Power Distribution Unit (PDU) for a Distributed Computing Network

Muhammad Faisal, Thomas Walter, Sergio Montenegro
Institute of Aerospace Information Technology.
Julius-Maximilians-University Würzburg,
Am Hubland D-97074 Würzburg, Germany.
muhammad.faisal@uni-wuerzburg.de
thomas.walter@uni-wuerzburg.de
sergio.montenegro@uni-wuerzburg.de

***Abstract*—A power distribution unit (PDU) is an electronic module which provides electrical power to the network of electronic devices. For modern distributed computing networks a smart PDU is required which does not only supply the power to all the attached units but also monitors current consumption and the load on each node. In this paper we presented the Hardware and Software module for an intelligent PDU which offers the capabilities to distribute the power and also supervise the current and load consumption and broadcast all it parameters to the administrator. This PDU is based on a state-of-the-art ARMCortex $M4$ microcontroller which is used under a Real time operating System. A real case scenario is presented at the end of this paper validating the hardware and software design of proposed PDU.**

I. Introduction

Power integrity has a paramount importance in embedded network systems. Malfunctions in the power distribution can inject severe problems such as data loss, Crosstalk, Timing delay etc. In this paper a time domain approach was presented for the power management in a distributed network of computing nodes. The goal of this work was the designing of a Power Distribution Unit for a *Visionary Data Management System for nano–satellites* ***(VIDANA)***which is a network of computing nodes. This network was designed to implement a reliable Data management system (DMS) for small scale space vehicles, This network realizes the dependability and fault tolerance in data management systems of small satellites [1]. Designing and implementation of the PDU was a sub–part of this large project VIDANA. The VIDANA project aims to deliver a comprehensively dependable and operationally efficient computing systems with very minimal physical resources for small satellites in orbital and interplanetary missions, In order to achieve this goal a distributed computing strategy was chalked out where the whole system is decomposed into nano computing blocks which are part of a network [2], To distribute the power to these computing units a Power Distribution Unit or PDU was required which does not only provide switching capabilities but also the details of all the electrical parameters of the Power distribution so that faults can be quickly traced and network remain functional [3].
The Commercially off the shelf (COTS) PDUs were available but their usage was limited to large networks of computers as given in [3]. These COTS PDUs from different vendors were capable and efficient for a range of requirements as given in [4]. Their sizes and power consumption were relatively large and small and nano satellites have size, Weight and power constrains, Some of the vendors were even providing the nano Power modules which were designed as modular electrical power supplies for small spacecrafts [5]. There were two main problems related to these COTS PDUs, First their number of outputs were very limited and secondly they were attached with solar panel to recharge their batteries but the system presented here was based on laboratory experimental setup that's why for the VIDANA project a customized PDU was designed.

II. Functional description

The Power Distribution Unit was meant to be controlled by a small microcontroller development kit which is controlled by a real time operating system, The functional block–diagram is shown in Fig. 1. This intelligentent power distribution system for the VIDANA is based on a ARM–Cortex $M4$ based Stm32f4 microcontroller board (Discovery) [6], This microcontroller module is used as it is cost effective as well as operational efficient and the other reason to choose this development kit was that RODOS is already ported on this microcontroller module, All the interfaces for Hardware Abstraction Layers(HAL) e.g GPIOs, Analog to Digital Converters, Serial port etc are developed for Stm32f4 chip so that the development of a simple embedded software was possible. The controller of PDU is driving other fifteen nodes which are connected to the output of the PDU, These fifteen nodes are connected with the controller through its General Purpose Input and Output(GPIO) interface as shown in the block diagram of the Power Distribution Unit (Fig. 2). The controller produces 3.3v on its output and this output is connected to the power connector of every node. Every node is connected with the GPIO of the controller through an electric circuit which is described in the following section. The current and load consumption is monitored through a digital current sensor [7], Its output is directly feeded into the Analog–to–Digital input of the the controller.

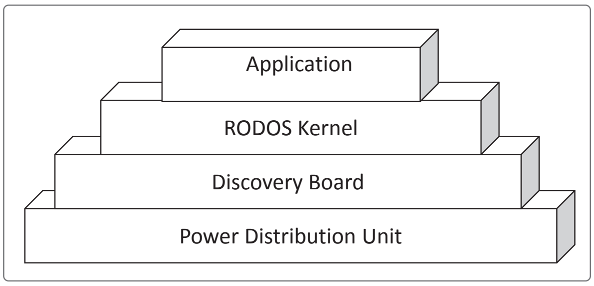


Fig. 1. Conceptual Diagram of the PDU module

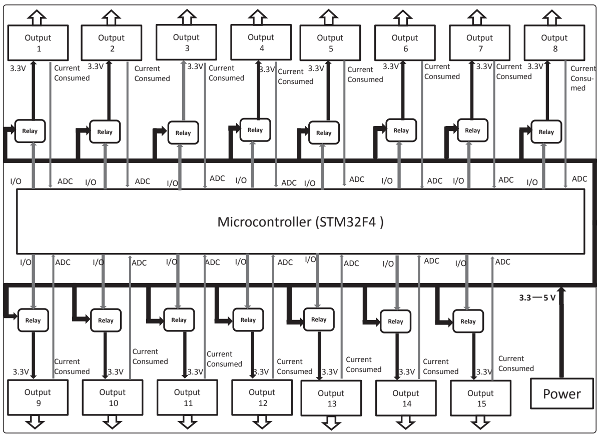


Fig. 2. Functional–Diagram of Power Distribution Unit

III. Circuit analysis

The electrical circuit for each node is shown in Fig. 3, This works as a programmable power supply for each node, The supply is controlled through the ARM controller. As shown in Fig. 3 the gate of MOSFET (Q3) is connected to the GPIO of the controller, this transistor is turned on through the controller and as a result the transistor Q2 switches on and the power is supplied to the corresponding node through its connector. Q2 is a power MOSFET which can provide maximum 1.2A current at 4.5 volt. Transistor Q3 acts as safety buffer and facilitates the quality power supply by protecting the main supply form surge current i.e sometimes load tries to draw more current than their normal ratings at the time when the circuit is first powered up, In this scenario the transistor Q3 protects the circuit from the damages that may be caused to the circuit. A current monitor is also attached to the output of Q2 to measure the current consumption of the related node. The output of the current monitor is also connected with the Analog–to–Digital input of the controller.

IV. Printed Circuit Board(PCB) for PDU

PCB for the Power Distribution unit is designed by using a Computer Aided Design software EAGLE 4.0. This PCB consists on two layers and the electrical components have been placed in both of the layers. The dimensions of the PCB are $97.155 * 80.01$ (mm). Power lines have been designed as thick as possible for the maximum heat dissipation. The complete layout, Placement and the routing of the components can be seen in the Fig. 4. The Printed Circuit Board was manufactured according to the original design and the front and back view of PCB is displayed in the Fig. 5a and 5b respectively. All the components and parts are solded on the PCB by hand. A complete PDU board is shown in Fig. 6

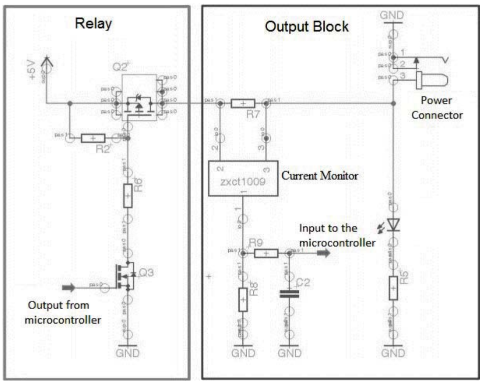


Fig. 3. Electrical Circuit of Each Node in PDU

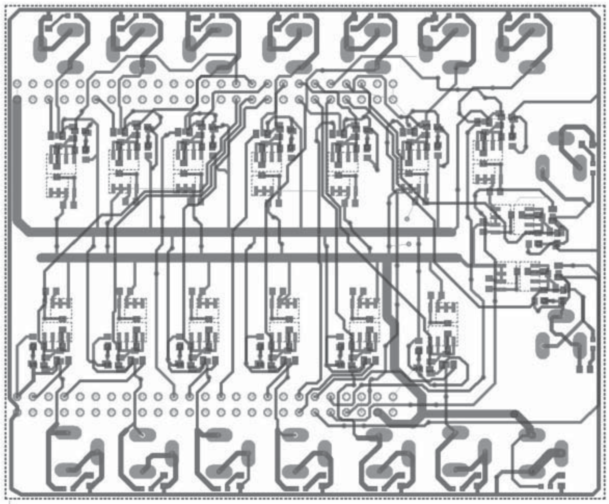
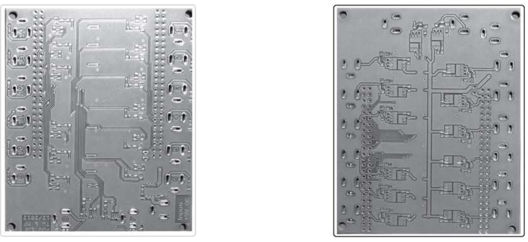


Fig. 4. Layout of Power Distribution Unit



(a)

(b)

Fig. 5. PCB: Power Distribution Unit (a) Front View (b) Back View

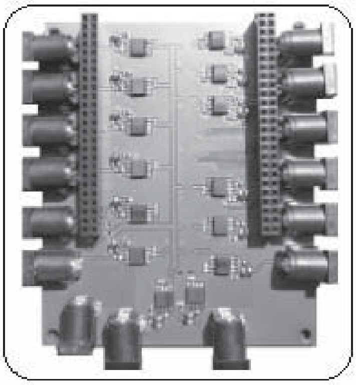


Fig. 6. PCB: Power Distribution Unit

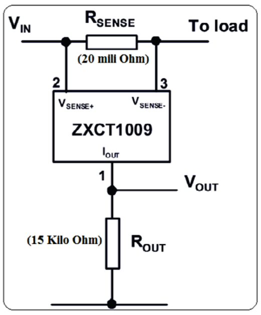


Fig. 7. Current Monitor [7].

V. Current Monitoring in PDU

Current Monitoring is performed through a current sensor ZXCT1009 [7], An Integrated Chip which measures the current on the high side of the circuit as shown in Fig. 7, The voltage developed on the high side of the shunt resistor is feeded into this IC and and it produces the proportional output current. As all the nodes attached to the Network does not require large amount of the current that's why the Shunt resistor in our case is selected as 20 mili Ohm which is quite enough to measure the current up-to 1 A. This selection of shunt resistor is based on the data sheet of this device. The output resistor of 15 kilo ohm is also selected in the same fashion. ***Note:***The Mathematical Model based on the data–sheet of zxct1009 [7] for the above circuit is described as follows:

$$\boxed{R_{out} = \frac{V_{out}}{V_{sense} * 0.01}} \tag{1}$$

$$V_{sense} = \frac{V_{out}}{15k * 0.01} \tag{2}$$

$$V_{sense} = \frac{V_{out}}{150} \tag{3}$$

$$\text{According to Ohm's Law:} \tag{4}$$

$$I_{sense} = \frac{V_{sense}}{R_{sense}} \tag{5}$$

$$\boxed{I_{sense} = \frac{V_{out}}{150 * 0.02}} \tag{6}$$

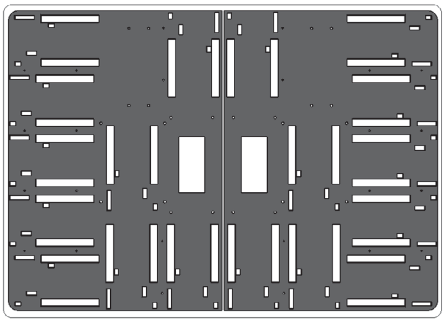


Fig. 8. Base Sheet for the Experimental Setup

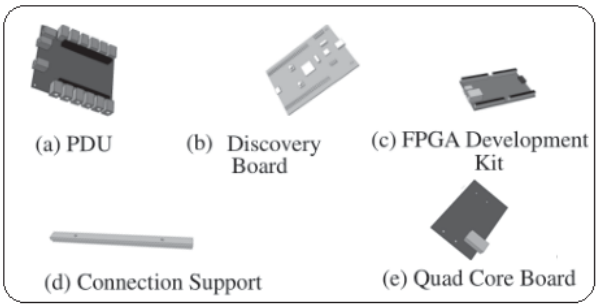


Fig. 9. 3D models of Components of the PDU

VI. Physical platform for the network

In the VIDANA project an experimental setup was required with twenty development kits, sixteen were stm32f407 based Discovery board, two FPGA kits and two multicore boards. These development kits will function as the computing nodes of VIDANA and two PDUs were planned to attach with this setup for the switching of all the development kits. This setup will be extensively used to test and explore the new ideas for the new computer systems for fault tolerance that's why a rugged test-bed was required. In order to have the fault free test–bed a complete design was implemented in Dassault's Systems's CATIA software. First the base sheet was designed with the 350mm * 500 mm dimension as shown in Fig. 8, These dimensions were taken according to the design of the container, Which was selected to contain the whole experimental Setup as depicted in Fig. 11. All the development kits which are planned to use in this setup have different dimensions and have different connectors and pins attached with the edges of the board that's why it was very necessary first to make a simulated design for the whole setup before manufacturing the base sheet. First the basic structure of all the development kits was designed as shown in the Fig. 9. In the end it was possible to uniformly place the development kits on the base sheet, All the discovery boards were placed next to each other although there were space shortage to maintain the uniformity in the setup it was decided to move the Quad–core development kit and FPGA board in the vertically upward direction which was very much possible because both of the boards have proper screw placements as shown in the Fig. 10.

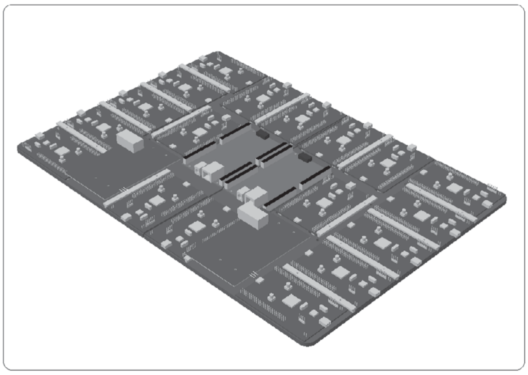
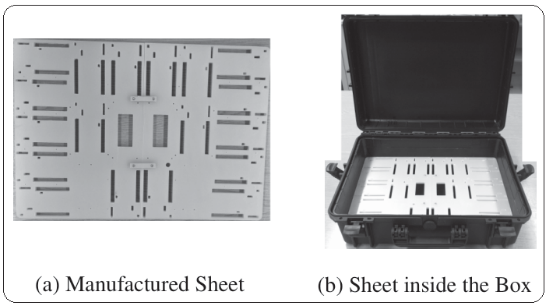


Fig. 10. 3D model of the Network



(a) Manufactured Sheet (b) Sheet inside the Box

Fig. 11. Physical Platform for the computing nodes of the Network

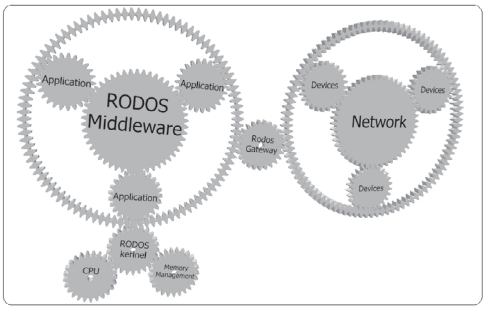


Fig. 12. Structure of RODOS

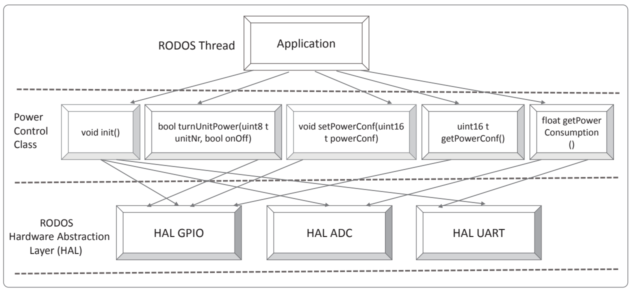

RODOS Thread
Application
Power Control Class
void init()
bool turnUnitPower(uint8 t unitNr, bool onOff)
void setPowerConf(uint16 t powerConf)
uint16 t getPowerConf()
float getPower Consumption ()
RODOS Hardware Abstraction Layer (HAL)
HAL GPIO
HAL ADC
HAL UART

Fig. 13. PDU Software Abstraction Layer

In the Fig. 10 the manufactured design of the base sheet is shown, The CNC machine in the Laboratory of Univerity of Wuerzburg can not handle the size of the sheet at one time therefore the sheet was designed in such a way that it should be constructed in two parts, Each part was designed as the mirror image of the other and in the end both parts were joined together as shown in the Fig. 10.

VII. Assembly

In the actual test-bed (experimental setup) it was made sure that there should be minimum visibility of the wires and connections in the front view that's why the power connections to each development kit is done from the back side of the setup and therefore the PDU is placed on the downward side of the test-bed, this has given the whole setup a very comprehensive and simple look.

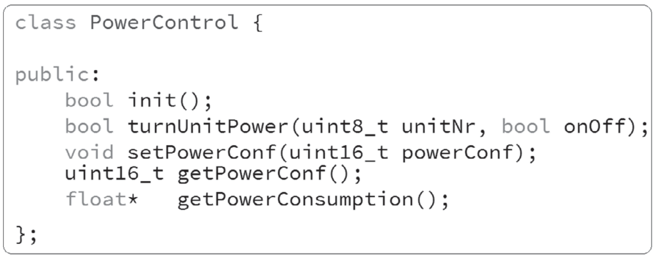
VIII. Software architecture of PDU

The software for the PDU is based on a real time operating systems RODOS (Real time Object Oriented Dependable Operating system). RODOS is jointly developed by the Central Core Avionics department of German Aerospace Center and Chair of Aerospace Information Technology University of Wuerzburg Germany [8]. RODOS is specifically developed for aerospace applications as it has minimal footprint but it is also very well suited to all applications that demand high dependability [9]. Software components in RODOS adjust each other to provide dependable computing [9], It is also shown in the Fig. 12. RODOS is an opensource software which is already being used in some satellites such as Technology testing vehicle (TET) and FireBIRD [10], [11].

A. On-Board Software

The Software of this PDU is developed under RODOS, RODOS provides a simple, specific and effective Hardware Abstraction Layer(HAL) for the different peripherals of a microcontroller, The primary focus of this HAL interface is to provide a less complex interface to the user to communicate with the different resources available in a micrcontroller and on the other hand if the designer wishes to change the current controller of PDU with the advanced–one where RODOS is ported too then it is not required to re–write the software of PDU. The On-Board Software is based on a control class which is publicly derived from RODOS thread class as shown in figure 13, This class provides all the functions to control the Switching of nodes, Monitoring the active and inactive nodes as well as the power consumption of each nodes, The interface of the this power contorl class is described in the Fig. 14, The details of all the data members of the control class is given as follow:

- **void** init():
  Provides the initialization of the UART, ADC and GPIO(Nodes).
- **bool** turnUnitPower(uint8_t unitNr, bool onOff) :
  Facilitates the switching for Specific Node.
- **void** setPowerConf(uint16_t powerConf):
  Switching Multiple Nodes Together.



```
class PowerControl {

public:
    bool init();
    bool turnUnitPower(uint8_t unitNr, bool onOff);
    void setPowerConf(uint16_t powerConf);
    uint16_t getPowerConf();
    float*   getPowerConsumption();
};
```

Fig. 14. Power Control Class

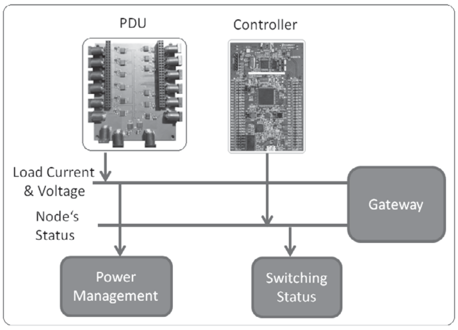


Fig. 15. RODOS Topics for Software and Hardware of the PDU

- **uint16_t** getPowerConf():
  Notifies the status of Nodes which are switched On.
- **float** getPowerConsumption():
  Measure the Power Consumption in Each Node in PDU

B. Communication Mechanism

All nodes are running their tasks and communicating with each other through Subscriber Publisher protocol. Publishers make messages public under a given topic. The Subscribers get all messages which are published under this topic. For this communication there is no difference in which node (computing unit or device) the publisher and subscribers are running. To establish a transfer path, both the publisher and the subscriber must share the same topic. A Topic is a pair consisting of a data-type and an integer representing a topic identifier. Both the software middle-ware and network switch (called middle-ware switch), interpret the same publisher/subscriber protocol in the same way [9]. In PDU there are two threads running to control the operation, One is responsible for the Load Current and voltages and other is managing the switching, The Electrical parameters such as the load current and voltages and Switching status of every nodes are published as separate topics on the Gateway as shown in Fig. 15 .

IX. Demonstrator for PDU

A Hardware demonstrator was developed with many computing nodes. This demonstrator resembles an Space Segment in the VIDANA project. As our PDU can only operate fifteen devices but the network required twenty devices to be attached together so two PDU units were used in this demonstrator. All the processing entities are joined together via a Field Programmable Gate Array (FPGA), i.e the harness is implemented by FPGA which offers a simple mechanism to avoid complex wiring. The number of active nodes in the network can be adapted via the PDU depending on the load. [1]. The hardware demonstrator is shown in Fig. 16.

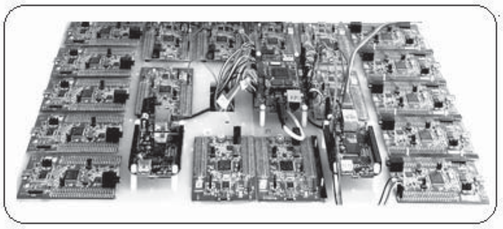


Fig. 16. Hardware Demonstrator for the Network of Computing Nodes

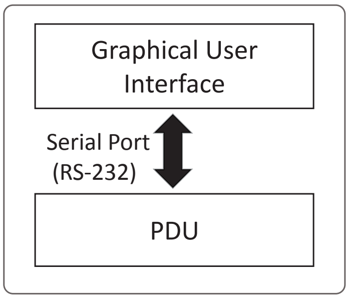


Fig. 17. Block Diagrame for Graphical User Interface for PDU

X. Graphical User Interface (GUI) for PDU

The Graphical User Interface for the PDU wad designed in C++ with the help of Qt Library [13] in Linux. A set of C++ wrapper classes has been developed to communicate between the network and the GUI application through RODOS Gateway. RODOS provides different Gateways protocols such as Serial, UDP, CAN, I2C etc, In this application a RS–232 based serial protocol was used because of the its simplicity as shown in Fig. 17. With the help of this GUI the PDU can be fully controlled, The information exchange between PDU and GUI takes place via the RODOS's Gateway as depicted in Fig. 18, Both PDU and GUI can translate the RODOS's network message and perform the requested task. All the computing nodes can be toggled with the control buttons on GUI. This interface also displays the amount of current consumption by each node as well as its switching status as shown in Fig. 19. For the VIDANA project different tasks has to run on different computing nodes.

XI. Conclusion

In this project a stand-alone and smart power distribution module was designed for a network of computing units which gives a complete solution from power distribution to the continuous analysis of all the electrical parameters of each individual node in a distributed network. The weight of PDU

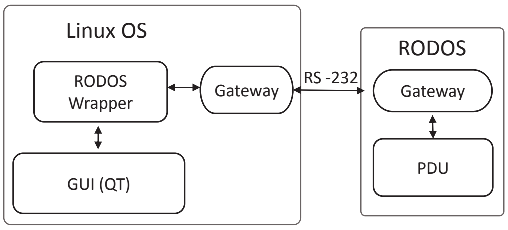
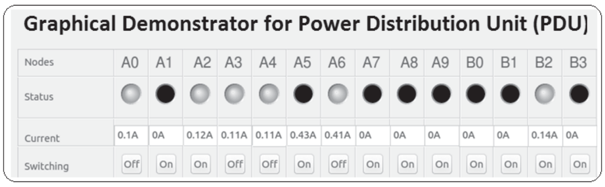


Fig. 18. Communication between PDU and GUI

is 68.7 gm and the discovery board weighs 36.6 gm. The power consumption of PDU is dependent on the discovery board which is controlling the PDU and according to the data sheet of discovery board [6], It needs 100 mA for normal operation with 5V as input so the power consumption of PDU is 0.5 Watt. The accuracy of measurement is dependent on the accuracy of ADC of microcontroller. The stm32f407 micrcocontroller has 3 ADC of upto 12 bit resolution. This PDU is a complete module with hardware and software which provides the power management for a network of computers. The Power Distribution Unit was controlled by a real time operating system RODOS which is running on an ARM Cortex M4 discovery board. The physical network was first modeled on a 3D design software to form a compact setup. The mechanical model of all the components was designed in a 3–D CAD software and then all the 3D–model of components were placed them together and then based on this 3d model the real design was implemented, This helped to develop a compact Laboratory based platform of the system. In the end by using the RODOS middle-ware and Gateway mechanism a simple Graphical user interface was developed to monitor the operation of the PDU and the network. This power distribution unit is a cost–effective, Simple, Compact and dependable approach to smartly deliver the quality electrical power to the multiple outlets and with the advanced feature of monitoring the continuous real time information of the system. Although this PDU was primarily designed for Aerospace applications but the approach behind this PDU can be easily tailored for different applications related to the embedded networking systems.



Graphical Demonstrator for Power Distribution Unit (PDU)

Nodes	A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	B0	B1	B2	B3
Status	○	●	○	○	○	●	○	●	●	●	●	●	○	●
Current	0.1A	0A	0.12A	0.11A	0.11A	0.43A	0.41A	0A	0A	0A	0A	0A	0.14A	0A
Switching	Off	On	On	Off	Off	On	Off	On	On	On	On	On	On	On

Fig. 19. Graphical User Interface for the Demonstrator

References

[1] Thomas Walter, Alexander Hilgarth, Tobias Mikschl, Sergio Montenegro *VIDANA: A fault tolerant approach for a distributed data management system in nano-satellites* , 10th Symposium on Small Satellites for Earth Observation, April 8 - 12, 2013, Berlin, Germany .

[2] Dr.Sergio Montenegro, Thomas Walter, *VIDANA: Data management system for nanosatellites* German Aerospace Congress Stuttgart, Germany, 10-12 september 2013.

[3] Rittal Gmbh, *Whitepaper Rittal PDU International*. http://www.rittal.com/

[4] EMERSON Network Power, *Power Distribution Units - PDU*. http://www.emersonnetworkpower.com/

[5] GOMspace, *NanoPower P60 System*. http://www.gomspace.com/

[6] *Discovery kit with* $STM32F407VG$ *MCU*. http://www.st.com/en/evaluation-tools/stm32f4discovery.html

[7] ZXCT1009: *High Side Current Monitor* from DIODES Incoporated http://www.diodes.com/datasheets/ZXCT1009.pdf

[8] Frank Dannemann and Sergio Montenegro, *Embedded Logging Frame-Work For SpaceCrafts*. DASIA (Data Systems In Aerospace), Porto, Portugal, 2013.

[9] *RODOS Real time kernel design for dependability. http://http://www.dlr.de/irs/en/Portaldata/46/ Resources/dokumente/produkte/rodos.pdf*

[10] *Technology testing vehicle*. http://www.dlr.de/rd/desktopdefault.aspx/tabid22743396_read5085

[11] *Bispectral InfraRed Optical System*. http://www.dlr.de/firebird/desktopdefault.aspx/tabid9104/15718_read-42455

[12] *NITROGEN6X: Embedded single board computer based on the popular i.MX6 ARM-Cortex A9 processor from NXP/Freescale* . https://boundarydevices.com/product/nitrogen6x-board-imx6-arm-cortex-a9-sbc/

[13] *Qt — Cross-platform software development for embedded & desktop*. https://www.qt.io/