reserved by the BLE specification, hence the sixth bit is used as an indication that the advertiser is an automation device that can be integrated. The other fields are used as additional information (e.g., the appearance field can be reused to indicate what type of automation device is broadcasting its information).


Advertising Packet

Flags	TX Power Level	Appearance	Short Local Name
0b001xxxxx	-21dB	0x1234	KUKA KR16

Fig. 5. Structure of BLE advertising packet for automation devices

Skill-based device description has to be manufacturer independent and equivalent between devices with the same functionality. Skills are classified thematically, so similar abilities can be encapsulated in groups. These skill groups give a rough overview of the functionality and can be used for an initial comparison. GATT services are used to represent these. For certain device types, services can be defined as either required or optional. If services are required, a device type must implement this service to be considered a ”Plug and Produce” compatible device of its type. The unique identification of services through the use of UUIDs ensures that services are defined across devices.

Individual skills within the services / skill groups are represented by GATT characteristics. Characteristics are also identified by UUIDs. The semantic representation and abstraction of a skill by a characteristic enables the separation of a functionality and an explicit device driver implementation. Characteristics can be defined as necessary or optional. Devices of a certain type must have different basic functionality. For example, industrial grippers must be able to grip and release objects. However, moving the gripper fingers to a defined position can only be optional, as grippers without moving fingers such as vacuum grippers, exist. By comparing the existing characteristics, a device model can be built up and compared with required device profiles.

To further specify device skills with regards to their possible parameters, BLE descriptors are used. Skills can be of three different types:

- Methods to execute
- Device parameters

- Others

To differentiate between these, custom descriptors have been defined and added to the characteristics.

B. Methods, Parameter and Capabilities

Methods are represented as characteristics using descriptors to specify their parameters. Every method has at least a return parameter followed by a further input parameters. In order to identify a characteristic as an executable method, the custom descriptor 0x2918 was defined. It represents the return value of the method and consists of one field defining the data type, called format type, of the return value and a second field containing the lastly computed return value. This descriptor specifies the simplest form of a method consisting of zero input and one return parameter. Furthermore, to describe method parameters and device capabilities, it is necessary to precisely identify the type of parameter or capability. For that purpose, levels of measurement for scale types are introduced. Scale types are grouped into different types. First of all, a distinction must be made between qualitative and quantitative scale types. Qualitative ones have a lower information content because they do not have a ranking order. Quantitative ones can be compared with each other. Thus, a size relation can be established in addition to the direct comparison. In addition, characteristics can be available in a continuous and discrete form. Discrete means that they can take only a finite set of values. Continuous scale types can take on an infinite number of values within one or more defined intervals.

In total four different scale types were defined to identify the type of parameter or capability: metric continuous, metric discrete, ordinal scaled and nominal Scaled. Each method parameter or device capability uses a descriptor specifying which scale type is used to express the underlying data. The descriptor structure for each scale type is depicted in Fig. 6.

This approach enables retrospective descriptions of device skills (self-description) and can be used to infer and assure usage of correct data types.

C. State machine

At any time during the process execution, the base station must be able to map the system in its entirety. For this purpose, each device must be able to transmit its current status (e.g., idle, busy, error etc.) to the base station in order to initiate appropriate actions for a different state. For this, devices must be equipped with a finite state machine. This state machine will be stored in the information model as a necessary service for all devices.

The state machine service therefore has a single characteristic that reports the current state when it is read. This characteristic has the possibility to give feedback about state changes, hence notifications are used to achieve eventing of these changes. To provide an image of all possible states, these are added to the active state characteristic as descriptors. States are globally identified using UUIDs. Some generic states are defined, in which all devices can be. These comprise the, e.g., *uninitialized, idle* or *error* states. Furthermore, specialized states are defined depending on device type and capabilities. Fig. 7 depicts a sample of a state machine for a servo-electric gripper.

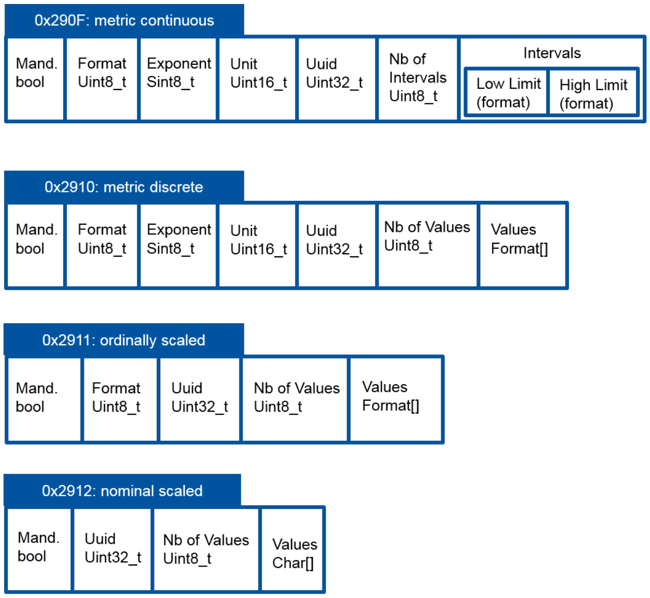


Fig. 6. Definition of different scale types for method or skill parameterization

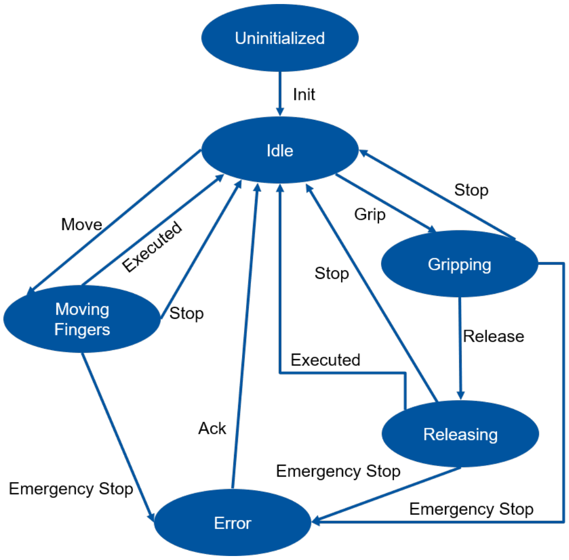


Fig. 7. Example of a state machine definition for a servo electric gripper

IV. ONLINE AND OFFLINE PROCESS DEVELOPMENT

Using the self-description concept explained above it is possible to scan for nearby devices and check device skills against developed processes. Furthermore, methods and parameters of devices can be acquired and used in order to assist engineers in their process development phase.

Based on the presented concepts of the last section a new approach is as follows:

1) Wireless-based field device recognition
2) Method, Parameter and capability device scan
3) Definition of process steps
4) Filtering of process steps according to device capabilities
5) Mapping of steps to scanned field devices
6) Wireless execution of process

A. Base station

The base station is the central node of the network architecture. It connects to the devices and controls them. The basic requirement for the base station is its ability to assume the role of a BLE master. The station contains a list of all process steps and carries out device management. It assigns individual process steps to devices and checks whether the devices are suitable for the assigned step. Furthermore, it continuously scans for new devices while the old devices are monitored for availability. Before process execution, the base station initializes devices by calling a set of initialization methods and sets device parameters. During execution, the process is monitored for failures. Furthermore, the base station provides human operators with diagnostic capabilities that indicate which device is within range and which process steps are assigned to which device. The configuration and assignment is manually adjustable by the operator if the station is not able to automatically find the optimal matching of process step to device. Finally, the station offers the operator or engineer the possibility to define and execute processes.

B. Process development

Due to the fact that information can be scanned using the approach presented, it is possible to assist the engineer or operator during the process development phase and even provide new possibilities for process development. Offline and online process development as two possible methods for process development are provided.

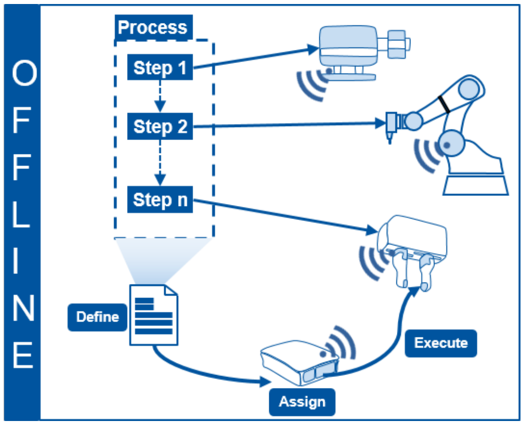


Fig. 8. Offline development of automation processes

In offline process development, the engineer defines subsequent process steps and assigns names for each step in a first phase (see Fig. 8). In this phase, direct interaction with devices is not mandatory. In a second phase, a device scan is performed and a suitable device is assigned to each step. Based on the requirements of the process step, devices are proposed. For this, device skills and process step requirements must match (e.g., when force-controlled gripping is needed for component handling). After all process steps are assigned, the overall process can be executed by the base station. In case of device replacement, the old device is automatically detached from its assigned process steps. As soon as the new device is within range of the base station, it is proposed as the device to be assigned for the process steps of the old device, if the requirements are fulfilled. If standard device parameters are stored in the base station (e.g., default gripping force, acceleration, safety-offsets etc.), it is possible to automatically override these values for the new device.

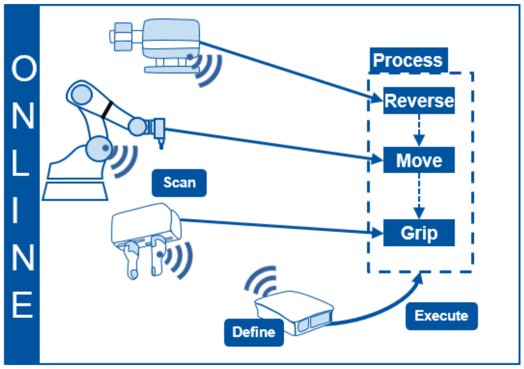


Fig. 9. Online development of automation processes

In online process development, a skill scan is initially performed for every device in range (see Fig. 9). Based on the skill scan, an engineer now selects methods from devices and defines subsequent process steps. Because a device scan was performed before the process development phase, the engineer can directly specify concrete parameters for device method calls. Additionally, default device parameter can be set live on the device. After all process steps are defined, the overall process is executed. The case of device replacement is handled in a similar way to offline development.

The matching of device skills with process step requirements decreases unintentional faults made during the process development (e.g., gripper does not provide force-controlled gripping). Device integration times can be decreased, as the system is able to automatically override default parameter values and assist the operator or engineer in the selection of appropriate devices. Furthermore, the a priori device scan used in online process development can be used to obtain an up-to-date image of built-in devices and their properties. Hence, it is not necessary to check out and search for the correct version of device specifications or descriptions in dedicated files. Finally, online process development can be used to perform rapid process prototyping, since the engineer is able to test the process directly on the plant and detect errors in the process at an early stage.

V. DEMONSTRATION

To evaluate the concepts and methods presented in the last two sections, a demonstrator as well as a sample scenario were developed. These will be explained in the following section.

A. *Technical Setup*

The implementation of the field device concepts was realized using an ESP32 and a LAN8720 ETH Board as shown in Fig. 10. The media independent interface (MII) of the ESP32 is interfaced with the media dependent interface (MDI) of the LAN8720 using a custom made printed circuit board (PCB) to provide electrical modulation of Ethernet signals. The ESP32 includes both the concrete device driver and the BLE service implementation allowing the automation device to get a wireless communication upgrade. The implementation of the base station concept was realized using a Raspberry Pi 3 Model B and an application developed in Node.js. The demonstrator consists of three devices in total. Fig. 11 shows as first device a servo-electric gripper equipped with an Ethernet port for control and status data exchange that is capable of performing force-controlled gripping commands. A second device is another servo-electric gripper equipped with a UART interface that is capable of performing gripping commands but without force-controlled feedback. Finally, Fig. 12 shows a linear axis comprised of a NEMA 17 stepper motor, a stepper motor controller and an incremental encoder with 4096 steps per revolution.

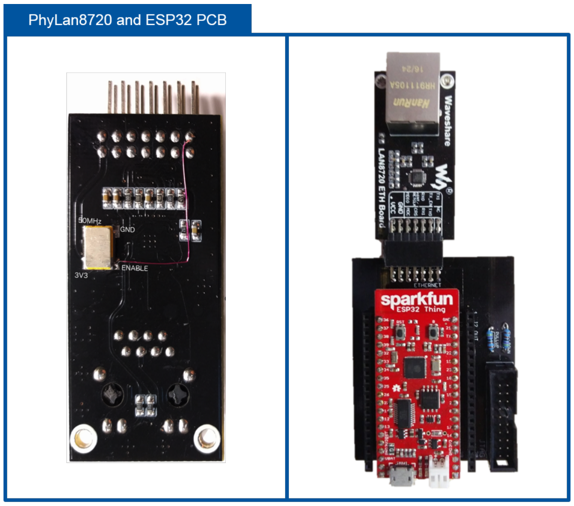


Fig. 10. LAN8720 ETH Board with attached ESP32

B. *Process development via wireless "Plug and Produce"*

The sample process consist of three process steps. First a movement command is executed in which the linear axis or a gripper moves to a specific position. Then a force-controlled gripping is performed. Finally a release command is executed. Fig. 13 shows a snippet of the offline process development overview in which device assignment is already done. The offline process development user interface allows for adding, removing and configuring process steps. Process steps are simply named and can be assigned to scanned available devices.

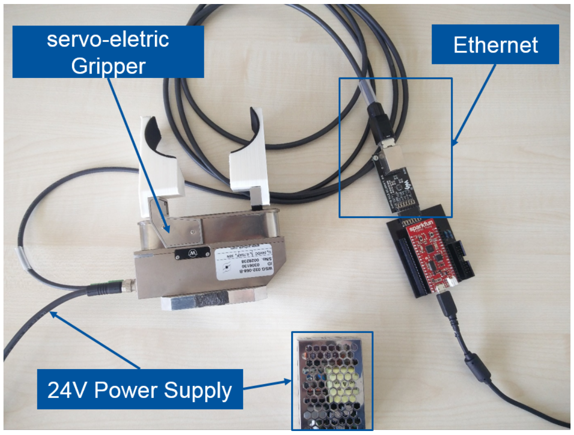


Fig. 11. Servo-electric gripper

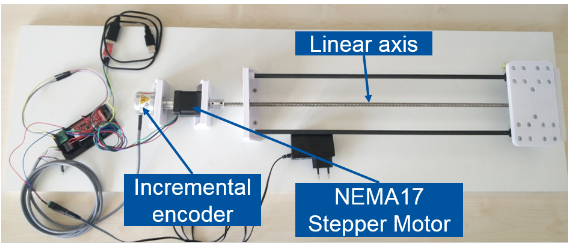
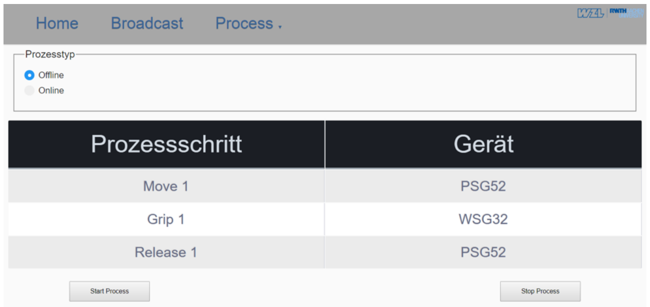


Fig. 12. Linear axis comprised of incremental encoder, stepper motor, motor controller and linear guiding unit



Prozesstyp
● Offline
○ Online

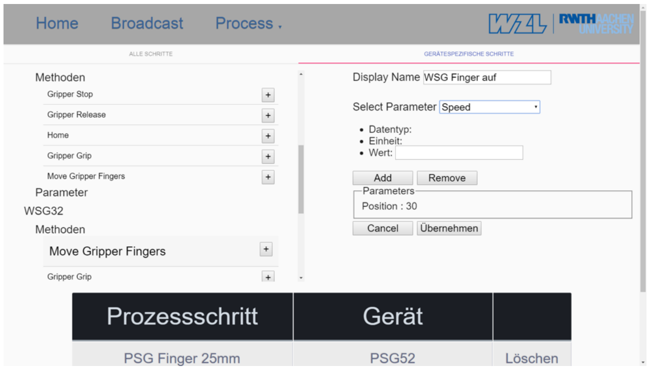
Prozessschritt	Gerät
Move 1	PSG52
Grip 1	WSG32
Release 1	PSG52

Fig. 13. GUI for offline process definition

Fig. 14 shows a snippet of the online process development in which a scanned servo-electric gripper is shown with its available methods and the current process step. This user interface allows for displaying all methods of all scanned devices as well as displaying methods related to each device. Furthermore, analogically to offline process development, it is possible to add, remove and configure process steps.

VI. CONCLUSION AND OUTLOOK

This paper presented concepts and methods for facilitating commissioning processes of automation devices and assist engineers during the process phase development based on



Display Name WSG Finger auf

Select Parameter Speed

- Datentyp:
- Einheit:
- Wert:

Parameters
Position : 30

Prozessschritt	Gerät	
PSG Finger 25mm	PSG52	Löschen

Fig. 14. GUI for online process definition

802.15.1 (Bluetooth Low Energy) and an information model for self-description of automation devices.

The approach showed new potential for industrial automation in the area of device discovery, commissioning, configuration, ramp-up processes as well as process development assistance. Skill-guided device selection based on scanned device capabilities and process requirements can fasten integration processes and development phases. Furthermore, direct interaction with automation devices and thus rapid configuration is possible due to the usage of BLE which is available in almost all smart devices used today. Due to the retrospective / self-descriptive information model of each automation device, a fast device replacement is made possible as the base station is able to map / override methods and parameters for newly integrated devices. In addition, a retrospective information model enables a base for further improvements in other fields like machine learning. As the state machine is extendable, it is be possible to train new device states based on existing ones in combination with executed methods and read parameters.

Finally, as the concept focuses on commissioning and configuring automation devices, the actual real-time data exchange has not been considered. Hence, further concepts exist in which the overall configuration is seperated into commissioning and real-time mode. In ramp-up and configuration phases, all devices use the concept presented above for commissioning purposes. As soon as a process data exchange should happen, all radio links change into real-time mode to achieve real-time requirements. Furthermore, an automated training of a state model of an automation system based on the concepts and machine learning algorithms will be investigated. Both approaches will be examined in future work.

REFERENCES

[1] D. Trotta and P. Garengo, “Industry 4.0 key research topics: A bibliometric review,” in *2018 7th International Conference on Industrial Technology and Management (ICITM)*, March 2018, pp. 113–117.

[2] L. Drkop, L. Wisniewski, S. Heymann, B. Lcke, and J. Jasperneite, “Analyzing the engineering effort for the commissioning of industrial automation systems,” in *2015 IEEE 20th Conference on Emerging Technologies Factory Automation (ETFA)*, Sept 2015, pp. 1–4.

[3] M. Wienke, O. Niggemann, S. Faltinski, and J. Jasperneite, “minadl: A novel description language enabling dynamic reconfiguration in industrial automation,” in *ETFA2011*, Sept 2011, pp. 1–4.

[4] L. B. Le, V. Lau, E. Jorswieck, N.-D. Dao, A. Haghighat, D. I. Kim, and T. Le-Ngoc, “Enabling 5g mobile wireless technologies,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, p. 36, 2015.

[5] A. Muthanna, P. Masek, J. Hosek, R. Fujdiak, O. Hussein, A. Paramonov, and A. Koucheryavy, “Analytical evaluation of d2d connectivity potential in 5g wireless systems,” in *Internet of Things, Smart Spaces, and Next Generation Networks and Systems: 16th International Conference, NEW2AN 2016, and 9th Conference, ruSMART 2016, St. Petersburg, Russia, September 26-28, 2016, Proceedings*, O. Galinina, S. Balandin, and Y. Koucheryavy, Eds. Cham: Springer International Publishing, 2016, pp. 395–403. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-46301-8_33

[6] S. A. Ashraf, I. Aktas, E. Eriksson, K. W. Helmersson, and J. Ansari, “Ultra-reliable and low-latency communication for wireless factory automation: From lte to 5g,” in *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*, IEEE Standards Association, Ed. IEEE, 2016, pp. 1–8. [Online]. Available: http://ieeexplore.ieee.org/document/7733543/

[7] A. E. Mostafa, Y. Zhou, and V. W. Wong, “Connectivity maximization for narrowband iot systems with noma,” in *2017 IEEE International Conference on Communications (ICC)*, IEEE Standards Association, Ed. IEEE, 2017, pp. 1–6.

[8] Link Labs, “Selecting a wireless technology for new industrial internet of things products: A guide for engineers and decision makers,” Annapolis. [Online]. Available: https://www.link-labs.com/selecting-a-wireless-technology-for-industrial-iot

[9] Vodafone, “Narrowband iot: Internet of things.” [Online]. Available: https://www.vodafone.de/media/downloads/pdf/vodafone-whitepaper-narrowband-iot.pdf

[10] IEEE Computer Society, “Ieee standard for information technology— telecommunications and information exchange between systems local and metropolitan area networks— specific requirements: Part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications,” New York, USA. [Online]. Available: http://www.techstreet.com/ieee/standards/ieee-802-11-2016?product_id=1867583#jumps

[11] S. Raza, P. Misra, Z. He, and T. Voigt, “Bluetooth smart: An enabling technology for the internet of things,” in *2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. Piscataway, NJ: IEEE, 2015, pp. 155–162.

[12] K. Lipinski and H. Lackner, “Fdt (field device tool),” 2009. [Online]. Available: http://www.itwissen.info/FDT-field-device-tool.html

[13] K. Bender and D. Großmann, “Fdt + edd + opc ua = fdd ua?” 2007. [Online]. Available: http://www.openautomation.de/detailseite/fdt-edd-opc-ua-fdd-ua.html

[14] EDDL, “Eddl technical description.” [Online]. Available: http://www.eddl.org/SiteCollectionDocuments/wp_EDDL-technical-description-rF.pdf

[15] ——, “Device integration made easy.” [Online]. Available: http://www.eddl.org/SiteCollectionDocuments/TechPapers/wp_EDDL device integration.pdf

[16] M. Damm, S.-H. Leitner, and W. Mahnke, *OPC Unified Architecture*. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2009. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-68899-0

[17] M. Gunzert, “Compatibility and interoperability in field device integration — a view on eddl, fdt and fdi,” in *2015 54th annual conference of the Society of Instrument and Control Engineers of Japan (SICE)*. Piscataway, NJ: IEEE, 2015, pp. 941–946.

[18] S. Hodek and J. Schlick, “Ad hoc field device integration using device profiles, concepts for automated configuration and web service technologies: Plug&play field device integration concepts for industrial production processes,” in *9th International Multi-Conference on Systems, Signals and Devices (SSD), 2012*. Piscataway, NJ: IEEE, 2012, pp. 1–6.

[19] ISO International Standards Organisation, “Informationstechnik - upnp gerätearchitektur - teil 1: Upnp device architecture version 1.0,” Berlin, 09.2011. [Online]. Available: https://www.beuth.de/de/norm/iso-iec-29341-1/146156953

[20] C. Brecher, D. Kolster, W. Herfs, M. Plessow, and S. Jensen, “Plug and play device integration for industrial automation with instant machine visualization using rfid technology,” *Production Engineering*, vol. 6, no. 2, pp. 179–186, 2012.