D6.6.31 Enhanced prototype of EMS decision-making component

Description of the Enhanced Simulation Model for the Microgrid's Energy Management System

Revision History

Edition	Date	Status	Editor
v0.01	10.12.2013	Created	Pekka Savolainen
v0.05	27.02.2014	Added Simulink model	Jouni Hiltunen
v0.06	28.2.2014	Added MGC & MG model chapters	Riku Pasonen, Pekka Savolainen
v1.0	28.2.2014	Final adjustments	Pekka Savolainen

Authors: Pekka Savolainen (VTT), Jouni Hiltunen (VTT), Heli Kokkoniemi-Tarkkanen (VTT), Riku Pasonen (VTT)

Abstract

This deliverable describes the work done in SGEM-project Phase 4, also known as funding period four (FP4). The work done in FP4 is a direct continuation to the work in SGEM funding periods two (FP2) and three (FP3). In deliverable D6.12.7 "*Survey Report on KPIs for the Control and Management of Smart Grids*" [1], written in FP2, an idea of a microgrid with an Energy Management System (EMS) was introduced and defined. Deliverable D6.6.21 "*A Prototype of an Intelligent Decision-making Component for Smart Grids*" [2], written in FP3, contains the description for a Simulink/Matlab simulation model for the microgrid's Energy Management System (EMS) and microgrid model containing various generation, storage and load devices.

In the fourth funding period, work has concentrated on building a completely new microgrid simulation platform, which will allow us to test different EMS algorithms in a much more realistic model than what was possible with the Simulink microgrid model implemented in FP3. The new platform combines VTT's expertise from electrical and telecommunications side and it is being developed in co-operation with SGEM task 5.1. This deliverable also describes the updates done in the Simulink microgrid simulator.

Table of Contents

1 Preface

This report was written as a part of the Finnish national research project "Smart Grid and Energy Market" SGEM Phase 4 and it was funded by Tekes – the Finnish Funding Agency for Technology and Innovation and the project partners.

2 Introduction

This deliverable describes the work done in SGEM-project Phase 4, also known as funding period four (FP4). The work done in FP4 is a direct continuation to the work in SGEM funding periods two (FP2) and three (FP3). In deliverable D6.12.7 "*Survey Report on KPIs for the Control and Management of Smart Grids*" [1], written in FP2, an idea of a microgrid with an Energy Management System (EMS) was introduced and defined. Deliverable D6.6.21 "*A Prototype of an Intelligent Decision-making Component for Smart Grids*" [2], written in FP3, contains the description for a Simulink/Matlab simulation model for the microgrid's Energy Management System (EMS) and microgrid model containing various generation, storage and load devices.

In FP4, the work has concentrated on building a completely new microgrid simulation platform, which will allow us to test different EMS algorithms in a much more realistic model than what was possible with the Simulink microgrid model implemented in FP3. The new platform combines VTT's expertise from electrical and telecommunications side and it is being developed in co-operation with SGEM task 5.1. Some components, namely the APROS made microgrid controller and the new microgrid model, in the new simulation platform are documented in detail in a deliverable prepared in SGEM task 5.1.

However, because the new microgrid simulation platform is still in development phase, we needed to update the Simulink microgrid model done in FP3 as well. This enables us to continue our work with the EMS algorithm, while the new APROS models are still in development. The Simulink model also helps us in testing the new EMS algorithm. The description for the updated Simulink microgrid model is found in chapter 6.

3 Selected High Level Use Cases

SGEM task 5.1 deliverable D5.1.46 [5] lists use cases for microgrid simulations. The high level use cases listed there are mainly based on a use-case study for microgrids published by the FINSENY project [4]. The use cases in those documents have been divided into two categories; Control & Management use cases and Business use cases. The microgrid simulation platform will be capable of simulating the following use cases:

Control & Management Use Cases:

- Balancing supply and demand on different time-scales
- Demand-side management
- Supply-side management
- Storage management
- Planning

Business Use Cases:

- Microgrid operator sells/buys energy on external market

Resources available in this project will not allow us to simulate and analyze all of the above use cases so we are forced to concentrate on just one or two.

4 Microgrid Simulation Platform

Microgrid simulation platform consists of a microgrid model, a microgrid controller, an energy management system (EMS), and a smart grid telecommunications emulator. The microgrid model and controller have been implemented with APROS, which is an electricity network and component simulation tool designed by VTT. Figure 1 depicts the simulation platform, which consists of three host computers; Host A, Host B and Host C. Hosts B and C are virtual machines running Windows 7 as their operating system. The virtual machines are running on the same server computer. Host A can be a regular Windows 7 computer e.g. a laptop. Host A is running the energy management system, which is implemented in C++. Host B is running the microgrid controller implemented as an APROS model. Host C is running the microgrid model as its own APROS model. The hosts are connected via an OPC UA client-server protocol. OPC is an open standard for data transmissions, used frequently in industrial automation to integrate control and diagnostic applications to process devices [3]. OPC Data transmission is based on client-server architecture and through a standard interface. OPC UA (OLE for Process Control – United Architecture) is a new definition, which aims to combine earlier separate definitions of OPC.

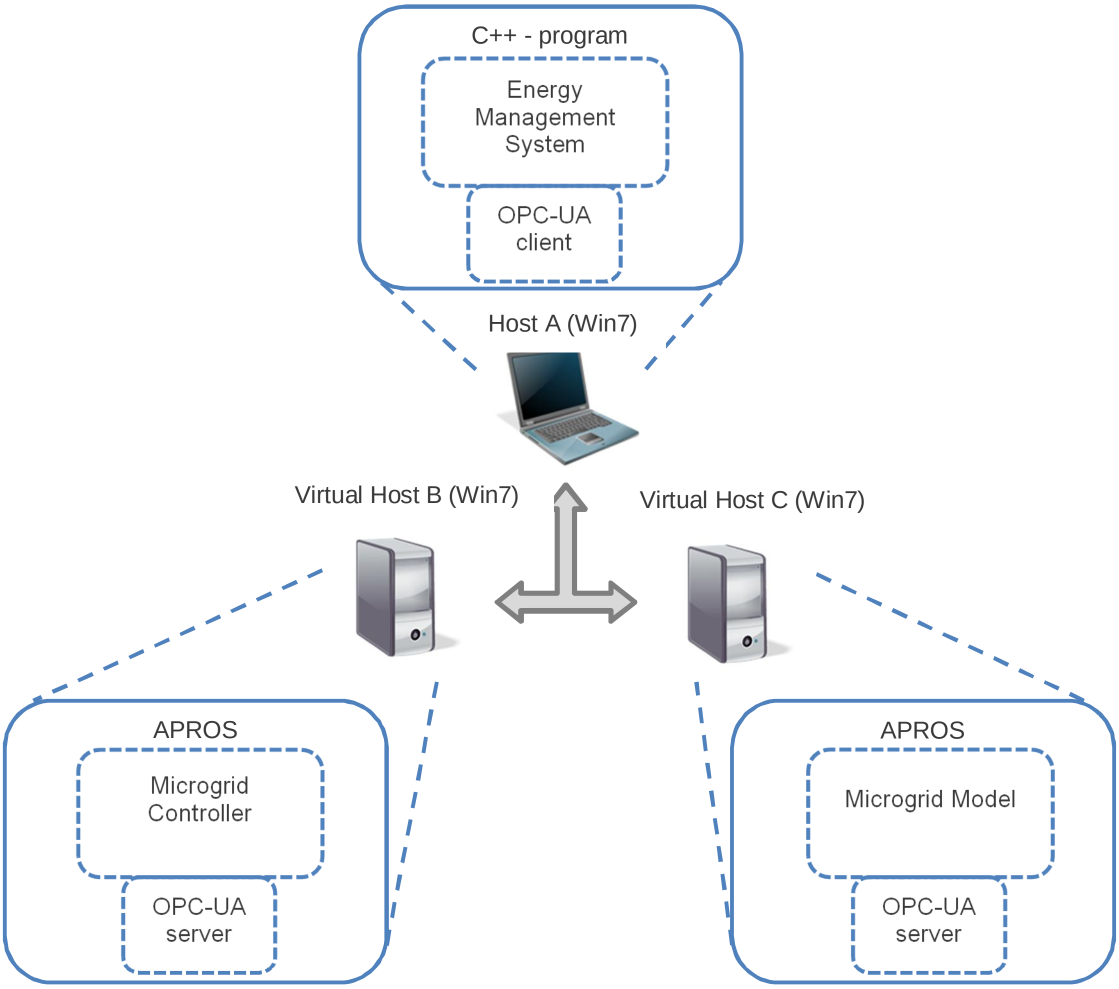

Figure 1. Simulation Platform

In the microgrid simulation platform, the hosts are using OPC UA to realize connectivity. The latest releases of APROS implement the OPC UA server so hosts B and C can be connected quite easy. The OPC UA client must be integrated to the energy management system C++ program so that it can be connected to hosts B and C as well. With OPC UA, the communication can be done so that the OPC UA servers can create signals that are sent periodically or when the signal has changed. Other OPC UA servers or clients can then listen to the signals that they are interested in. OPC UA client can also create its own signals.

Figure 2 depicts the data communication between the EMS, MG controller and the MG model. The EMS mostly sends control messages to the MG controller, which then relays them to the microgrid model. The EMS also has a possibility to send data directly to the microgrid model without sending it through the MG controller. The EMS receives feedback messages from the MG controller and measurement messages from the MG components. When the MG controller receives control messages from EMS, it adjusts them if needed and then relays them to the MG components. The MG controller receives measurements directly from the MG components. MG component receives control messages from MG controller and data from EMS. It sends component measurements or status messages to both MG controller and EMS. All messages between EMS, MG controller and MG components are sent with the OPC UA communication. During a simulation run, the EMS additionally receives data about weather conditions and market prices. In addition, initialization data is read from a configuration file in the start of a simulation.

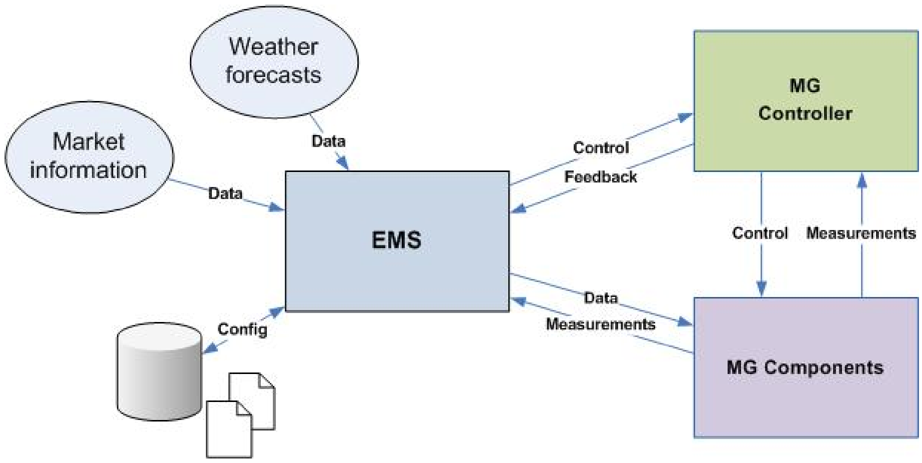

Figure 2. Data communication between EMS, MG Controller, and MG model components

Figure 3 gives a more detailed picture on different blocks in the simulation platform. The EMS block is divided into the EMS Core block and controller blocks for each type of component. These blocks are the Network Controller (NC), Generation Controller (GC), Storage Controller (SC) and Load Controller (LC). The EMS Core is responsible for running the algorithm for the energy management system.

The energy management system algorithm concentrates on the energy balance of the microgrid from the economical point of view. The Microgrid controller makes quick decisions to keep the microgrid running. It communicates directly with the microgrid components. The two algorithms make separate decisions. It can be thought that the EMS makes suggestions and the MG controller decides if those suggestions can be executed without harming or damaging the microgrid. MG Components contain a combination of controllable microgrid components, like PV units, wind power generators, controllable loads, electric vehicles, storages, etc.

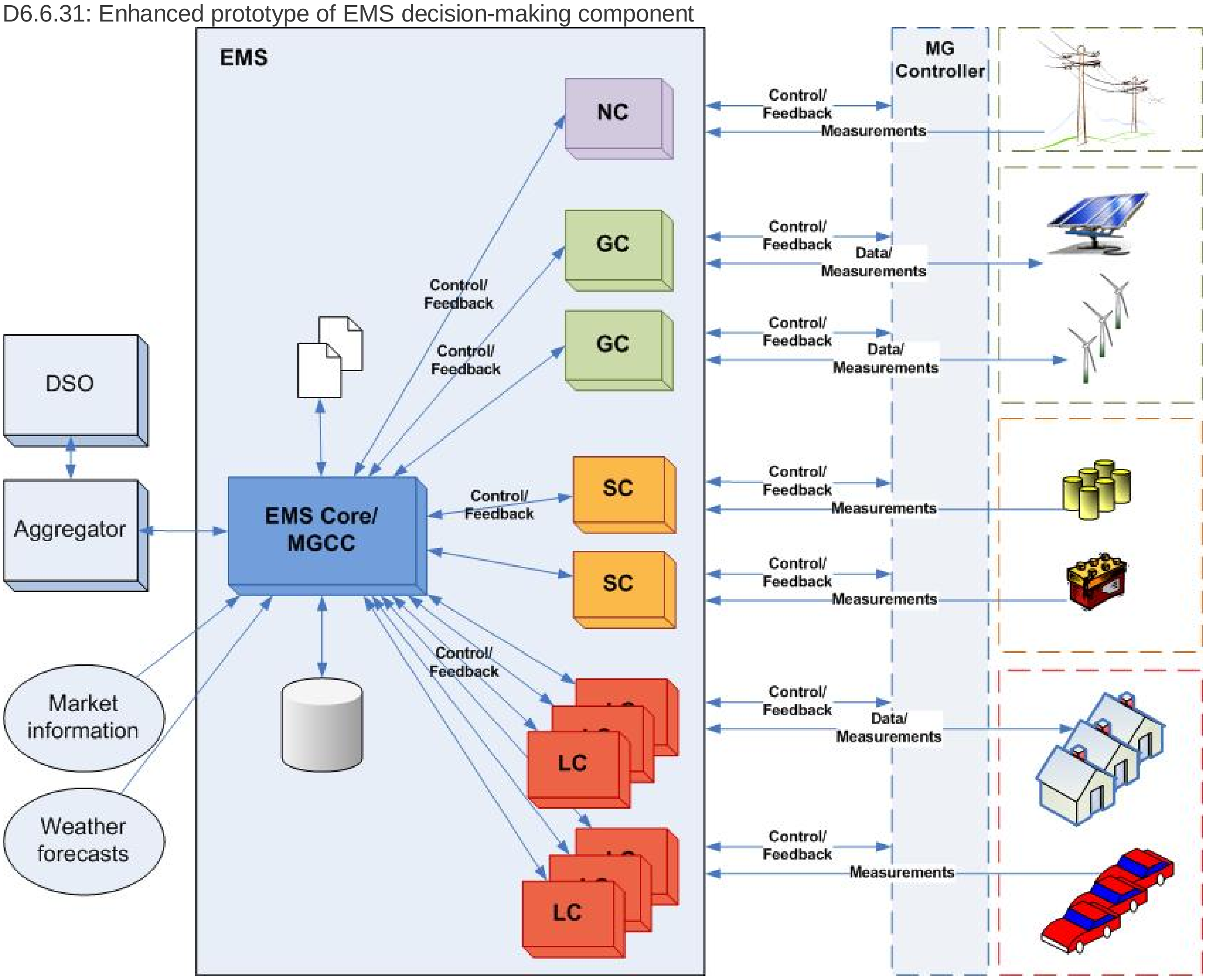

Figure 3. Simulation Platform

4.1 Energy Management System (EMS) Algorithm

The responsibilities and tasks of the EMS have been documented in previous deliverables [1] and [2]. The Simulink model is also used as a design support tool for EMS algorithm development. Running different MG setups provides information about the behavior and characteristics of the MG, which is used as a main guideline for EMS design. The EMS design challenges are divided to three categories: MG dynamics, setup and policies. Common descriptive nominator for all these categories is excess MG energy production depicted in Figure 4. The excess MG production is presented during one year starting at the beginning of the year. When the excess production is positive then energy is sold out of the MG and bought otherwise. In the depicted setup, the MG production consists of wind turbine (100kW nominal power) and solar panels (20kW maximum power) and the load consists of 200 family houses (20 MWh yearly consumption) and 50 EVs (16 kWh battery capacity and random presence pattern). MG energy storage consists of 768 kWh of battery capacity that is filled before selling and drained before buying energy outside the MG.

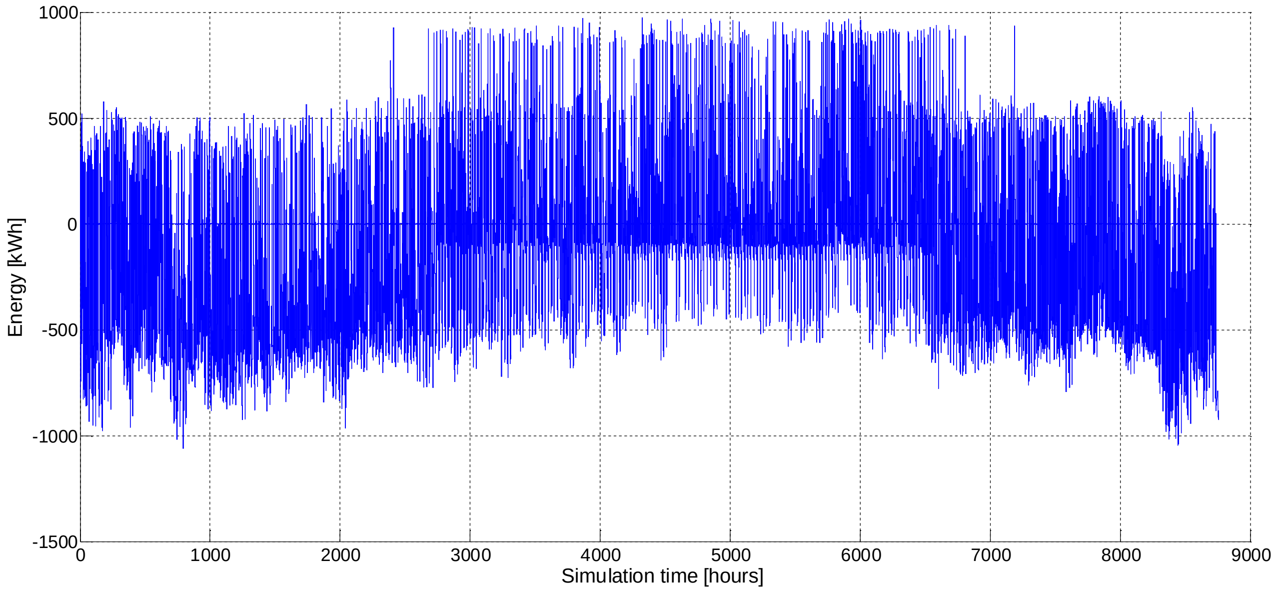

Figure 4. Excess MG energy production during one year

As Figure 4 clearly shows the dynamics of the energy production and consumption provides the main challenge for the EMS algorithm control reasoning. In the presented case the main dynamics in the production are caused by variable wind speeds and solar radiation while the consumption variation is mainly caused by house heating demand (outside air temperature) and day-time increase in household energy usage (time of day) as well as presence of the EVs. On some extend these dynamics are predictable; average outside air temperature and solar radiation are lower during winter and nights, households consume more energy during day than night (excluding heating) and more EVs are present at night. In the short term, the weather forecasts and EV presence schedules can be used to increase the predictability.

On the other hand, the EMS control decisions will have predictable consequences such as load control that may be used to limit energy usage of controllable household loads. Decision can also be made not to charge EVs batteries. On the MG production side the control decisions do not have such a significant role since the operating costs are usually much lower than the market prices, (also purchase price is usually higher than feed-in price) which stipulates constant full utilization in the MG production.

A crucial issue related to MG dynamics is the maximum impact duration that the EMS algorithm can affect the system in the future. An arguable issue is the load control implementation. Is it acceptable not to charge EVs as fast as possible and could they be used as a power source (assuming only one charging rate)? What is the maximum household load that can be controlled (and for how long) and will the controlled load be used later? The MG storages provide another mechanism to affect the future of the MG operations. In practice, storage charging and discharging limits restrict the full utilization of the storage capacity and therefore the future impact would not be significant. Therefore, for the EMS algorithm design the MG dynamical temporal properties should be examined to select proper forecast length and enable control decisions in which short term impact would not ruin more potentially beneficial long term impact.

EMS policy will eventually define how the EMS algorithm will process the dynamics of the system. For instance, if the policy is profit motivated and MG forms a single monetary entity, a possible underlying control principle is to adapt MG consumption to the MG production to eliminate buying energy from the market. Such principles could be the basis of a reference EMS algorithm design but it should be redesigned if some fundamental MG dynamic behavior changes (such as market sell price is lower than feed-in price). Therefore, the main benefit of the developed EMS algorithm would be adaptation whereas the performance of the setup tailored EMS algorithms are the benchmarks that are pursued.

Considering MG dynamics the main EMS design challenges are

- To what extent are the dynamics in energy production and consumption as well as energy market prices predictable?
- What is the maximum impact duration of the proactive EMS control decisions and how to take into account the long-term benefits over the short-term control decisions?
- How to achieve an adaptable EMS algorithm design without underperforming compared to simple setup-tailored reference algorithms?

The maximum performance that can be achieved with the EMS algorithm depends on the MG setup, namely the kind of MG production that is present and the nominal powers, the overall MG consumption and their consumption profiles as well as the amount of MG storage capacity and its maximum charging/discharging powers. The setup reasoning could be done e.g. by trying to balance the MG production and consumption such as in Figure 4, which would result usually in dynamics that are most challenging to the EMS algorithm. Another setup reasoning could be policy based, in which e.g. MG production investment costs would be taken into account to select the equipment. However, the policy based reasoning could lead to setups that are trivial to EMS algorithm e.g. in which the MG production would never meet MG consumption.

As described above the load control and prediction depends much on the MG setup. One can easily come up with multiple means for control and prediction such as scheduling, forcing and reward based mechanisms. Taking account all these mechanisms in detail would not be reasonable but simple feedback primitives from load to EMS are utilized that would compass most of the practical control and prediction mechanisms.

Considering MG setup the main EMS design challenges are

- To what extent are the dynamics in energy production and consumption controllable?
- What kinds of means are used to enable better forecasts of the energy production and consumption?

The EMS algorithm will follow a specified policy that defines the high-level goal for the control decisions. Some of the common policies might emphasize economic benefits, environmental sustainability, MG self-sufficiency or consumer happiness. The challenge with the EMS algorithm is to enable integration of different policies. Therefore, it is necessary to create taxonomy for different policies that clearly define not only what the included goals are but also what the excluded goals are and what the relationships/priorities between the goals are. For example, economic benefits could be achieved easily by causing power outages to the households. In addition, it is not always clear what is the MG components relationship with the policies. This is evident for example when deciding if the economic benefit of the MG takes into account the energy prices of the load consumption individually.

Considering MG policies the main EMS design challenges are

- Provide taxonomy and integration of EMS policies.
- Define relationship between EMS policy and components.

4.2 Microgrid Controller

Main task of the microgrid controller is to maintain power quality in microgrid while operating disconnected from distribution system by controlling different DER devices in the system. Other important task is to relay the requests of the EMS algorithm to DER devices taking into account technical limitations such as power grip and power limits. Microgrid controller can operate without an external EMS algorithm. EMS algorithm steers the microgrid to more economic operation in the long run while the APROS microgrid control algorithm takes care of rapid steering of components to ensure safe operating conditions.

Simple algorithm for Stand-alone MG controller when changing grid to islanded mode:

1) Measure grid status
2) Send information to energy storage about whether island mode is enabled
3) Energy storage changes operation mode according to grid status
4) Energy storage power flow is measured in islanded mode
5) If power flows away from storage, load control is enabled if available
6) After load control is tried, power flow measurement of ES power flow is sent to generators to relieve ES from power supply
7) Power generators supply remaining power demand and ES takes care of quick power fluctuations

Simple algorithm for returning to grid connected mode

1) Measure grid status
2) If main grid is ok, send command for energy storage to change voltage angle to grid angle
3) Synchronize MG to main distribution system after angles match
4) Change ES control mode to power control and shutdown diesel
5) Reconnect disconnected loads
6) Adjust ES power to charging at steady current
7) Adjust ES power to zero when ES capacity is fully charged

More details about the APROS microgrid control algorithm will be available in a deliverable prepared in SGEM project task 5.1.4.

4.3 Microgrid Simulation Model

Figure 5 depicts a simple microgrid model used in the design of the MG controller. Microgrid model described in this chapter consists of a MG controller, a communication emulator, an energy storage system, a diesel generator, two buildings with solar panel installation, a controllable load, and connection point to medium voltage grid via transformer. Simulation model is built so that it operates according to algorithm described in chapter 4.2.

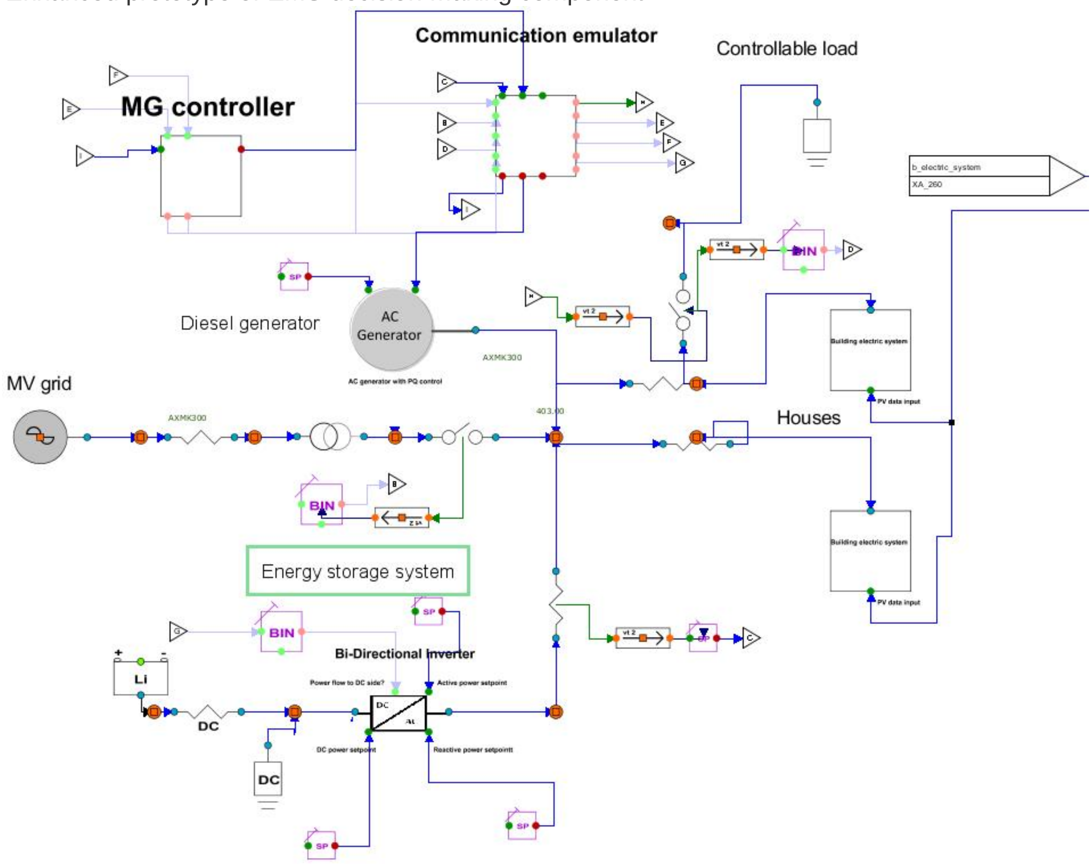

Figure 5. Simulation diagram of a microgrid in APROS software

All measurement and control signals are routed via communication emulator, which simulates different properties and behavior of various means of communication. Communication emulator is a separate model for APROS simulator and it communicates with APROS via an OPC server like it was described in chapter 3.

More details about the APROS microgrid model will be available in a deliverable prepared in SGEM project task 5.1.4.

5 Signaling Between EMS, Microgrid Controller and Microgrid Models

This chapter tells how OPC UA data communication works in the simulation platform. Figure 6 depicts the initialization signaling when the EMS is turned on. When the simulation is started, it is assumed that the microgrid controller and the microgrid model are already running before the EMS is started. Power generators, EVs, storages, breaker, and loads send status updates periodically to the MG controller and the EMS. Status messages tell the EMS the presence or absence of components in the microgrid model. Power generators also send periodic updates on the amount of their current output power. Electric vehicles and batteries send periodic updates on their state of charge and their current power. Load components send updates on their current power load in addition to their status messages. EMS must wait for status messages from all components before it can start the decision-making process.

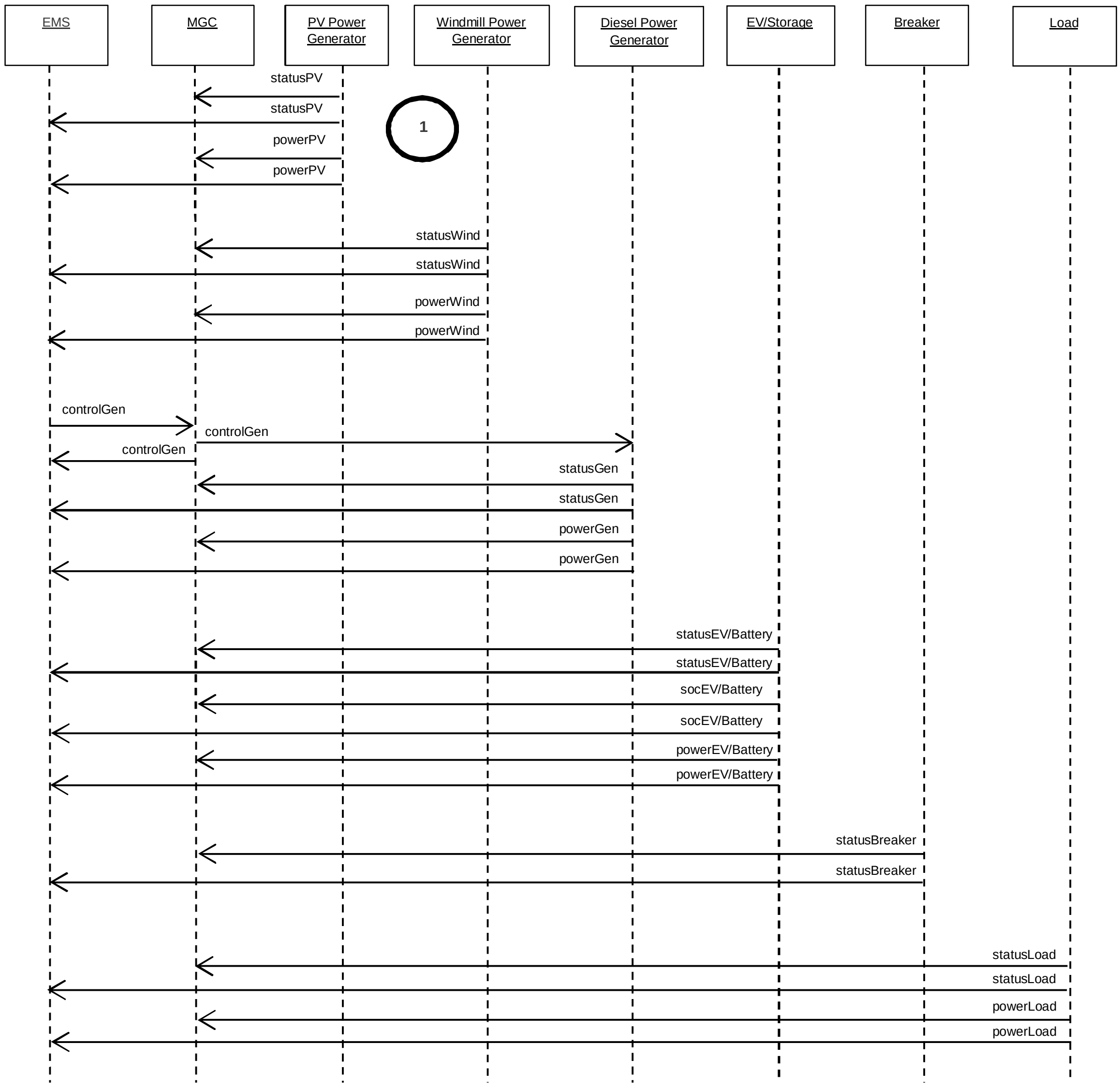

Figure 6. Initialization. (1) Components in the Microgrid model send status and power signals to Microgrid controller and EMS periodically.

Figure 7 depicts the signaling with the microgrid's generation components. When the EMS decides to request more or less power from the generation components, it first sends a *powerSet* signal to the MG controller. The MG controller checks the request and decides if it is executable or not. If the request is ok, it is sent to the generation component as is, but if the MG controller decides to modify the request, then the modified request is sent to the generation component instead. The *powerSet* value is also sent back to the EMS as a feedback because it needs to know the actual command that was sent to the generation component. The generation component sends periodic updates (*powerPV*) about the current generated power to the MG controller and the EMS. If the generation component requires additional information during the simulation, this data is sent to it by the EMS block.

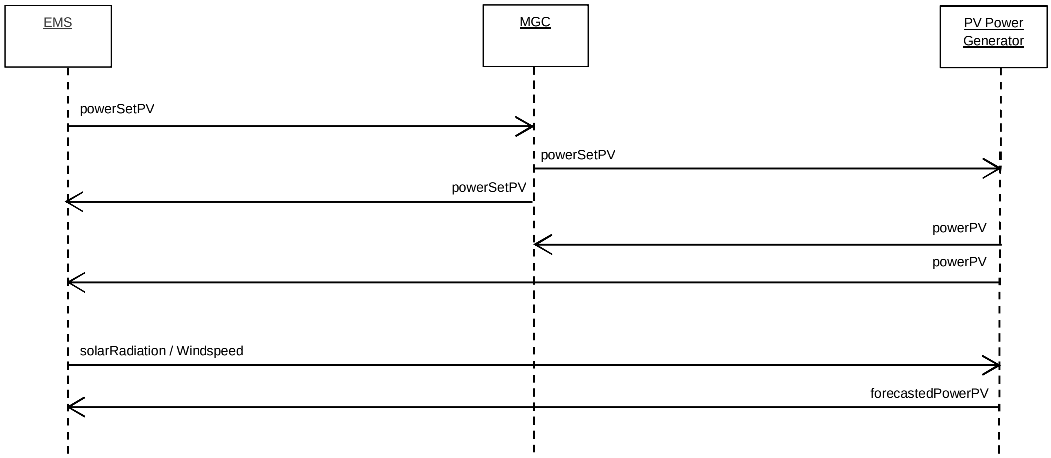

Figure 7. PV/Windmill/Diesel Generation Control

Figure 8 depicts the signaling of the EMS, MG controller and the load components in the microgrid model. During the simulation, the load components send a *controllabilityLoad* signal to the MG controller and the EMS. This signal tells whether the load component participates in demand response or not. The loads that can be controlled, receive *powerSetLoad* signals from the EMS via the MG controller. The logic with *powerSetLoad* signals is similar to the way generation components are controlled with *powerSetPV* signals. The EMS sends the *powerSetLoad* request to the MGC, which adjusts it or sends it as is to the load component. Either way the EMS receives the signal sent to the load component as a feedback. The EMS also controls the duration of the *powerSetLoad* command with the *durationLoad* signal. If the load components need weather condition data, it can be sent from the EMS. The load components also send periodic updates on the current power load with the *powerLoad* signal.

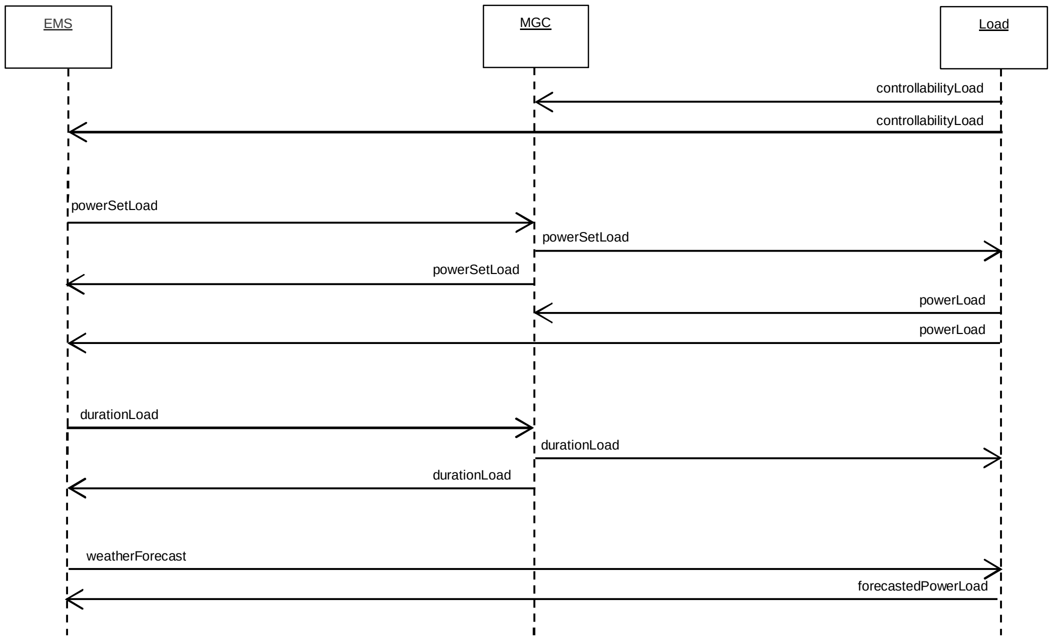

Figure 8. Load Control / Demand Response

Figure 9 depicts the signaling with the electric vehicle and storage components. EVs and battery components behavior is quite similar in the simulation. Their interfaces are similar, but the difference is that EVs are not always present for as batteries are. The presence of EVs is indicated by the *presenceEV* signal. If the EV is not present in the simulation then it naturally cannot be charged or discharged. However, EVs can and should discharge when they are not plugged in. The control signals available for the EMS include *controlEV/Battery* and *powerSetEV/Battery*. The EVs and batteries send a signal *statusEV/Battery* telling whether they are charging, discharging or idle. A *socEV/Battery* signal tells the state of charge and the *powerEV/Battery* signal tells the current discharging or charging power depending on the control state.

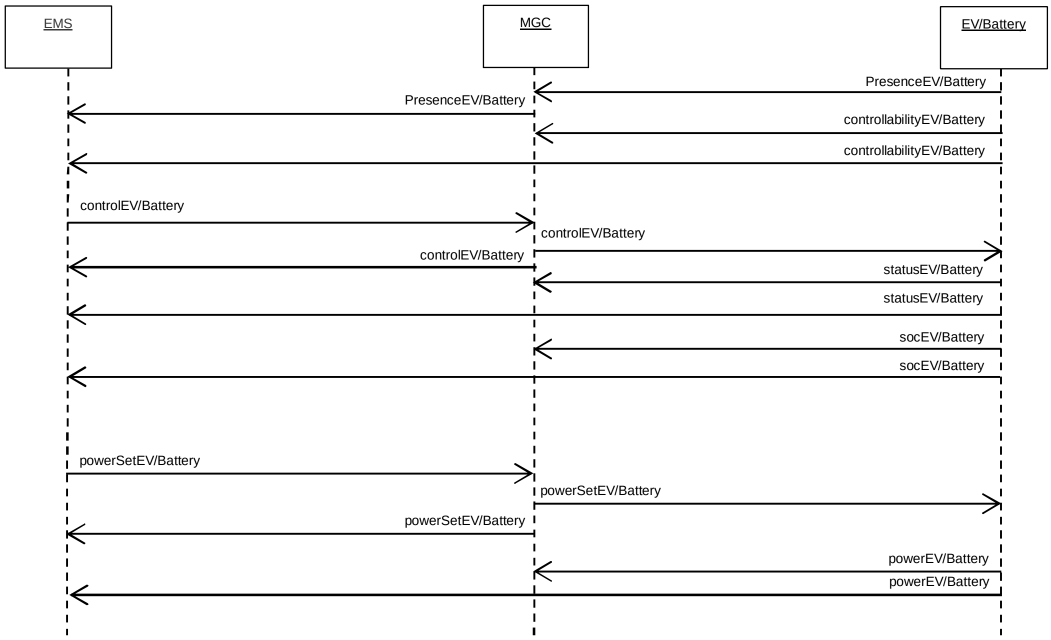

Figure 9. EV/Battery Charge & Discharge

The complete interface tables can be found in the appendices.

6 Updated Microgrid Simulink Model for EMS Algorithm Development

The Simulink model described in [2] was updated to suit the testing and development of the new EMS algorithm. The model provides an independent EMS algorithm development environment that captures the essential functionalities of the microgrid to enable rapid algorithm testing in different microgrid setups. The main updates to the model are related to EMS algorithm interface, signaling and interfaces between microgrid components and an electric vehicle (EV) model.

The EMS algorithm interface is implemented as a MATLAB function block and its implementation is imported from a .dll-file that contains an EMS API written in C++. The inputs for the block are simulation time and EMS feedback signals specified in chapter 9. Within the block, the input signals are converted to C++ format for EMS API function calls that are used to provide input for the EMS algorithm and to receive control decisions from the EMS algorithm.

6.1 Electric Vehicle (EV) Model

The model of the electric vehicle is depicted in Figure 10 and the parameters are presented in Table 1. The model is based on a MATLAB Function block that implements the model behavior. The number of EVs present at every simulation step is based on the mask parameter *TotalEVs* that remains constant during the simulation and on the input parameter *presenceEVIn* that is read from a file. Various scenarios with different patterns of EV presence can be therefore easily implemented.

The main algorithm of the EV model determines how the changes in the EV presences affect the EV batteries total state-of-charge. The algorithm decides the total state-of-charge as follows

- Low battery state-of-charge (10-50%) EVs tend to arrive
- High battery state-of-charge (50-90%) EVs tend to leave

Therefore, the algorithm induces some randomness to each simulation run.

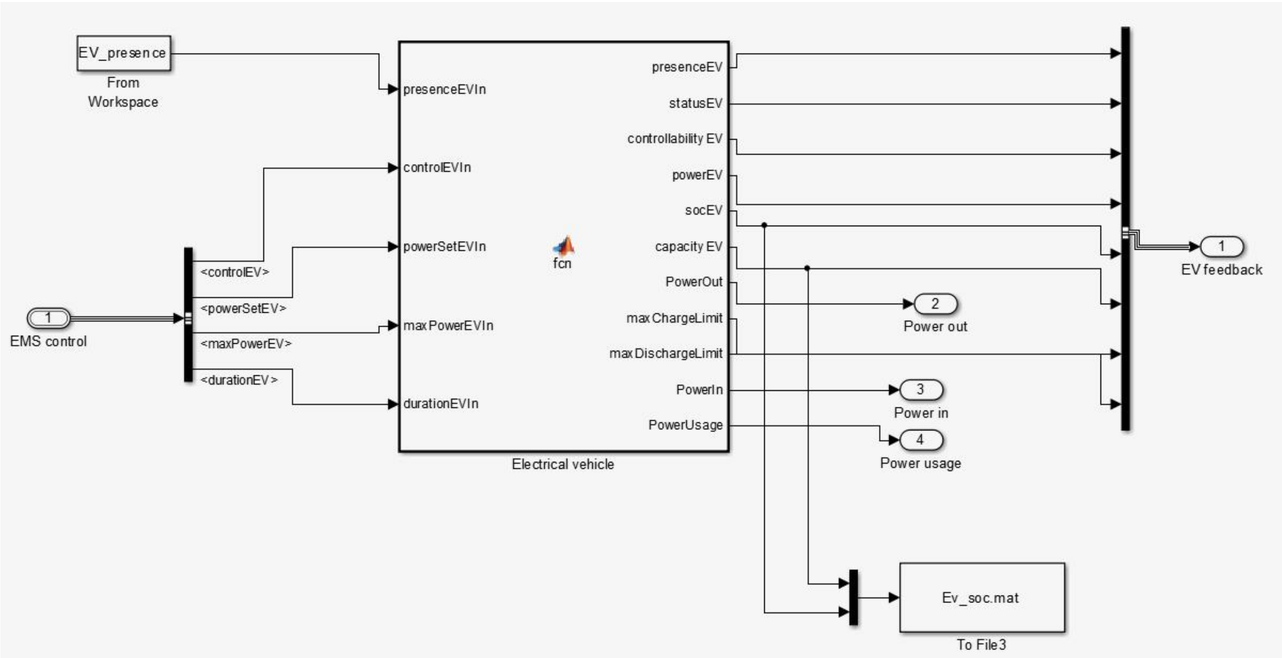

Figure 10. Electric vehicle model

Table 1. Electric vehicle model parameters

Type	Name	Unit	Note
Mask parameter	TotalEVs	integer	Total number of EVs
Mask parameter	EVBatteryCapacity	kWh	EV Battery capacity
Mask parameter	EVMaxChargingPower	kW	EV Maximum charging power
Mask parameter	EVMaxDischargingPower	kW	EV Maximum discharging power
Mask parameter	controllabilityEVIn	decimal	Fraction of controllable present EVs
Input	presenceEVIn	decimal	Fraction of total EVs present
Input	powerSetEVIn	kW	Power set given by MGC
Input	controlEVIn	[-1,0,1]	Charge/idle/discharge command from MGC
Input	durationEVIn	h	Duration for the powerSetEVIn

Output	presenceEV	[0,1]	Indicates if there are any EVs present
Output	controllabilityEV	[0,1]	Indicates if there are any EVs present that are controllable
Output	capacityEV	kWh	total capacity of the present controllable EVs
Output	socEV	%	state of charge in percentage of the capacityEV
Output	statusEV	[-1,0,1]	Current charge/idle/discharge status
Output	powerEV	kW	Power related to statusEV
Output	PowerOut	kW	Indicates current output power
Output	PowerIn	kW	Indicates current input power
Output	maxChargeLimit	kW	Maximum limit for total charge
Output	maxDischargeLimit	kW	Maximum limit for total discharge
Output	PowerUsage	kW	Actual energy usage in EVs

7 Conclusions and Future Work

This deliverable reports the development of the new microgrid simulation platform and the updates to the previous Simulink microgrid simulator. This is still a work in progress because agreeing on mutually beneficial targets with task 5.1 has taken some extra time. However, we are confident that the benefits of co-operating with another SGEM task far outweigh the challenges in the end.

Completed work includes the specifications for microgrid simulation platform, more precisely, the definition of internal and external interfaces, signaling between components and enhanced Simulink simulation model for testing the EMS algorithm.

Next steps in the development include:

- Defining use case(s) for the simulation platform
- Decisions on EMS design challenges:
    - MG dynamics
    - MG setup
    - MG policies

Work will continue in SGEM Phase 5 (FP5), starting in March 2014.

8 References

[1] Pekka Savolainen and Heli Kokkoniemi-Tarkkanen: SGEM project deliverable (FP2): D6.12.7 “Survey Report on KPIs for the Control and Management of Smart Grids”. 2012.

[2] Pekka Savolainen, Heli Kokkoniemi-Tarkkanen, Jouni Hiltunen, and Pekka Ruuska: SGEM project deliverable (FP3): D6.6.21 “A Prototype of an Intelligent Decision-making Component for Smart Grids”. 2013.

[3] http://www.opcfoundation.org/ link checked on 21.2.2014.

[4] K. Eger, J. Götz, R. Sauerwein, A. von Jagwitz, D. Boëda, R. del Olmo Arce, C. Matrose, M. Gödde and F. Chatzipapadopoulos, “FI.ICT-2011-285135 FINSENY D.3.1 v1.0 Microgrid Scenario Building Blocks,” (FP7) Siemens AG, B.A.U.M., Grenoble INP, Iberdrola S.A., RWTH, Synelixis, 05 08 2011. [Online]. Available: http://www.fi-ppp-finseny.eu/wp-content/uploads/2012/05/D3.1_Microgrid_Scenario.pdf. [Checked 28 02 2014].

[5] Robert Weiss: SGEM project deliverable (FP3): D5.1.46 “Requirement documentation on selected promising use-cases for Smart Grid communication simulator environment”. August 2013.

9 Appendices

Interfaces of EMS, MG controller and microgrid model

Description of the columns in the following tables:

- **Name**: Parameter name
- **Type [unit]**: Data type and unit for the parameter
- **Source**: Component(s) which provides the parameter
- **Destination**: Component(s) which uses the parameter
- **Note**: Short description of the parameter and other possible comments

Microgrid configuration <-> EMS

Name	Type [unit]	Source	Destination	Note
generator	struct	conf. file	EMS	Static information of windmill, PV, and diesel gen. units
load	struct	conf. file	EMS	Static information of houses and electric vehicles
storage	struct	conf. file	EMS	Static information of batteries

Market information <-> EMS

Name	Type [unit]	Source	Destination	Note
marketPurchasePrice	double [€/kWh]	file/Nordpool	EMS	Hourly price
marketSellingPrice	double [€/kWh]	file/Nordpool	EMS	Hourly price

Weather forecasts <-> EMS

Name	Type [unit]	Source	Destination	Note
weatherForecast	time series - double [°C]	file/Foreca	EMS	Average temperature at certain location. Needed for calculating load forecasts.
windSpeed	time series - double [m/s]	file/Foreca	EMS	Average wind speed at certain location. Needed by APROS windmill

				generators.
solarRadiation	time series - double [W/m²]	file/Foreca	EMS	Average solar radiation at certain location. Needed by APROS windmill generators.

MG busbar <-> MG Controller

Name	Type [unit]	Source	Destination	Note
voltageMeas1	double [V]	MG busbar	MG controller	Phase A voltage measurement
voltageMeas2	double [V]	MG busbar	MG controller	Phase B voltage measurement
voltageMeas3	double [V]	MG busbar	MG controller	Phase C voltage measurement
currentMeas1	double [amp]	MG busbar	MG controller	Phase A current measurement
currentMeas2	double [amp]	MG busbar	MG controller	Phase B current measurement
currentMeas3	double [amp]	MG busbar	MG controller	Phase C current measurement

Breaker <-> MG Controller

Name	Type [unit]	Source	Destination	Note
controlBreaker	int [0/1]	MG controller	Breaker	Open/close breaker (command)
statusBreaker	int [0/1/2]	Breaker	MG controller	Breaker status (open/closed/fail)

Breaker <-> EMS

Name	Type [unit]	Source	Destination	Note
statusBreaker	int [0/1/2]	Breaker	EMS	Breaker status (open/closed/fail)

PV unit <-> MG Controller

Name	Type [unit]	Source	Destination	Note
controlPV	int [1/0]	MG controller	PV unit	Connect/disconnect (command)
statusPV	int [1/0]	PV unit	MG controller	Connection status (connected/dis-connected)

powerSetPV	double [kW]	MG controller	PV unit	Setpoint for power production (command)
powerFactorSetPV	double [-1..1]	MG controller	PV unit	Setpoint for power factor
powerPV	double [kW]	PV unit	MG controller	Current output power
powerFactorPV	double	PV unit	MG controller	Current power factor

PV unit <-> EMS

Name	Type [unit]	Source	Destination	Note
statusPV	int [1/0]	PV unit	EMS	Connection status (connected/dis-connected)
powerPV	double [kW]	PV unit	EMS	Current output power
solarRadiation	time series - double [W/m²]	EMS	PV unit	Average solar radiation at certain location. Needed by APROS windmill generators
forecastedPowerPV	time series - double [kW]	PV unit	EMS	Future output power
controlStepPV	double [s]	(PV unit or) conf. file	EMS	Control step length
maxControlLimitPV	double [kW]	(PV unit or) conf. file	EMS	Maximum limit for the power generation
minControlLimitPV	double [kW]	(PV unit or) conf. file	EMS	Minimum limit for the power generation

Wind power generator (Wind) <-> MG Controller

Name	Type [unit]	Source	Destination	Note
controlWind	int [1/0]	MG controller	Wind	Connect/disconnect (command)
statusWind	int [1/0]	Wind	MG controller	Connection status (connected/dis-connected)
powerSetWind	double [kW]	MG controller	Wind	Setpoint for power production (command)
powerFactorSetWind	Double	MG controller	Wind	Setpoint for power factor

powerWind	double [kW]	Wind	MG controller	Current output power
powerFactorWind	Double	Wind	MG controller	Current power factor

Wind power generator (Wind) <-> EMS				
Name	**Type [unit]**	**Source**	**Destination**	**Note**
statusWind	int [1/0]	Wind	EMS	Connection status (connected/dis-connected)
powerWind	double [kW]	Wind	EMS	Current output power
controlStepWind	double [s]	(Wind or) conf. file	EMS	Control step length
maxControlLimitWind	double [kW]	(Wind or) conf. file	EMS	Maximum limit for the power generation
minControlLimitWind	double [kW]	(Wind or) conf. file	EMS	Minimum limit for the power generation
windSpeed	time series - double [m/s]	EMS	Wind	Average wind speed at certain location. Needed by APROS windmill generators
forecastedPowerWind	time series - double [kW]	Wind	EMS	Future output power

Diesel generator (Gen) <-> MG Controller				
Name	**Type [unit]**	**Source**	**Destination**	**Note**
controlGen	int [1/0]	MG controller	Gen	Connect/disconnect unit (command)
statusGen	int [1/0]	Gen	MG controller	Connection status (connected/dis-connected)
powerSetGen	double [kW]	MG controller	Gen	Setpoint for power production (command)
powerFactorSetGen	Double	MG controller	Gen	Setpoint for power factor
powerGen	double [kW]	Gen	MG controller	Current output power
powerFactorGen	Double	Gen	MG controller	Current power factor

- NOTE: Diesel generator is relevant for MG Controller only when running. Startup cannot be used. statusGen and powerGen can be used to know whether the generator is running.

Diesel generator (Gen) <-> EMS

Name	Type [unit]	Source	Destination	Note
statusGen	int [1/0]	Gen	EMS	Connection status (connected/dis-connected)
powerGen	double [kW]	Gen	EMS	Current output power
controlStepGen	double [s]	(Gen or) conf. file	EMS	Control step length
maxControlLimitGen	double [kW]	(Gen or) conf. file	EMS	Maximum limit for the power generation
minControlLimitGen	double [kW]	(Gen or) conf. file	EMS	Minimum limit for the power generation
forecastedPowerGen	time series - double [kW]	Gen	EMS	Future output power

Controllable load (Load) <-> MG Controller

Name	Type [unit]	Source	Destination	Note
controlLoad	int [1/0]	MG controller	Load	Load on / off (command)
statusLoad	int [1/0]	Load	MG controller	Load status (on/off)
controllabilityLoad	int [1/0]	Load	MG controller	Available for control
powerSetLoad	double [kW]	MG controller	Load	Power setpoint for local controller (command)
durationLoad	double [s]	MG controller	Load	Control duration (command)
powerLoad	double [kW]	Load	MG controller	Current power consumption

- NOTE: Local override during the load control is needed. controllabilityLoad can be used for this when.

Controllable load (Load) <-> EMS

Name	Type [unit]	Source	Destination	Note
statusLoad	int [1/0]	Load	EMS	Load status (on/off)
controllabilityLoad	int [1/0]	Load	EMS	Available for control
powerLoad	double [kW]	Load	EMS	Current power consumption
maxControlLimitLoad	double [kW]	(Load or) conf. file	EMS	Maximum limit for the power consumpion
minControlLimitLoad	double [kW]	(Load or) conf. file	EMS	Minimum limit for the power consumpion
weatherForecast	time series - double [°C]	EMS	Load	Average temperature at certain location. Needed for calculating load forecasts.
forecastedPowerLoad	time series - double [kW]	Load	EMS	Future consumption estimation. Simple solution is to use the realized load profile of today for tomorrow's forecast. Other solution is to modify Kari Mäki's Finnish home's load forecast-EXCEL for APROS

Electric vehicle (EV) <-> MG Controller

Name	Type [unit]	Source	Destination	Note
presenceEV	int [1/0]	EV	MG controller	Connected/disconnected
controlEV	int [1/0/-1]	MG controller	EV	Charge/idle/discharge (command)
statusEV	int [1/0/-1]	EV	MG controller	Charging status (charge/idle/discharge)
controllabilityEV	int [1/0]	EV	MG controller	Available for control
powerSetEV	double [kW]	MG controller	EV	Power set for control depending on the control state (command)
maxPowerEV	double [kW]	MG controller	EV	Maximum charging power (command)
durationEV	double [s]	MG controller	EV	Control duration (command)
powerEV	double [kW]	EV	MG controller	Current discharging or charging power depending on the control state
socEV	double [%]	EV	MG controller	Battery state of charge

Electric vehicle (EV) <-> EMS

Name	Type [unit]	Source	Destination	Note
presenceEV	int [1/0]	EV	EMS	Connected/ disconnected
statusEV	int [1/0/-1]	EV	EMS	Charging status (charge/idle/discharge)
controllabilityEV	int [1/0]	EV	EMS	Available for control
powerEV	double [kW]	EV	EMS	Current discharging or charging power depending on the control state
socEV	double [%]	EV	EMS	Battery state of charge
capacityEV	double [kWh]	(EV or) conf. file	EMS	How much energy can the EV store
maxDischargeLimitEV	double [kW]	(EV or) conf. file	EMS	Maximum limit for the discharging power
minDischargeLimitEV	double [kW]	(EV or) conf. file	EMS	Minimum limit for the discharging power
maxChargeLimitEV	double [kW]	(EV or) conf. file	EMS	Maximum limit for charging power
minChargeLimitEV	double [kW]	(EV or) conf. file	EMS	Minimum limit for charging power
forecastedPowerEV	time series - double [kW]	EV	EMS	Future discharging power or charging power depending on the control state

Storage (Battery) <-> MG Controller

Name	Type [unit]	Source	Destination	Note
presenceBattery	int [1/0]	Battery	MG controller	Connected/disconnected
controlBattery	int [1/0/-1]	MG controller	Battery	Charge/idle/discharge (command)
statusBattery	int [1/0/-1]	Battery	MG controller	Charging status (charge/idle/discharge)
controllabilityBattery	int [1/0]	Battery	MG controller	Available for control

powerSetBattery	double [kW]	MG controller	Battery	Power set for control depending on the control state (command)
maxPowerBattery	double [kW]	MG controller	Battery	Maximum charging power (command)
durationBattery	double [s]	MG controller	Battery	Control duration (command)
powerBattery	double [kW]	Battery	MG controller	Current discharging or charging power depending on the control state
socBattery	double [%]	Battery	MG controller	Battery state of charge

Storage (Battery) <-> EMS

Name	Type [unit]	Source	Destination	Note
presenceBattery	int [1/0]	Battery	EMS	Connected/disconnected
statusBattery	int [1/0/-1]	Battery	EMS	Charging status (charge/idle/discharge)
controllabilityBattery	int [1/0]	Battery	EMS	Available for control
powerBattery	double [kW]	Battery	EMS	Current discharging or charging power depending on the control state
socBattery	double [%]	Battery	EMS	Battery state of charge
capacityBattery	double [kWh]	(Battery or) conf. file	EMS	How much energy can the battery store
maxDischargeLimit-Battery	double [kW]	(Battery or) conf. file	EMS	Maximum limit for the discharging power
minDischargeLimit-Battery	double [kW]	(Battery or) conf. file	EMS	Minimum limit for the discharging power
maxChargeLimit-Battery	double [kW]	(Battery or) conf. file	EMS	Maximum limit for charging power
minChargeLimit-Battery	double [kW]	(Battery or) conf. file	EMS	Minimum limit for charging power
forecastedPowe-rBattery	time series - double [kW]	Battery	EMS	Future discharging power or charging power depending on the control state

MG Controller <-> EMS

Name	Type [unit]	Source	Destination	Note
controlBreaker	int [0/1]	EMS	MG controller	Open/close breaker (recommendation)
controlBreaker	int [0/1]	MG controller	EMS	Open/close breaker (feedback)
controlPV	int [1/0]	EMS	MG controller	Connect/disconnect (recommendation)
controlPV	int [1/0]	MG controller	EMS	Connect/disconnect (feedback)
controlWind	int [1/0]	EMS	MG controller	Connect/disconnect (recommendation)
controlWind	int [1/0]	MG controller	EMS	Connect/disconnect (feedback)
controlGen	int [1/0]	EMS	MG controller	Connect/disconnect unit (recommendation)
controlGen	int [1/0]	MG controller	EMS	Connect/disconnect unit (feedback)
controlLoad	int [1/0]	EMS	MG controller	Load on / off (recommendation)
controlLoad	int [1/0]	MG controller	EMS	Load on / off (feedback)
controlEV	int [1/0/-1]	EMS	MG controller	Charge/idle/discharge (recommendation)
controlEV	int [1/0/-1]	MG controller	EMS	Charge/idle/discharge (feedback)
controlBattery	int [1/0/-1]	EMS	MG controller	Charge/idle/discharge (recommendation)
controlBattery	int [1/0/-1]	MG controller	EMS	Charge/idle/discharge (feedback)
powerSetPV	double [kW]	EMS	MG controller	Setpoint for power production (recommendation)
powerSetPV	double [kW]	MG controller	EMS	Setpoint for power production (feedback)
powerSetWind	double [kW]	EMS	MG controller	Setpoint for power production (recommendation)
powerSetWind	double [kW]	MG controller	EMS	Setpoint for power production (feedback)
powerSetGen	double [kW]	EMS	MG controller	Setpoint for power production (recommendation)
powerSetGen	double [kW]	MG controller	EMS	Setpoint for power production (feedback)
powerSetLoad	double [kW]	EMS	MG controller	Power setpoint for local controller (recommendation)
powerSetLoad	double [kW]	MG controller	EMS	Power setpoint for local controller (feedback)
powerSetEV	double [kW]	EMS	MG controller	Power set for control depending on the control state (recommendation)
powerSetEV	double [kW]	MG controller	EMS	Power set for control depending on the control state (feedback)

powerSetBattery	double [kW]	EMS	MG controller	Power set for control depending on the control state (recommendation)
powerSetBattery	double [kW]	MG controller	EMS	Power set for control depending on the control state (feedback)
maxPowerEV	double [kW]	EMS	MG controller	Maximum charging power (recommendation)
maxPowerEV	double [kW]	MG controller	EMS	Maximum charging power (feedback)
maxPowerBattery	double [kW]	EMS	MG controller	Maximum charging power (recommendation)
maxPowerBattery	double [kW]	MG controller	EMS	Maximum charging power (feedback)
durationLoad	double [s]	EMS	MG controller	Control duration (recommendation)
durationLoad	double [s]	MG controller	EMS	Control duration (feedback)
durationEV	double [s]	EMS	MG controller	Control duration (recommendation)
durationEV	double [s]	MG controller	EMS	Control duration (feedback)
durationBattery	double [s]	EMS	MG controller	Control duration (recommendation)
durationBattery	double [s]	MG controller	EMS	Control duration (feedback)