ITC 2/47

Journal of Information Technology and Control
Vol. 47 / No. 2 / 2018
pp. 349-362
DOI 10.5755/j01.itc.47.2.19681

A Layered Software Architecture for a Flexible and Smart Organic Rankine Cycle (ORC) Turbine – Solutions and Case Study

Received 2017/12/17 | Accepted after revision 2018/04/23

http://dx.doi.org/10.5755/j01.itc.47.2.19681

A Layered Software Architecture for a Flexible and Smart Organic Rankine Cycle (ORC) Turbine – Solutions and Case Study

Carsten Wolff, Mathias Knirr, Klaus-Peter Priebe, Peter Schulz, Jörn Strumberg
Dortmund University of Applied Sciences,IDiAL institute, Otto-Hahn-Str. 23, 44227 Dortmund, Germany,
e-mail: carsten.wolff@fh-dortmund.de

Corresponding author: carsten.wolff@fh-dortmund.de

Intelligent Technical Systems are the consequent extension of the concept of Mechatronic Systems into the world of software-intensive, networked systems. The concept enhances the functionality and variability of physical (or electro-mechanical) technical systems by adding embedded electronics and software. This adds degrees of freedom (DoF) and controllability to the technical systems. It enhances the effectivity of integrating such systems into distributed infrastructures (e.g. smart grids) via a public network like the internet. Therefore, it enables them for an effective integration into the internet-of-things (IoT) since such Intelligent Technical Systems can add and draw significant functionality and optimisation potential from networking with other systems. They can become an integral part of a distributed system which is coupled via the IoT. Nevertheless, it is important to leverage between local control and distributed control carefully according to the requirements of the technical application. A smart heating grid is an example of a distributed cyber-physical system (CPS) which can be coupled via IoT. Such systems include components like block heat and power plants which produce electricity (fed into the public smart grid) and heat. Smart heating grids benefit from components which can convert heat into electricity in a flexible and controllable way and which can change fast from heat provision to electricity provision and vice-versa. An Organic Rankine Cycle (ORC) Turbine is such a component since it converts exhaust heat into electricity. This takes heat out of the heating grid and puts electricity into the electricity grid instead. Making such an ORC turbine intelligent means optimizing it for the usage in a smart heating grid. The challenge is to design a control software architecture which allows coupling via IoT on the required interaction level of the distributed system while guaranteeing a safe operation on the local level. The article is an extended version of a previous article at the ICIST conference. It presents the software architecture of an ORC turbine based on the architecture of the Operator-Controller-Module (OCM). Compared to the previous publication it provides a more in-depth presentation of the prototype implementation of the ORC turbine system. The OCM provides an architecture pattern which allows a seamless integration into a smart heating grid based on an IoT infrastructure while enabling maximum flexibility and efficiency of the local functionality of the turbine.

KEYWORDS: Intelligent Technical Systems, Operator-Controller-Module (OCM), Smart Heating Grid, Organic Rankine Cycle (ORC) Turbine.

1. Introduction

Smart-X applications are an important domain for the market deployment of Internet-of-Things (IoT) technology [30]. For smart heating grids and smart electricity grids, a major progress is expected if the coupling is moving from proprietary standards to open protocols as with the IoT. Nevertheless – apart from general challenges in communication via public infrastructure like security, reliability and standards for interfacing [29] – technical applications like smart grids face some specific challenges which need to be addressed carefully [30]:

- “Dealing with critical latencies, e.g. in control loops”: Technical systems like the example of the ORC turbine contain control loops which operate the system and its subsystems in certain operation points which are safe, reliable and efficient. The control loop has to react in a defined time to deviations. This requires the fulfilment of certain real-time requirements. Dependent on the reaction time this cannot be fulfilled via the internet. Therefore, the system architecture has to partition the functionality into control loops with different real time requirements.
- “System partitioning (local/cloud based intelligence)”: During partitioning of the system the decision needs to be taken which functionality is implemented locally without coupling via IoT and which functionality can be implemented e.g. as a cloud service. Apart from real-time considerations this has to take into account that the technical system should be able to operate safely or at least to fail safely without IoT connection. Functionality which requires coupling via IoT or certain cloud services needs to be separated from local functionality.
- “Real-time models and design methods describing reliable interworking of heterogeneous systems. Identifying and monitoring critical system elements. Detecting critical overall system states in due time.”: The ORC turbine of our example will be embedded into such a heterogeneous system and needs to signal critical states. Furthermore, it has to react appropriately to critical states from the overall system.
- “Power grids have to be able to react correctly and quickly to fluctuations” in demand and supply: This ability to react to fluctuations is the main benefit of an intelligent ORC turbine. Only with this flexibility it becomes useful to integrate the turbine into a smart heating grid and therefore connect it to the IoT.

The following sections are dealing with the respective steps to design the software architecture of an ORC turbine according to the requirements of an operation in a smart heating grid which is coupled via IoT. This serves as an example and a blue print for the design of Intelligent Technical Systems [15] for effective and efficient integration into IoT-coupled smart systems. The example case of the ORC turbine with its specific characteristics and requirements is presented in Section 2. Section 3 summarizes the relevant related work on IoT device architectures while Section 4 presents the Operator-Controller-Module (OCM) architecture [3, 17, 20]. Section 5 describes the software and system architecture of the ORC turbine with the details of the three architectural layers in Section 6-8. Section 9 presents results from the implementation of the ORC turbine and the integration into a district heating system. Section 10 summarizes the work and draws the main conclusions. The article is an extended version of a previous article at the ICIST conference [33].

2. Case Study Organic Rankine Cycle Turbine

Organic Rankine Cycle (ORC) turbines implement a thermodynamic process which converts low temperature heat (e.g. exhaust heat from a motor) into electricity [7]. With this functionality, it serves as a valuable component in sustainable energy systems and contributes to long term strategies for mitigating global warming [22]. Nevertheless, ORC turbines are not a big market success so far [25] which is caused by several factors. The price is too high compared to the power which is generated and the flexibility in terms of reaction to the fluctuation in heat and electricity demand is too low due to the slow changes of efficient operating points in the thermodynamic processes. Therefore, we developed the ORC system [31] according to the following main requirements:

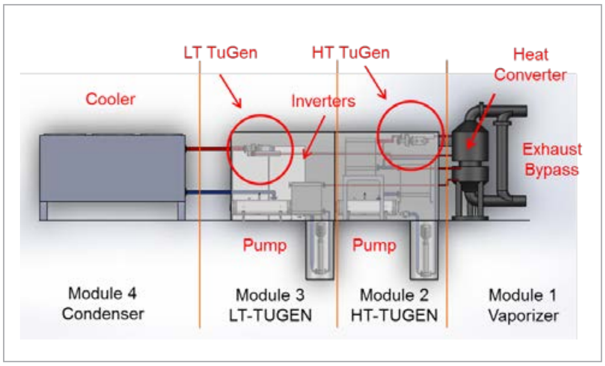
- Flexibility in electricity production (and heat usage) due to several efficient operating points for different heat sources, temperatures and electricity demands.
- Fast and robust change between the different operating points with guaranteed thermodynamic stability.
- Intelligent control and planning system with the ability to be connected to electricity and heat production control systems (e.g. smart grid) and to the overall energy system control and planning levels.
- Modular system with complete automation of the control and maintenance systems.

The main challenge is the design of an ORC turbine with several efficient operating points and a fast and robust change between the operating points. For using different temperature ranges, respective working fluids have been selected [27].

The result is a flexible, cascaded two-stage ORC turbine [8] which is based on four coupled thermodynamic modules:

- The first module contains a direct evaporator driven by exhaust gas from a waste heat source (e.g. biogas block heat and power plant) with temperature levels up to 600 °C, cooling the exhaust gas down to 75 °C.
- The second module contains the high temperature and high pressure ORC with a sophisticated turbo generator (HT TuGen). This module contains also the condenser of the working fluid, which produces heat at a level between 90 °C to 110 °C carried by a specific working fluid.
- The third module uses solar or waste heat sources in the temperature range between 90 °C and 200 °C. It contains the low temperature und low pressure ORC with a turbo generator (NT TuGen) using a grouping of nozzles for best efficiency at variable loads. Condensation energy is released on levels between 35 °C and 65 °C.
- The fourth module – if required – contains the liquefier of the working fluid of the low temperature ORC.

Figure 1
Organic Rankine Cycle (ORC) turbine [31]



The ORC turbine and the thermodynamic system were developed in a research project conducted by the authors [31].

One target application for the ORC turbine is the conversion of conventional District Heating Systems in the Ruhr-Metropolitan Area [18]. These heating grids have been supplied with heat from conventional coal and gas fired power plants or from large steel mills in the past. The conventional power plants go offline step by step. They are replaced by de-central heat sources. The university-industry-cluster ruhrvalley [32] is dealing with the development and set up of a smart heating grid for the Ruhr-Metropolitan Area – the so-called Ruhr Valley. The research on the flexible and smart ORC turbine is part of the efforts in ruhrvalley. It is a demonstrator and blue print for future smart energy systems within ruhrvalley. The IoT-based system and software architecture will be multiplied into a variety of system components. Smart combined heat and power district heating systems [10] are a major area of research for the sustainable energy system of the future. Heat (and cooling) is a bigger amount of energy than electricity in many countries. Since such systems are by nature very heterogeneous and distributed, a centralized control and steering system is not feasible. Technology based on big data and cloud computing [2] is perceived as a way of managing such systems. The Internet-of-things will play an important role in web-enabled smart grid system [6]. For the overall system architecture of the networked energy grids, the reference model for industry 4.0 (RAMI 4.0) [16] can serve as an architecture model. In ruhrvalley, based on this template, a model-driven architecture for the Internet of Things [26] is developed as a smart heating grid platform. Components like the ORC turbine have to fit into this architecture to play a beneficial role in future smart heating grids. Therefore, the ORC turbine is developed according the reference architecture model for intelligent technical systems in ruhrvalley, the Operator-Controller-Module (OCM) [3, 17, 20].

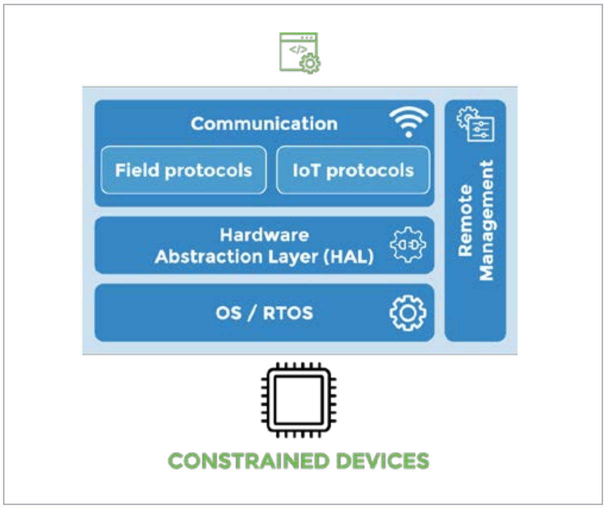
3. Architectures for Devices in IoT Systems

Within a smart heating grid, the intelligent technical system – the ORC turbine in our case study – is not only connected to the information network formed via the IoT. It is also connected with energy and material (e.g. hot water) flows via the physical (tube) network. Meaning, the smart heating grid is a "real" cyber-physical system (CPS) [1, 19] and not just a set of sensors and actuators of independent sub-systems connected via internet, but without physical connection. For such systems, the CERP-IoT [29] roadmap foresees a growing importance of the device architecture in terms of the trend "towards the autonomous and responsible behaviour of resources". Therefore, it is not sufficient to see the "device" only as an electronic IoT gateway. Devices in smart systems are smart objects or systems themselves. In many cases, they are complex mechatronic systems which are equipped with embedded devices which are connected via an IoT gateway to the cloud.

The Eclipse IoT project [9] is a major endeavour to develop an open source ecosystem (see Fig. 2) for all

Figure 2

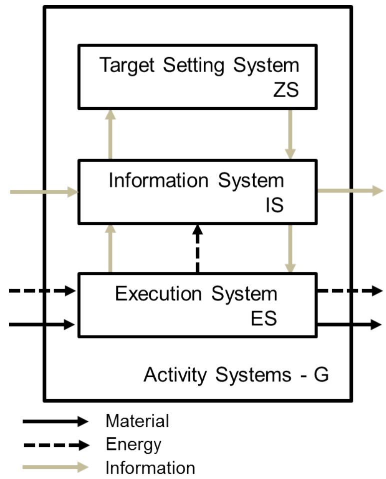
Constrained Device Architecture according to Eclipse IoT [9]



aspects of the IoT. Within its IoT architecture, the constrained device architecture plays an important role. Such IoT devices are perceived as traditional (microcontroller based) embedded systems. The IoT functionality is delivered by the implemented communication stacks. For the communication, various standards and interfaces emerged in the past years, e.g. for M2M system (ETSI oneM2M Release 2 specifications [12]), the OGC standards, e.g. the OGC SensorThingsAPI [23], and MQTT - Message Queue Telemetry Transport Protocol [21] as the dominating communication protocol for IoT applications. In automation industry, the OPC UA Unified Architecture [24] is an important standard to make programmable logic controllers (PLC) communicate with each other. This technology is typically used to connect systems like the ORC turbine to the internet or other energy systems. Nevertheless, such standards focus on data exchange with guaranteed qualities without looking deeper into the technical system. Wider approaches like the European Technology Platform on Smart Systems Integration (EPoSS) [11] strive for a system integration of the technical sub-systems. Several technical sub-systems cooperate as multi-agent systems [5] in that case. For the ORC-turbine, such standards (e.g. OPC UA and MQTT) are used, but the focus of this paper is more into the software architecture of the ORC itself then on the technologies for connecting it to the IoT. The intention is to provide a digital technology stack which separates the IoT layers from the underlying technical system by defining interfaces which make sure that the safe operation of the technical system cannot be disturbed by malfunctions or other issues on the IoT layers. To achieve a safe operation (or even a fail-safe-functionality) in case the IoT connection is interrupted or malfunctioning, the technical system has to provide an autonomous operation mode which either continues to operate the system in a stable and safe state or to ramp it down safely. In addition, the architecture has to allow tasks with different latency and real-time requirements depending on the real requirements of the specific task. This is different from systems which apply the same timing requirements to all tasks.

Figure 3

Technology stack according to Ropohl [28]



4. The Operator-Controller-Module (OCM)

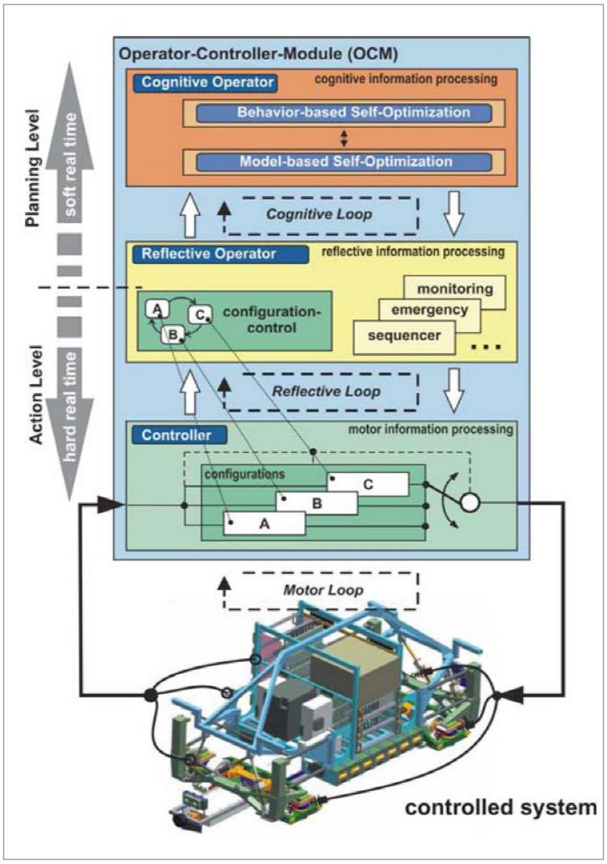
The architecture of the Operator-Controller-Module (OCM) can be connected to the very basic analysis and structuring provided by Ropohl in his description of a systems theory of technical systems (called “Technology at large”) [28]. Ropohl provides a three-layer-model (see Fig. 3) of a general activity system – which can be a technical system, any ecosystem or even a biological system. The basic execution system deals with the “things” or the interaction with the “real world”. The information system could be IT or a nervous system in biology. We can easily guess that it would be the layer connecting the things in the Internet-of-Things (IoT). The target setting system is a kind of a “brain”. In technical systems, it can be seen as the place where data analytics (in some case with “big data” methods – another buzzword) or artificial intelligence methods find their place. It can be the place where the user or the socio-economic system comes in.

The OCM technology stack [13, 14] is based on such a three layer approach (see Fig. 4) and supports the separation of controllers with hard real time requirements from more complex and strategy oriented planning and controlling tasks. On the lowest level, the motor loop is controlled by “classical” controllers. To make the approach flexible, these controllers can be configured with different parameters, that may be exchangeable according to the operation mode and the parameter or controller change may be done during operation (re-configuration). The development and the technology stack for this kind of controllers is state-of-the art. The re-configuration is done by the next layer, the reflective operator. This is a rule-based system with state machines and service functions, e.g. for emergency notification. The reflective controller is not allowed to interact directly with the technical but only via the controllers of the motor loop. With the reflective operator, the overall system can be operated and it can be connected to other systems in a network or to a smart grid. For intelligent technical systems, a more sophisticated layer for planning, reasoning and learning is added, the cognitive operator. This layer can include strategies for self-optimization and machine learning [3]. The user interaction can be added to this layer or to the reflective operator. In general, the implementation of this technology stack and the combination with a networking module turns mechatronics systems into networked mechatronic systems (NMS) [13, 14].

The OCM stack provides the required architecture for intelligent technical systems which are components in IoT based networked systems (e.g. smart grids). It is an architecture for autonomous mechatronic systems (AMS) if the OCM is used for a single,

Figure 4

Technology stack according to Operator Controller Module [13, 14]



independent system (vertical view) [3]. But it is also an architecture for networked mechatronic systems (NMS). In the OCM, systems can be connected on each of the three layers (horizontal view).

A *networking on the motor controller level* means that several technical components are interlinked by setting up a control system with coupled controllers. Such an approach is common if the controlled systems (e.g. the components of the ORC turbine) are physically connected to each other or if they are interacting. In such a case, the interaction follows real-world cause-and-effect chains which require real-time reaction of controllers and the whole control system. This causes hard real-time requirements to guarantee safe and robust operation on action level. The interlinking of technical systems on this level poses high demands on the complexity and quality of the control system design. It is difficult to develop, model and simulate such systems. The processing of the controllers has to guarantee real-time operation (which can cause high demand for computing power) and the networking requires real-time communication. This is the domain of specialized microcontroller systems (with real-time operating systems (RTOS) or bare-metal programming) and real-time programmable logic controllers (PLC) as processing devices. Field bus systems and real-time Ethernet are used for the communication. Due to the complexity and sensitivity to design errors it is recommended to interlink system functionality on that level only for tasks which really require this. It is recommended to separate all other communication tasks from the motor controllers and move them to a higher level of the OCM. The motor controllers should be single, autonomous units wherever possible. It is recommended to design them in a way which guarantees stable and safe operation in case the connection to the higher layers gets lost.

A *networking on the reflective operator level* is meeting the typical features of an IoT system. Therefore, it is recommended to use this layer to establish IoT or M2M systems. The reflective operator of the OCM can be viewed as the IoT or M2M gateway while the motor controller level is part of the IoT or M2M device. The *reflective loop* defines the interface between IoT gateway and IoT device. To connect systems on the level of the reflective operators, a message passing between the monitoring and sequencing functionalities can be implemented. This allows communication with soft real-time requirements. A coupling of the state machines on this level is possible, too, e.g. via shared memories. This allows operation with more hard real-time requirements. Therefore, the reflective operator provides very flexible option for the system design. It can be implemented on the microcontroller or PLC system of the technical device or in the cloud or anywhere in between (in the fog or edge) depending on the real-time requirements of the features implemented in the reflective operator. Therefore, the partitioning of functionality between motor controller and reflective operator has to take such considerations into account. In general, control loops should be fully contained within the single motor controller without reaching over to the reflective controller (or even other controlled systems). State machines should be part of the reflective controller.

The state machine must not to directly change settings of the controlled system but should re-configure the motor controllers which are the sole interface to the controlled system.

The *cognitive operator* should have no real-time requirements. It should plan the overall system operation well in advance. The planning will result into a sequence of timed changes of the operation points of the controlled technical system(s). This sequence is then transferred to the reflective operator and scheduled within the sequencer or a state machine. The reflective operator implements the timely changes of the operation points. Triggered by the reflective operator the motor controller(s) transitions the controlled system from one operation point to the following one. This de-coupling from the execution of the planning and therefore from the real-time processing gives the chance to apply offline-methods at the level of the cognitive operator. Plans and scenarios can be simulated and optimized. Data from other sources (e.g. pricing information for electricity from trading platforms) can be taken into account without endangering the systems operation with the resulting latencies. Data from the reflective operator can be used to train learning systems on the level of the cognitive operator. Finally, data from many controlled systems can be collected, consolidated and processed. The functionality of the cognitive operator makes it well suited for being placed in a cloud and it is straight-forward to centralize the planning level for several controlled technical systems within one cognitive operator. Due to the relaxed timing requirements and the de-coupling from safety-sensitive functionalities the cognitive operator is also a good candidate for user interaction and visualization of the system.

5. Software and System Architecture

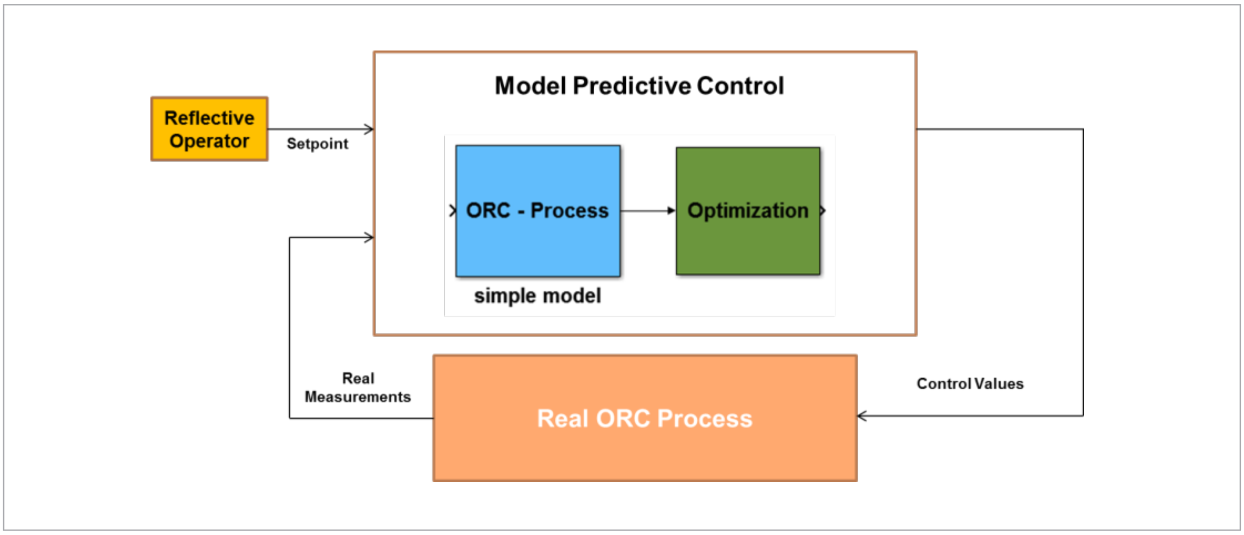
The system and software architecture for the ORC turbine is based on the 3-layer model of the OCM technology stack. The thermodynamic system (see Fig. 1) with its vaporizer, the two turbines (LT- and HT-TuGen), the pumps and tubes and the condenser plus the electrical system with the generators and the electrical converters are considered to be the controlled system. The motor loop with its controllers is based on a real time control via a real-time programmable logic controller (PLC) and operates the system autonomously around the desired operation points. The reflective operator implements the state machines for driving the machine from one operation point to another operation points (and from power on to operation and again back to power off). Furthermore, it implements the coupling and synchronization to the electric grid and the service, monitoring and alarm function via the IoT and up to the planning layer. The cognitive operator plans the operating sequence of the ORC turbine system based on the demand for heat and electricity. The controller knows the capabilities of the ORC system and may plan and optimize e.g. daily sequences of operation modes. It can take maintenance intervals into account and it can use self-optimizing and learning strategies. The cognitive operator is connected to smart power grid and smart heating grid interfaces and to the overall optimization instance of the regional energy system. On the different layers of the system, different architectural concepts, description of components and languages are deployed. The implementation is done on a distributed PLC and Industry-PC system and connected via IoT technology to an IoT cloud [26].

6. Motor Controller Loop

The controllers of the motor loop are developed in a model-based development approach by setting up a mathematical model of the controlled systems and by developing optimized controllers which are simulated with the model and later with the real system (Hardware in the Loop – HiL). They are basically data driven continuous systems which need to react real-time to any change of the system.

The Model Predictive Control (MPC) is used for the controllers [31]. In such a system (see Fig. 5), the (simplified) mathematical model of the thermodynamic and electro-mechanical processes of the ORC turbine (controlled system) are simulated online at any time. With this model, the controller can calculate (predict) what actuator values need to be set to achieve a certain system behaviour based on the current state of the system (which is signalled by the sensors). Furthermore, based on the model, the optimum placement of the actuators and sensors can be derived and the dynamics of the controlled system can be

Figure 5
Model Predictive Control (MPC) approach for the motor control loop

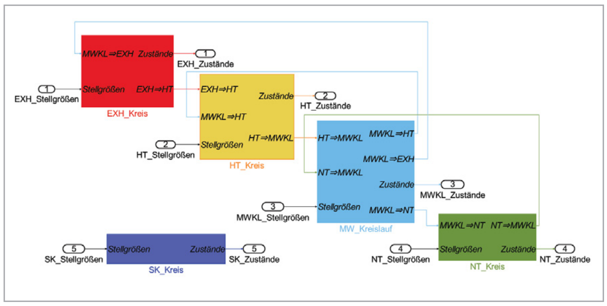


simulated. The MPC approach is quite powerful but consumes a lot of processing power. Therefore, it is better suited for slower processes like thermodynamic processes. For the high speed supersonic turbine, a faster independent control loop is implemented. The controllers are initiated and started by the state machines of the reflective controller. They read their parameters from a shared memory controlled by the reflective controller. All controllers and the system models are described and validated in Matlab/Simulink and deployed via automatic code generation into C-Functions of the PLC [31].

Due to the complexity of the model for the ORC process it was necessary to simplify system model and to break it down into sub-models (see Fig. 6). The breakdown follows the modules of the technical ORC system (see Fig. 1) and therefore the coupling parameters of the 5 sub-models (see Fig. 6) are the same thermodynamic and mechanical parameters as in the real system.

Since thermodynamic systems are rather "slow" technical systems with a latency and flexible coupling of the sub-processes, the coupling of the sub-models via these parameters results in an overall model which is "simulation friendly" and "processing friendly". The system model is validated with real data from the ORC turbine system and verified in terms of stability and safe transition from one operation point to anoth-

Figure 6
ORC system model broken down and simplified into 5 coupled system models



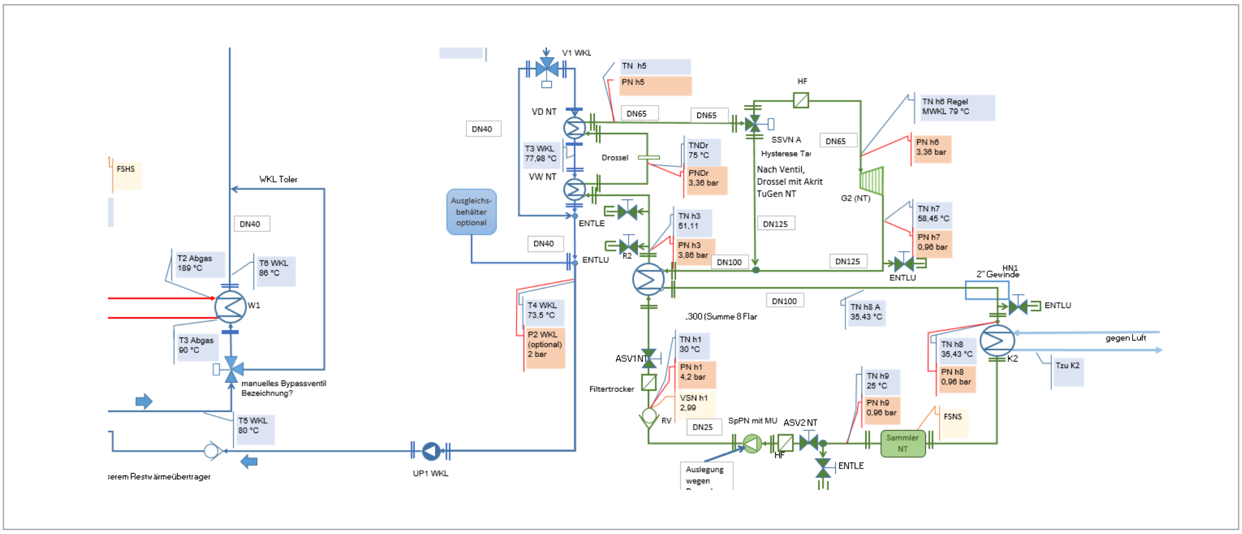
er. Furthermore, the system model is used to define where to put sensors and actuators in the real ORC turbine system. It can also provide information about the accuracy of measurement and the timing requirements of measurement data (e.g. sampling).

7. Reflective Controller Loop

The main functionality of the reflective controller loop is implemented into several state machines and interrupt service functions. It follows a control driven and an event driven paradigm. The functionality is described in CoDeSys which is a domain specific language (DSL) for state charts and event driven systems

Figure 7

Thermodynamic Circuit diagram of the ORC turbine with coupling points between reflective controller and motor controller loops



[4]. The main state machines for ramping the system up (to power up), ramping it down (to safe off) and operating it with changing operation points (see Fig. 7) are coupled via a central shared memory (called SVI).

The SVI contains all states of the system and all relevant values for sensors and actuators. Furthermore, it is used to set the operation points for the controllers. The data structure of the SVI is automatically generated from a master system scheme (see Fig. 8) which contains the main thermodynamic circuits and defines the coupling points. This master system scheme defines all parameters, their permitted range and accuracy and the threads of the ORC software which are allowed to read it or to modify it. This is stored in a table which allows an easy and simple overview. Modifications are done in the table, not in the code of the

Figure 8

Master system scheme of the ORC system with all relevant parameters of the SVI

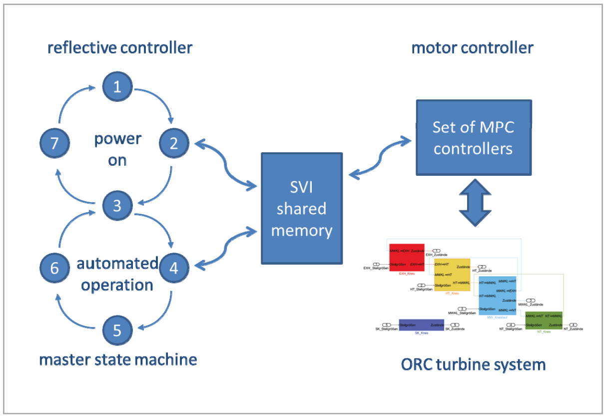
1	Revisionsstand			Legende:	Rot hinterlegte ID-Felder (Spalte B) deuten auf einen zu prüfenden Datensatz in der selbigen Zeile hin. Die entsprechenden Zellen sind durch einen Kommentar und ggf. farblich zu markieren.								
2											21.10.2015		
3	Inhalt												
4		Sensor-/Aktorbezeichnung											
5	SVI-Nr.	Typus		Bezeichnung	AUTOSAR-Bez.	prüffähige Bez.	Herkünfte	Kommentar	Modu	Regelkreis	ADIO/BUS	Sensortyp	Skalierun
6													
7	RK Abgas												
22	111	chk		AKM1 Winkel Ist	AgExhAKM1	AG-CV-AG01/03 Winkel Ist	Iec, Web	Rückmeldung Öffnungswinkel	1	RK Abgas	AI	20	
23	112	sp		AKM1 Winkel Soll	SpAgExhAKM1	AG-CV-AG01/03 Winkel Soll	Iec, Smink, Web	Stellglied Öffnungswinkel	1	RK Abgas	AO	21	
40													
41			AKab	nach ORC zum Bypass									
42	113	chk		AKM2 Endlage vom ORC geschlossen	FlgExhColdBy	AG-CV-AG02 Bypass	Com, Iec	vom ORC geschlossen (Bypass)	1	RK Abgas	DI	0	
43	114	chk		AKM2 Endlage vom ORC offen	FlgExhColdRun	AG-CV-AG02 Betrieb	Com, Iec	vorm ORC geöffnet (Betrieb)	1	RK Abgas	DI	0	
44													
45	115	sp		AKM2 AufZu	FlgExhAKM2Ena	AG-CV-AG02 AufZu	Iec, Web	Druckluftventil	1	RK Abgas	DO	0	
46				AKM2 Endlage Druckluftventil	FlgExhAKM2	AG-CV-AG02 Druckluftvers.	Iec, Web	Druckluft ja – nein	1	RK Abgas			
47			AKM3 Ventilator Rückführung mit FU										
48	117	sp		AKM3 Schuetz FU	SpFlgExhAKM3	V-G1 Schuetz FU	Com, Iec	Aktivieren des FU	1	RK Abgas	DO	0	
49	118	sp		AKM3 Drehzahl Soll	SpNExhAKM3	V-G1 Drehzahl Soll	Iec, Smlnk, Web	Stellglied Drehzahl	1	RK Abgas	AO	22	
50	119	sp		AKM3 Drehrichtung	SpFlgExhAKM3RotDir	V-G1 Drehrichtung		DI 2 – Vorwärts/Rückwärts			DO	0	
51				AKM3 Konstantdrehzahl 1	SpFlgExhAKM3ConSpd1			DI 3 – Auswahl konstantdrehzahl					
52				AKM3 Konstantdrehzahl 2	SpFlgExhAKM3ConSpd2			DI 4 – Auswahl konstantdrehzahl					
53				AKM3 Rampenauswahl	SpFlgExhAKM3Ramp			DI 5 – Rampenauswahl					
72													
73	120	chk		zweifach Prandlsonde	MFlowExhMix	Pitotsonde	IEC, Smink, Web	Massenstrom Abgas nach Tmix	1	RK Abgas	AI	12	
74	121	chk		Temperatursensor Prandlsonde	TMFlowExhMix				1	RK Abgas	AI	51	
75			Ventilator Schornstein mit FU										
76	124	sp		Vent-Schornst Schuetz FU	SpFlgExhVentChim	V-G2 Schuetz FU	Com, Iec	Aktivieren des FU	1	RK Abgas	DO	0	
77	125	sp		Vent-Schornst Drehzahl Soll	SpNExhVentChim	V-G2 Drehzahl Soll	Iec, Smlnk, Web	Stellglied Drehzahl	1	RK Abgas	AO	23	
78	126	sp		Schornstein Drehrichtung	SpFlgExhVentChimRotDir	V-G2 Drehrichtung		DI 2 – Vorwärts/Rückwärts			DO	0	
79				Schornstein Konstantdrehzahl 1	SpFlgExhVentChimConSpd1			DI 3 – Auswahl konstantdrehzahl					
80				Schornstein Konstantdrehzahl 2	SpFlgExhVentChimConSpd2			DI 4 – Auswahl konstantdrehzahl					
81				Schornstein Rampenauswahl	SpFlgExhVentChimRamp			DI 5 – Rampenauswahl					
82	.....												

PLC or the controllers. Therefore, the master system scheme serves as design documentation, too.

The MPC controllers read their parameters from the SVI (see Fig. 9) but can operate the system safely afterwards independently. The access of the MPC is a read-only access for all sensor values since sensor values are collected and written to the SVI by specific sensor data collection threads. The write access to the actuator values of the SVI is only allowed for the MPC, not for the state machines of the reflective operator. The SVI implementation allows setting of the respective access rights for each individual parameter based on the defined access patterns. The parameters in the SVI are monitored and alarm is generated if one of the parameters leaves the defined range. In that case, the MPC ramps the ORC turbine automatically down to a safe state.

Figure 9

Coupling of state machines and controllers via SVI shared memory



The SVI implements connectivity via OPC-UA [24], too. For this purpose, a set of parameters in the SVI is opened for read or write access to the OPC-UA server which can be run on the PLC. This allows access via the OPC-UA protocol and transfer of data to and from the IoT gateway component.

8. Cognitive Controller – Planning Level

The cognitive controller implements the integration of the ORC turbine into a biogas power plant (which supplies the exhaust heat via its block heat and power plant into the vaporizer of the ORC system, see Fig. 10) and into a smart heating grid providing heat to residential areas and industry. The cognitive controller is partly implemented on an industrial PC and partly within an M2M cloud solution [26]. The M2M cloud solution provides online access and monitors all operation data since the ORC system is a research prototype where the data are needed for evaluation and validation. Furthermore, the cognitive operator implements the user interface which is formed by a cockpit visualization of the ORC system scheme. The cockpit does not only allow monitoring all relevant parameters while the ORC system is under operation. It also allows manual setting of operation points or sequences of operation points. This is required for the test operation and the validation and verification of all relevant steps. Furthermore, the functionality is needed for the auditing of the system where certain system states have to be set according the checks done by the auditor. The cockpit can be operated at the control panel of the ORC turbine cabinet (see Fig.

Figure 10

Vaporizer of the ORC turbine prototype

12) or via the web interface of the cloud system.

The industrial PC implements the operating strategy of the biogas power plant and the smart heating grid. It provides the ORC system with the required information about how much heat and how much electricity is needed at certain times of the day. The cognitive controller will be extended with simulation and optimization components (a learning system for optimized heat and power production planning) in future.

9. Implementation on a Biogas Power Plant

The MPC motor controllers and the main parts of the reflective controller are implemented on a Bachmann control system [4] which is tailored to the requirements of renewable energy systems (see Fig. 12). It includes the required real-time PLC functionality for the MPC and for some additional C++ functionality. It connects to the high speed controllers of the super-sonic turbo generators (see Fig. 11) via CAN-Bus. Bachmann provides the functionality for the SVI shared memory, too. Other features of the reflective operator are implemented within the industrial PC part of the Bachmann system. This allows the connection to OPC UA Unified Architecture [24]. The connection to the cognitive operator is done via the Bachmann WebMI which connects to the SVI, too.

10. Conclusion and Future Work

All components of the technology stack (see Fig. 13) have been tested on a Bachmann laboratory installation. The ORC system prototype is currently under test at a biogas power plant in Germany. The test operation allowed auditing and test for gaining the public permit for a continuous test operation. With the granting of the required public permits in October and November 2017, the ORC turbine system is allowed to feed the generated electricity into the public grid. If operated on full power, the system is able to provide electricity for 150 (average) households (consuming 3000 kWh/year).

The ORC system and software architecture will be evaluated and validated during the coming 12 month.

Figure 11

Low Temperature (LT) Turbo Generator

Figure 12

Cabinet with Bachmann PLC. Actuator (Drives) and Sensor Electronics and Power Converters

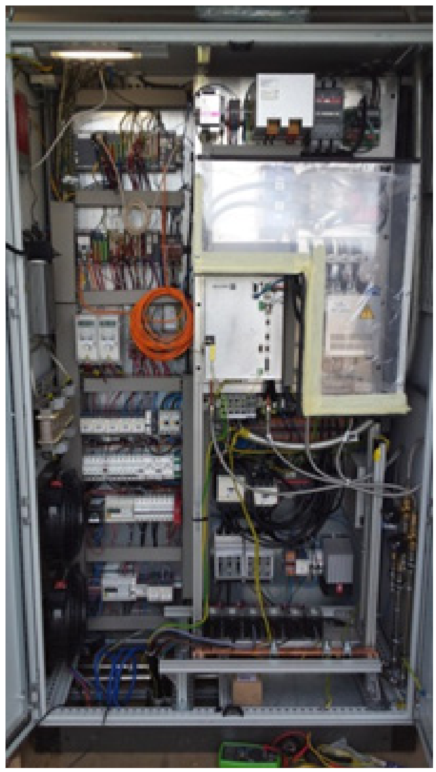
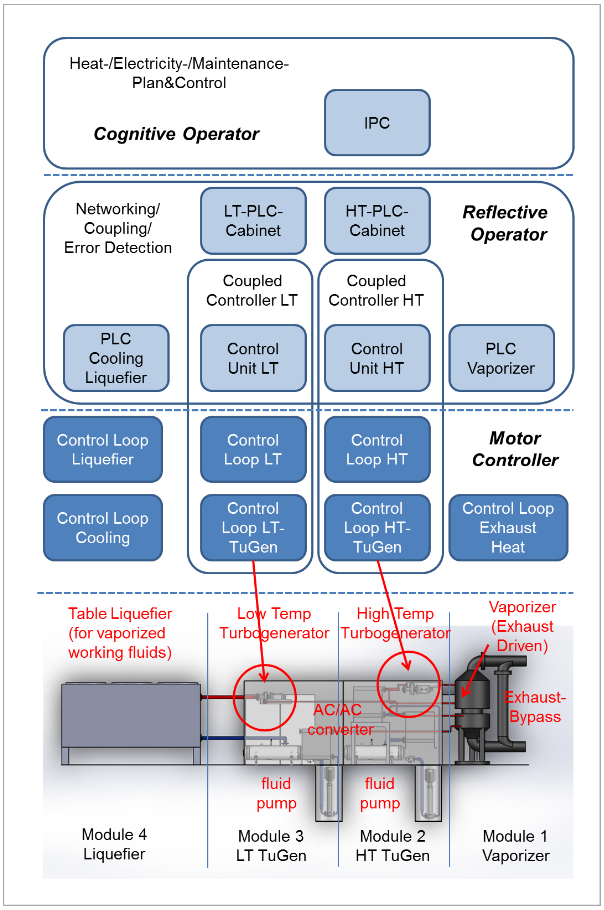


Figure 13

Implemented technology stack of the ORC turbine system



The aim is to generate more than 75 kW of electrical power from a 650 kW thermal exhaust source (see Table 1) which is an outstanding efficiency (see [7]). Afterwards, the full integration into the ruhvalley smart heating grid platform will be done (see Section 2). The described system and software architecture allows coupling via IoT into smart systems based on the functionality of the reflective operator and/or the cognitive operator. Therefore, it enables system integration beyond traditional systems which are just extended with an IoT gateway. It will serve as a blue print for the development of other intelligent technical systems as components in smart systems, specifically within the context of ruhrvalley.

Table 1

Performance parameters of the ORC demonstrator system with a two stage turbine system (high-temperature (HT) turbine and low-temperature (LT) turbine)

Estimation of Electrical Power		
Usable Heat Energy HT	kJ/s	245,69
Vaporizer Temperature	K	495
Condenser Temperature	K	308
Mechanical Power (Turbine)	kW	66,30
Usable Heat Energy LT	kJ/s	365,55
Vaporizer Temperature	K	85
Condenser Temperature	K	30
Mechanical Power (Turbine)	kW	30,78
Electrical Power HT (net)	**kW**	**53,673**
Electrical Power LT (net)	**kW**	**21,522**
Electrical Power total (net)	**kW**	**75,19**

References

1. Acatech AgendaCPS: Integrierte Forschungsagenda Cyber Physical Systems, 2012.

2. Agrawal, D., Das, S., El Abbadi, A. Big Data and Cloud Computing: Current State and Future Opportunities. Proceedings of the 14th International Conference on Extending Database Technology, ACM, 2011, 530-533. https://doi.org/10.1145/1951365.1951432

3. Adelt, P., Donoth, J., Gausemeier, J., Geisler, J., Henkler, S., Kahl, S., Klöpper, B., Krupp, A., Münch, E., Oberthür, S., Paiz, C., Podlogar, H., Porrmann, M., Radkowski, R., Romaus, C., Schmidt, A., Schulz, B., Vocking, H., Wittkowski, U., Witting, K., Znamenshchykov, O. Selbstoptimierende Systeme des Maschinenbaus – Definitionen, Anwendungen, Konzepte. HNI-Verlagsschriftenreihe, Band 234, Paderborn, 2008.

4. Bachmann M1 Automation System: http://www.bachmann.info/branchen/erneuerbare-energien/, last accessed November 20, 2015.

5. Bellifemine, F. L., Caire, G., Greenwood, D. Developing Multi-Agent Systems with JADE.

John Wiley & Sons, Hoboken, 2007. https://doi.org/10.1002/9780470058411

6. Bui, N., Castellani, A. P., Casari, P., Zorzi, M. The Internet of Energy: A Web-Enabled Smart Grid System. IEEE Network, 2012, 26(4), 39-45. https://doi.org/10.1109/MNET.2012.6246751

7. Chen H., Goswami D. Y., Stefanakos E. K. A Review of Thermodynamic Cycles and Working Fluids for the Conversion of Low-Grade Heat. Renewable and Sustain-able Energy Reviews, 2010, 4, 3059-3067. https://doi.org/10.1016/j.rser.2010.07.006

8. Dubberke, F. H., Priebe, K.-P., Vrabec, J., Roedder, M., Neef, M. Thermodynamic Simulation and Experimental Validation of a Cascaded Two-Stage Organic Rankine Cycle Turbine. 3rd International Seminar on ORC Power Systems, Belgium, 2015.

9. Eclipse IoT Constrained Devices https://iot.eclipse.org/devices/, last accessed March 10, 2017

10. Elci, M., Oliva, A., Herkel, S., Klein, K, Ripka, A. Grid-Interactivity of a Solar Combined Heat and Power District Heating System. International Conference on Solar Heating and Cooling for Buildings and Industry (SHC2014), 2014, 560-567.

11. EPosSS Strategic Research Agenda of The European Technology Platform on Smart Systems Integration, Version 2, 2014, available at www.smart-systems-integration.org

12. ETSI oneM2M Release 2 Specifications http://www.onem2m.org/technical/published-documents, last accessed March 10, 2017.

13. Gausemeier, J., Steffen, D., Donoth, J., Kahl, S. Conceptual Design of Modularized Advanced Mechatronic Systems. 17th International Conference on Engineering Design (ICED`09), Stanford, CA, USA, 2009, 263-274.

14. Gausemeier, J., Kahl, S. Architecture and Design Methodology of Self-Optimizing Mechatronic Systems. In: Mechatronic Systems Simulation Modeling and Control, InTech, 2010, 255-282. https://doi.org/10.5772/9121

15. Gausemeier, J., Rammig, F. J., Schäfer, W. (Eds.) Design Methodology for Intelligent Technical Systems – Develop Intelligent Technical Systems of the Future, Springer, 2014. https://doi.org/10.1007/978-3-642-45435-6

16. Hankel, M., Rexroth, B. Industrie 4.0 Das Referenzarchitekturmodell Industrie 4.0 (RAMI 4.0), Zentralverband Elektrotechnik und Elektronikindustrie e.V., 2015.

17. Hestermeyer, T., Oberschelp, O., Giese. H. Structured Information Processing For Selfoptimizing Mechatronic Systems. In: Proceedings of 1st International Conference on Informatics in Control, Automation and Robotics, ICINCO (3) 2004, Setubal, Portugal, 2004, 230-237.

18. Knutzen, L. K., Stolpe, H., Bracke, R., Beese, E. Conversion of Conventional District Heating Systems in the Ruhr-Metropolitan Area, Western Germany, by Stepwise Integration of Geothermal Sources. Proceedings World Geothermal Congress, 2015.

19. Lee, Edward A. Cyber Physical Systems: Design Challenges. International Symposium on Object/ Component/ Service-Oriented Real-Time Distributed Computing, 2008, 363-369.

20. Lückel, J., Hestermeyer, T., Liu-Henke, X. Generalization of the Cascade Principle in View of a Structured Form of Mechatronic Systems. 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2001), Como, Italy, 2001, 123-128. https://doi.org/10.1109/AIM.2001.936441

21. MQTT – Message Queue Telemetry Transport Protocol http://www.ibm.com/developerworks/webservices/library/ws-mqtt/index.html, last accessed March 10, 2017.

22. Nakicenovic, N., Grubler, A., Inaba, A., Messner, S., Nilsson, S., Nishimura, Y., Rogner, H.-H., Schafer, A., Schrattenholzer, L., Stubegger, M., Swisher, J., Victor, D., Wilson, D. Long Term Strategies for Mitigating Global Warming. Energy, 1993, 5, 18, 401-609. https://doi.org/10.1016/0360-5442(93)90019-A

23. OGC, OGC SensorThingsAPI: http://www.opengeospatial.org/standards/sensorthings, last accessed March 15, 2017.

24. OPC-UA, OPC Unified Architecture (UA), 2008. https://opcfoundation.org/about/opc-technologies/opc-ua/, last accessed March 15, 2017.

25. Quoilin S., Van Den Broek M., Declaye S., Dewallef P., Lemort V. Techno-Economic Survey of Organic Rankine Cycle (ORC) Systems. Renewable and Sustainable Energy Reviews, 2013, 22, 168-186. https://doi.org/10.1016/j.rser.2013.01.028

26. Rademacher, F., Peters, M., Sachweh, S. Model-Driven Architecture for the Internet of Things. 41st Euromicro Conference on Software Engineering and Advanced Applications, Madeira, Portugal, 2015.

27. Roedder, M., Neef, M., Laux, C., Priebe, K.-P. Systematic Fluid Selection for Organic Rankine Cycles (ORC) and Performance Analysis for a Combined High and Low Temperature Cycle. Proceedings ASME Turbo Expo

2015: Turbine Technical Conference and Exposition, Montreal, Canada, 2015.

28. Ropohl, G. Allgemeine Technologie, Eine Systemtheorie der Technik, 3. überarbeitete Auflage, Universitätsverlag Karlsruhe, 2009. https://doi.org/10.26530/OAPEN_422388

29. Vermesan, O., Friess, P., Guillemin, P., Gusmeroli, S. et al. (Eds.) Internet of Things Strategic Research Roadmap, Cluster of European Research Projects on the Internet of Things, CERP-IoT, 2011.

30. Vermesan, O., Friess, P. (Eds.) Internet of Things – From Research and Innovation to Market Deployment, River Publishers Series in Communications, 2014.

31. Wolff, C., Knirr, M., Schulz, P., Priebe, K.P., Strumberg, J., Vrabec, J. Flexible and Controllable ORC Turbine for Smart Energy Systems. Proceedings of the IEEE EnergyCon 2016, IEEE Xplore, Leuven, Belgium, 2016. https://doi.org/10.1109/ENERGYCON.2016.7514055

32. Wolff, C., Telgmann, L. Systems Engineering for Metropolitan Mobility and Energy – RuhrValley. Proceedings of the Dortmund International Research Conference, 2016, 141-153, ISBN 978-3-00-055144-4.

33. Wolff, C., Knirr, M., Pallwitz, T., Igci, H., Priebe, K.-P., Schulz, P., Strumberg, J. Software Architecture for an ORC Turbine – Case Study for an Intelligent Technical System in the Era of the Internet of Things. 23rd International Conference on Information and Software Technologies (ICIST 2017), Springer CCIS, Druskininkai, Lithuania, 2017, 226-237.