An embedded software middleware that supports PCS

Chao Chen, Jiangtao Li

2023 IEEE Sustainable Power and Energy Conference (iSPEC) | 979-8-3503-2269-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/iSPEC58282.2023.10402926

***Abstract* — Search PCS-related articles to find a large number of articles that introduce from the perspective of hardware, systems, and AC/DC conversion principles, and few articles explain how to design and implement from the perspective of embedded platform software. Embedded software plays the role of "brain" in PCS and is an extremely important part. The author's "A solution for relay protection equipment" presented at iSPEC2021 introduced a GSP secondary equipment platform, and as the work was adjusted, the author had the opportunity to learn more about similar platforms and products in the industry. This paper mainly describes another embedded software platform, which is used as embedded software middleware for PCS products to support product development. The design idea of the middleware and the various access capabilities provided in the form of components are given. In specific product development, the final PCS product can be quickly built with the help of the task scheduling, sampling, valuation, events, modbus communication, BCMU management and other capabilities provided by the middleware. At the end, the two platform solutions proposed in the two articles are compared and summarized, and some common ideas and needs are extracted.**

Keywords — PCS; Power Conversion System;PCS Platform middleware; PCS Embedded software;Secondary equipment platform;

I. INTRODUCTION (*HEADING 1*)

Power Conversion System(PCS), also known as bidirectional energy storage inverter. It is mainly used in AC coupled energy storage systems such as grid-connected energy storage and micro-grid energy storage. PCS has been widely used in power systems, photovoltaic power generation, wind power generation, petrochemical industry, rail transit, new energy vehicles and other fields. In the power grid peak shaving, energy recovery and utilization, smooth new energy fluctuations and other occasions to achieve two-way energy flow, provide active support for the grid voltage and frequency, and improve the quality of power supply.

PCS is a device that connects the battery pack to the grid or load, and realizes the two-way conversion of electrical energy. It can reverse the direct current of the battery into alternating current, transmit it to the power grid or use it for AC load; It can also rectify the alternating current of the power grid into direct current to charge the battery. In the realization of energy storage battery and grid power exchange, is the key component to determine the performance of battery energy storage system BESS, this paper focuses on a kind of embedded software middleware when implementing PCS, the middleware by providing setting value debugging and storage capabilities, event storage and access capabilities, recording storage and access capabilities, modbus communication capabilities, BCMU communication capabilities to support the development of PCS products.

II. INTRODUCTION TO PCS ENERGY STORAGE CONVERTER

A. Introduction to PCS functions

As one of the important forms of large-scale energy storage system, BESS has various uses such as peak shaving, phase modulation, frequency regulation, valley filling, and disaster preparedness. Compared with conventional electric energy, large-scale BESS power stations can adapt to the rapid change of load, which plays an important role in improving the safe and stable operation level of the power system, the quality and reliability of power supply of the power grid, and at the same time can optimize the power structure, achieve green environmental protection, achieve the overall energy saving and consumption reduction of the power system, and improve the overall economic benefits.

PCS is composed of AC/DC bidirectional converter, control unit, etc., in the electrochemical energy storage system, connected between the battery system and the power grid or load, to achieve two-way conversion of electric energy, can control the charging and discharging of the battery, carry out AC and DC conversion, and can also directly supply power to the AC load in the absence of a network. The PCS controller can receive background control instructions through communication, and then control the converter to charge or discharge the battery according to the instructions to realize the regulation of the active power and reactive power of the power grid. At the same time, PCS can also communicate with BMS through the CAN port to obtain battery pack status information, realize protective charging and discharging of the battery, and ensure the safety of battery operation. Real-time monitoring of various states of BS (voltage, current, temperature, state of charge, state of health, etc.), safety management of BS charging and discharging process (such as prevention of overcharge, overdischarge management), alarm and emergency protection treatment of possible faults of BS and optimal control of BS operation, and ensure the safe, reliable and stable operation of BS. The State of Charge (SOC) of BS and its various components is a key indicator to achieve safe and reliable operation of the entire BS, as well as accurate management and control of it.

From the entire electrochemical energy storage system, PCS plays an important role in connecting the top and bottom.

979-8-3503-2269-9/23/$31.00 ©2023 IEEE

B. *Introduction to PCS embedded software platform architecture*

The embedded software running on the PCS is the intelligent brain of the PCS, allowing the PCS to successfully complete its work. The PCS embedded software platform provides basic component support for the normal operation of the brain, and the core components include: variable debugging and storage, event triggering and storage, ADC sampling and recording, PWM drive, modbus communication, BCMU communication, etc. Figure 1 below is a software hierarchy diagram of the PCS system.

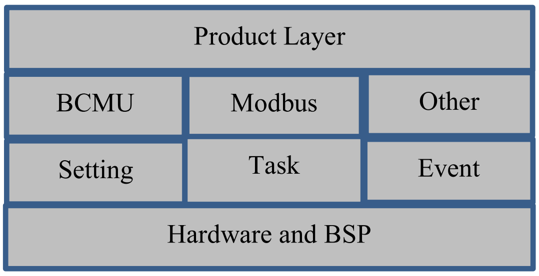

Fig. 1 PCS sfotware layer

III. PCS EMBEDDED SOFTWARE PLATFORM ACCESS CAPABILITY BUILD

The PCS embedded software platform provides access to upper-layer applications in the form of components, and upper-layer applications integrate relevant capabilities to build the final PCS product.

A. *Task management component*

The task management module is divided into four levels of tasks: Task1~Task4, and each level of task has its own function linked list that needs to be executed. The module provides a task registration function to register a task in the linked list of the corresponding level. After the system is running, the platform automatically completes the operation of the task according to the registered task.

Among them, Task 1 is the task with the highest real-time requirements, and the sampling calculation task with a scheduling period of about 40us is scheduled. The scheduling cycle of Task 2 is at the 1ms level, and the scheduling cycle of Task 3 is at the 10ms level. To avoid spectral aliasing caused by downsampling, when Task 2 needs to use variables in Task 1, you need to perform data transfer from Task 1 to Task 2.

The following figure is the interface diagram of the task management component.

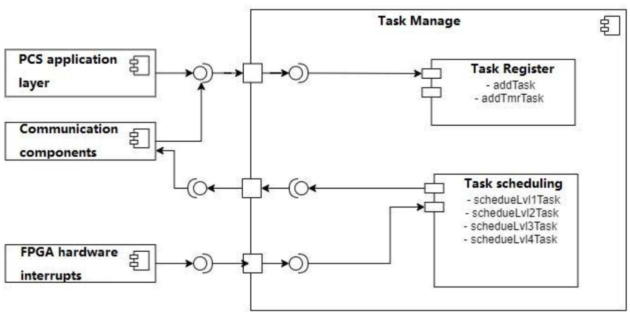

Fig. 2 Task Manage Component

B. *Analog acquisition implementation*

Analog acquisition is achieved through two 16-bit ADCs built into the DSP, and 12 analog data can be acquired simultaneously. At the software level, the drive interface is directly provided to the upper module, which can read the ADC sampling data, the driving beat is in Task1, and the design diagram of the analog acquisition function is as above.

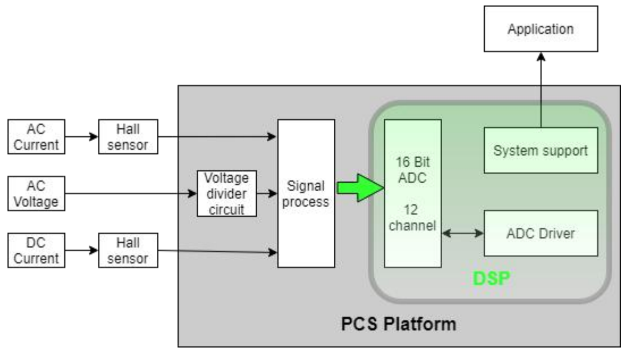

Fig.3 PCS Analog acquisition

C. *PCS setting value management and storage components*

Setting value management refers to the function that after configuration in the modbus table, it can be read and written through the modbus protocol, and can be stored in eeprom when writing to support power-down saving.

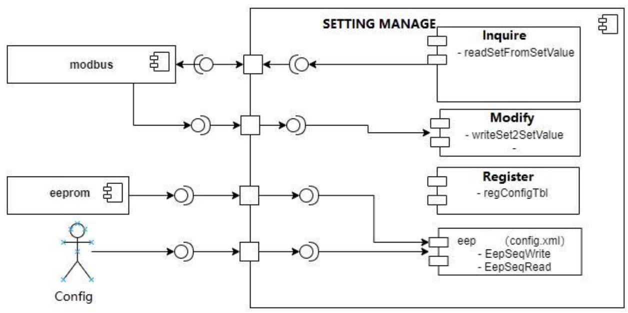

Fig. 4 Setting Component

The setting initialization process includes:

Step 1: Calculate the "parameter crc" according to the app_param_tbl[] and sys_param_tbl[] configured in config.xml

Step2: Read the "parameter crc" from the eeprom (corresponding to the "parameter crc" in step 1) and "parameter run value crc"

step3: If the "parameter crc" is inconsistent, the parameter configuration is considered to have changed. Read the default value in config.xml to the buffer area, update the "parameter crc" in eeprom (and store the default value in eeprom), and then go to step 5.

If the "parameter crc" is consistent, the parameter configuration is considered to have not changed. Read the "parameter run value" in eeprom to a buffer area that can store all application and platform parameter values in config.xml, and calculate the "parameter run value crc" step4：The "parameter run value crc" read from eeprom in step2 is inconsistent with the "parameter run value crc" calculated in step3, then the data in eeprom is considered to have been invalidated, and the default value in config.xml is read to the buffer of the parameter value.

If the "parameter running value crc" is consistent, the data in the EEP is considered valid.

Step5: Finally synchronize the values in the buffer of the above parameter values into the "Parameter Run Original Variable".

D. PCS event storage and access components

Events are divided into real-time events and historical events, and when upstream tools or machine controls obtain events, the event-related information is transmitted through modbus by calling the system's event-related interface. An event is mainly composed of the following properties.

TABLE I
PCS EVENT PROPERTY

Event properties	Type
Live event flags	UINT16
Real-time event register address	UINT16
Real-time event time seconds	UINT32
Real-time event time in microseconds	UINT32
Real-time event data	UINT32
Historical event log file entries	UINT16
Historical event record file size	UINT32

In the PCS solution, the interaction between the event and the external application is shown in the following figure.

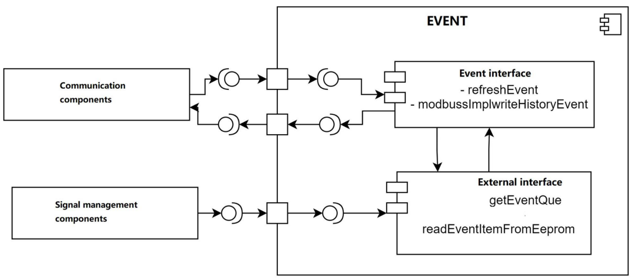

Fig. 5 Event Component

E. BCMU components

The BCMU service is responsible for data interaction with the BMS device, obtaining medium and slow data such as system status, battery voltage and temperature sent by the BMS device on the can bus, and issuing application data such as battery cluster current and PCS status.

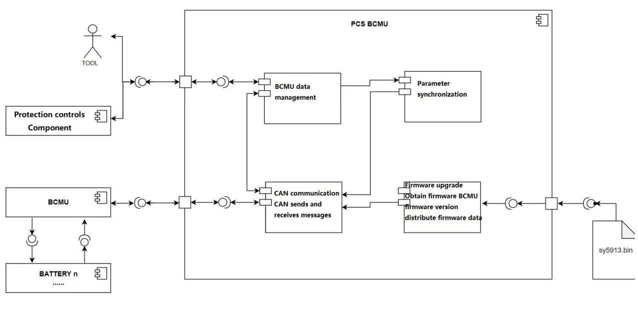

Fig. 6 BCMU Component

The main data provided by BCMU is shown in the table below.

TABLE II
BCMU DATA

Event properties	Type
Battery cluster current	App data
PCS status, voltage, impedance	
Time synchronization	Control data
BCMU firmware update	
Data updates	
Battery parameters	
Protection threshold parameters	
Protection delay parameter	
Software version	
BMS system status, current limit, SOC	Medium speed data
Battery cluster failure	
Battery cluster alarm	
Maximum voltage and number	
Maximum temperature, number, SOE	
Battery maximum differential pressure	
Battery maximum temperature difference, differential pressure number, temperature difference number	
Temperature difference number	
Battery cells, temperature point, cumulative total voltage	Slow speed data
Average voltage, average temperature, SOH, cell rated capacity	
Charging cumulative capacity	
Discharge cumulative capacity	
Cell voltage	
Cell temperature	

F. Modbus communication component

MODBUS is an application-layer messaging protocol on layer 7 of the OSI model that provides client/server communication between devices connected to different types of buses or networks. The server side of modbus communication with machine control and tools can realize PCS monitoring information transmission, setting value modification, variable debugging, file upload and download, timing, reset, return and other functions. Function codes are defined in the following table:

TABLE III
MODBUS FUN CODE

Service type	Fun code	description
Read the coil	0x02	Upload the remote telematics data of the PCS
Read registers	0x03	Forward register information to the PCS
Read registers	0x04	Telemetry data sent to PCS
Write register	0x06	Write a single register
Write register	0x10	Write multiple registers
Read file	0x14	File reading, summoning wave files
Write	0x15	File delivery, synchronization

file		program

The following figure is Modbus component diagram.

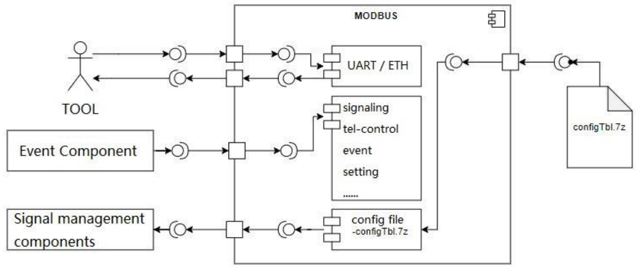

Fig. 7 Modbus Component

IV. IN CONCLUSION

This article mainly introduces some platform-level components that need to be provided to implement a PCS product in embedded software from the perspective of embedded platform software. Providing capabilities in the form of components is an important implementation idea for this product. This helps the layering and division of labor of PCS software, improves the code reuse degree of embedded software, and reduces the labor cost of product development. It was implemented smoothly in the actual product.

Compared with the GSP platform scheme introduced in the author's 2021 article, the design idea of this case mainly revolves around components. However, the requirements to be solved in this case do not go beyond the requirements in 2021 article as follows:

One channel of real-time sampling data, one channel of historical waveform data.

The current value of a digital quantity, and the historical displacement of a digital quantity.

The current value of a sensor, and the historical data of a sensor.

The state of a relay, and the historical displacement of a relay.

One control process status, one control process record.

The state of a soft switch, and a record of the switch on/off.

A current value of a setting value, and a modification record of a setting value.

The current value of a parameter, the modification record of a parameter.

An abnormal event state, an abnormal scene record.

One logical operation result, one logical operation result record.

About the new PCS software platform, it's just that these requirements are implemented through one or more components, combined with the code developed by the upper-layer application. It's just that the frequency of data acquisition in PCS is higher, and the real-time control requirements for output signals are also higher. Simply put, if the right processor is selected, the GSP solution can also implement the PCS product functions described in this article.

[1] LIANG Chenguang,WANG Bin,LI Fengting, et al.Data Correction Algorithm of Logarithmic Simulation Model for Single-phase Arc Grounding Fault on High-voltage Transmission Line[J].Automation of Electric Power Systems,2019,43(4):126-131. DOI: 10.7500/AEPS20180717008.

[2] Sang-Hyun Park, Ki-Bum Park, Hyoung-Suk Kim, Gun-Woo Moon, Myung-Joong Youn. Single-magnetic cell-to-cell charge equalization converter with reduced number of transformer windings IEEE Trans. Power Electron., 27 (2012), pp. 2900-2911, 10.1109/TPEL.2011.2178040

[3] LIU Xiaosheng, ZHENG Jian, HAI Tianxiang, et al. Reliability and Real-time Study of GOOSE Network Based on EtherCAT[J]. Automation of Electric Power Systems, 2015, 39(13):109-115. DOI:10.7500/AEPS20140523007.

[4] Sankarkumar R.S., Natarajan R.Energy management techniques and topologies suitable for hybrid energy storage system powered electric vehicles: An overview Int. Trans. Electr. Energy Syst., 31 (4) (2021), Article e12819

[5] W. Aziz, A. El Magri, A. Raihani, Nonlinear control of wind energy conversion system involving dfg, Proceedings of the 3rd International Conference on Smart City Applications, SCA ' 18, Association for Computing Machinery, New York, NY, USA (2018), 10.1145/3286606.3286820

[6] Chao Chen; Wei Xi; Changsong Qin, et al. General secondary equipment simulation drive system. 2021 IEEE 5th Conference on Energy Internet and Energy System Integration (EI2), Date of Conference: 22-24 October 2021 DOI: 10.1109/EI252483.2021.9713336

[7] Chao Chen; Changsong Qin; Xinjie Zhang, et al. A solution for relay protection equipment. 2021 IEEE Sustainable Power and Energy Conference (iSPEC) Date of Conference: 23-25 December 2021 DOI: 10.1109/iSPEC53008.2021.9735663

[8] LI Q,CAO G. Multicast authentication in the smart grid with one-time signature[J]. IEEE Transactions on Smart Grid,2011,2(4):686-696.

Manuscript received Jul 23, 2023.
Chao Chen works in the central research of Sieyuan Electric Co., Ltd,
Addr: Building 1, Zhongxing IoT R&D Building, No. 90 Huashen Avenue, Yuhuatai District, Nanjing
(corresponding author to provide phone: 13584048645; e-mail: 13320995@qq.com).