



RELATÓRIO DE ESTÁGIO

Mestrado em Engenharia Electrotécnica

# **Electronic Interface Board to Monitor and Control CERN's Sputter Ion Pump**

MARCO ANDRÉ FELÍCIO RODA

Leiria, Março de 2015





MASTER INTERNSHIP REPORT

Electrical Engineering

# **Electronic Interface Board to Monitor and Control CERN's Sputter Ion Pump**

MARCO ANDRÉ FELÍCIO RODA

Internship performed under the guidance of Professor Luís Miguel Moreira Mendes and Professor Sérgio Manuel Maciel Faria of Escola Superior de Tecnologia e Gestão of Instituto Politécnico de Leiria.

Leiria, March 2015



# Acknowledgments

Luís Mendes,  
Gregory Pigny,  
Paulo Gomes,  
Sérgio Faria,  
the members of the TE-VSC-ICM section at CERN  
and all PROFIS project contributors,

*Thank you for your help and support!*

I dedicate this thesis to my Mom.  
Marco



# Abstract

This report presents the work developed during the master internship at the European Organization for Nuclear Research (CERN). The described work resulted from two initial objectives: improvement of the control system of vacuum sectors and the development of a new Profibus-DP intelligent slave interface board for sputter ion pump controllers. Since the objectives were distinct, the work was split into two different projects.

In the first project, the control system for vacuum sectors of Large Hadron Collider, Super Proton Synchrotron and Complex Proton Synchrotron was improved. A new control firmware based in VHDL was developed, tested and upgraded for the control cards.

The second project presented in this report involved the development of an electronic card for sputter ion pump controllers. Currently, the communication between the Profibus network and the ion pump controllers is performed by a remote input-output station based on the ET200 module from Siemens. The operation and status information of the ion pumps and their controllers provided to the network are limited since the remote input-output station does not acquire all the available signals. Moreover, the physical connections between the remote input-output station and each ion pump controller is done with a dedicated cable, which is prone to connection malfunctions. In order to reduce the complexity, improve the signal integrity and upgrade the monitoring and control of the ion pumps system, a new Profibus-DP intelligent slave interface was developed. It aims to provide direct Profibus connection to the ion pump controller and to give to the Profibus master access to all its control signals, improving the controllers usability and flexibility. In this report, the card topology, its internal modules, the firmware, the LabVIEW application and assessment tests are presented and described.

**Keywords:** Profibus-DP, industrial automation and control, embedded system, signal acquisition, microcontroller.



# Resumo

Este relatório descreve o trabalho desenvolvido no âmbito do estágio de Mestrado em Engenharia Electrotécnica na Organização Europeia de Pesquisa Nuclear (CERN). O trabalho aqui documentado resulta de dois objetivos iniciais: aperfeiçoamento do sistema de controlo dos setores de vacuum e desenvolvimento de uma nova placa electrónica para o controlador de bombas iónicas.

No primeiro projeto, o sistema de controlo para os setores de vacuum dos aceleradores Large Hadron Collider, Super Proton Synchrotron e Complex Proton Synchrotron foi revisto e melhorado. Foi desenvolvido, testado e implementado um novo controlador digital baseado em lógica combinacional.

O segundo projeto apresentado neste documento, envolveu o desenvolvimento de uma nova placa electrónica para controladores de bombas iónicas. Atualmente, a comunicação entre a rede Profibus e o controlador da bomba iónica é estabelecida através de um módulo entradas/saídas da Siemens. A operação e a leitura do estado do par controlador-bomba iónica são limitados, uma vez que o módulo entrada/saída não adquire toda a informação disponibilizada pelo controlador. Além disso, as conexões entre este módulo e cada controlador de bomba iónica são executadas através de um cabo dedicado, que pode proporcionar falhas na comunicação. Com o objetivo de reduzir a complexidade, melhorar a aquisição e quantidade dos sinais do controlo e melhorar o sistema de monitorização e controlo das bombas iónicas, foi desenvolvido um novo controlador baseado em Profibus-DP. O principal objetivo deste controlador é fornecer a cada bomba iónica um endereço único na rede Profibus, fornecendo um controlo melhorado e mais flexível aumentando a sua eficiência. Neste relatório, é apresentada a topologia da placa electrónica desenvolvida, os seus módulos internos, o firmware, a aplicação LabVIEW e o procedimento de testes.

**Palavras chave:** Profibus-DP, controlo e automação industrial, sistemas embebidos, aquisição de sinal, microcontrolador.



# Contents

<b>Acknowledgments</b>	<b>i</b>
<b>Abstract</b>	<b>iii</b>
<b>Resumo</b>	<b>v</b>
<b>Contents</b>	<b>x</b>
<b>List of Figures</b>	<b>xi</b>
<b>List of Tables</b>	<b>xv</b>
<b>List of Abbreviations</b>	<b>xvii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 The CERN Accelerator Chain . . . . .	1
1.2 The LHC Vacuum System . . . . .	2
1.2.1 Vacuum Controls Architecture . . . . .	3
1.2.2 Vacuum Instrumentation . . . . .	4
1.2.3 Sputter Ion Pump Controller . . . . .	5
1.3 Motivation and Objectives . . . . .	6
1.4 Thesis Outline . . . . .	7
1.5 Scientific Contributions . . . . .	8
<b>2 Firmware Upgrade on Vacuum Sector Valve Control Unit</b>	<b>9</b>
2.1 Vacuum Sectors of Vacuum . . . . .	9
2.2 Interlocks . . . . .	10
2.3 Control System Architecture . . . . .	12
2.4 SVCU Card . . . . .	13

2.5	Control Logic . . . . .	17
2.6	Modelsim Simulations . . . . .	19
<b>3</b>	<b>Sputter Ion Pump Controller</b>	
	<b>Architecture</b>	<b>21</b>
3.1	Present VPI Controller Architecture . . . . .	21
3.1.1	VPI Controller Modules . . . . .	22
3.1.2	Present Control Architecture . . . . .	22
3.1.3	Present Control Signals . . . . .	23
3.1.4	Ionization Current Measurement and Status . . . . .	24
3.2	Proposed VPI Supply Architecture . . . . .	25
3.2.1	Proposed Control System . . . . .	26
3.2.2	Hand-Unit . . . . .	26
3.2.3	PROFIS Signals . . . . .	28
<b>4</b>	<b>Board Design and Development</b>	<b>33</b>
4.1	Board Architecture . . . . .	33
4.2	Profibus-DP Interface . . . . .	34
4.2.1	Profibus Transceiver . . . . .	35
4.2.2	Galvanic Isolation . . . . .	36
4.3	Controller - Profibus ASIC Module . . . . .	36
4.3.1	Profibus ASIC . . . . .	36
4.3.2	Market Survey . . . . .	36
4.3.3	VPC3+S Overview . . . . .	39
4.3.4	VPC3+S RAM . . . . .	40
4.3.5	ASIC Mode Registers . . . . .	41
4.3.6	Interrupt Controller . . . . .	42
4.3.7	SPI Interface . . . . .	42
4.3.8	IOs of the Profibus ASIC Module . . . . .	45
4.4	Controller - Microcontroller Module . . . . .	46
4.4.1	General Overview of PIC18F4680 . . . . .	46
4.4.2	Ports Distribution . . . . .	46
4.4.3	IOs of the Microcontroller Module . . . . .	48
4.5	USART . . . . .	50

4.6	IO Module . . . . .	51
4.6.1	Digital Inputs . . . . .	52
4.6.2	Digital Outputs . . . . .	53
4.6.3	Analog Signals . . . . .	54
4.6.4	IOs of the IO Module . . . . .	55
4.7	Power Supply Module . . . . .	56
4.7.1	IOs of the Power Supply Module . . . . .	57
4.8	PROFIS . . . . .	57
4.8.1	Top Level Schematic . . . . .	57
4.8.2	PCB and Enclosure Box . . . . .	59
<b>5</b>	<b>Firmware Implementation</b>	<b>63</b>
5.1	Application Firmware of the PROFIS . . . . .	63
5.1.1	Main Program Flowchart . . . . .	64
5.1.2	Microcontroller Initialization . . . . .	65
5.1.3	ASIC Initialization . . . . .	69
5.1.4	Profibus-DP State Machine . . . . .	69
5.1.5	Data Exchange State Machine . . . . .	71
5.2	SIMATIC STEP 7 Interface . . . . .	73
5.2.1	Hardware Configuration . . . . .	73
5.2.2	GSD file . . . . .	74
5.2.3	Simatic STEP7 I/O Data . . . . .	75
5.3	LabVIEW Interface . . . . .	76
5.3.1	State Machine . . . . .	76
5.3.2	Case Structure . . . . .	77
5.3.3	FOR loop . . . . .	77
5.3.4	Application Panel . . . . .	78
<b>6</b>	<b>Conclusion and Future Work</b>	<b>81</b>
6.1	Conclusions . . . . .	81
6.2	Future Work . . . . .	82
	<b>Bibliography</b>	<b>83</b>
<b>A</b>	<b>PROFIS - Price Estimation</b>	<b>85</b>

A.1 CERN . . . . .	85
A.2 External Company . . . . .	85
<b>B PROFIS Schematics</b>	<b>87</b>
<b>C Designed PCB</b>	<b>95</b>
C.1 First Prototype . . . . .	95
C.2 Final Prototype . . . . .	95

# List of Figures

1.1	Accelerator chain of CERN. . . . .	2
1.2	LHC vacuum controls architecture [2]. . . . .	4
1.3	Block diagram of the VPI Controller. . . . .	6
2.1	Vacuum sector of LHC. . . . .	10
2.2	Interlock system of LHC. . . . .	11
2.3	Interlock system of SPS. . . . .	11
2.4	Interlock system of SPS. . . . .	11
2.5	SVCUs general architecture. . . . .	12
2.6	PLC communication and MUX. . . . .	13
2.7	SVCU card. . . . .	15
2.8	SVCU card front panel of LHC. . . . .	15
2.9	SVCU card front panel of SPS. . . . .	16
2.10	SVCU card front panel of CPS. . . . .	16
2.11	Control logic of SVCU cards (a) logic gates; (b) VHDL. . . . .	17
2.12	SVCU card front panel of CPS. . . . .	19
3.1	VPI architecture of the LHC. . . . .	22
3.2	VPI control architecture. . . . .	23
3.3	Signals of the current VPI control architecture. . . . .	23
3.4	0-10 V feedback control signal [2]. . . . .	24
3.5	New VPI control architecture. . . . .	25
3.6	Proposed VPI control architecture. . . . .	26
3.7	PROFIS IO connection. . . . .	28
3.8	Signals of the proposed architecture. . . . .	30
4.1	Block diagram of the new PROFIS board. . . . .	34

4.2	Schematic of the Profibus-DP interface module. . . . .	34
4.3	Oscilloscope screenshots of the Profibus communication (a) 2.55 <i>ms</i> /div and (b) 50 $\mu$ <i>s</i> /div. . . . .	35
4.4	Profibus-DP ASIC market survey [13] [14]. . . . .	37
4.5	Solutions for the Profibus ASIC module. . . . .	38
4.6	Internal modules of the VPC3+S [15]. . . . .	39
4.7	Physical buffer address of the RAM (a) 2kB mode and (b) 4 kB mode. . .	41
4.8	Block diagram of the VPC3+S interrupt controller. . . . .	42
4.9	Interface between the microcontroller and the VPC3+S. . . . .	43
4.10	Oscilloscope screenshots of the SPI communication at 156 kHz (a) 25 $\mu$ <i>s</i> /div: (b) 10 $\mu$ <i>s</i> /div. . . . .	44
4.11	Oscilloscope screenshots of the SPI communication at 2.5 MHz (a) 5 $\mu$ <i>s</i> /div: (b) 1 $\mu$ <i>s</i> /div. . . . .	45
4.12	ASIC module IOs. . . . .	45
4.13	Microcontroller IO ports. . . . .	48
4.14	Microcontroller module IOs. . . . .	49
4.15	Schematic of the USB-RS232 IC converter. . . . .	51
4.16	USART module IOs. . . . .	51
4.17	Circuit used to acquire the digital input PROTECTION. . . . .	52
4.18	Circuit used to acquire the digital input LEVEL 1. . . . .	53
4.19	Turn on the high-voltage. . . . .	54
4.20	Turn off the high-voltage. . . . .	54
4.21	Conditioning circuit of the analog input IONIZATION MEASURE. . . . .	55
4.22	Circuit used to acquire the analog input LEVEL 2 THRESHOLD. . . . .	55
4.23	Inputs and outputs of the IO module. . . . .	56
4.24	DC-DC converters. . . . .	57
4.25	Power-supply module IOs. . . . .	57
4.26	Main schematic of the PROFIS board. . . . .	58
4.27	PCB layer stack. . . . .	59
4.28	PCB organization. . . . .	60
4.29	PROFIS final PCB prototype. . . . .	60
4.30	Designed plastic box for the PROFIS board. . . . .	61
5.1	Master-Slave communication principle. . . . .	64

5.2	Main program flowchart of the software application. . . . .	64
5.3	ASIC VPC3+S initialization. . . . .	70
5.4	The state machine of Profibus-DP slave [3]. . . . .	71
5.5	Data exchange state. . . . .	72
5.6	Data processing (a) read and process input data; (b) receive output data. . . . .	73
5.7	Simatic STEP7 hardware configuration. . . . .	74
5.8	Simatic STEP7 network configuration. . . . .	75
5.9	GSD symbol of the VPI controller. . . . .	75
5.10	LabVIEW application state machine. . . . .	77
5.11	LabVIEW state machine (a) Idle state; (b) Refresh state. . . . .	78
5.12	LabVIEW CASE structures (a) HV on button; (b) HV off button; (c) stop application button; (d) time out. . . . .	79
5.13	Example of FOR cycles. . . . .	79
5.14	LabVIEW application panel (a) high-voltage off and (b) high-voltage on. . . . .	80
B.1	Main schematic of PROFIS. . . . .	88
B.2	Profibus-DP interface module. . . . .	89
B.3	Microcontroller module schematic. . . . .	90
B.4	UART module schematic. . . . .	91
B.5	Power-supply module schematic. . . . .	92
B.6	I/O module schematic. . . . .	93
C.1	First PCB prototype. . . . .	95
C.2	Final PCB prototype. . . . .	95
C.3	Power planes. . . . .	96
C.4	Ground planes. . . . .	96



# List of Tables

1.1	Output voltage from the VPI controller. . . . .	6
3.1	Hands-unit signals. . . . .	27
3.2	Hands-unit used signals in PROFIS. . . . .	29
3.3	PROFIS to VPI controller signals. . . . .	30
3.4	VPI controller to PROFIS signals. . . . .	31
4.1	2k RAM organization [15]. . . . .	40
4.2	Microcontroller SPI pins. . . . .	43
4.3	VPC3+S pins for the SPI interface. . . . .	43
4.4	ASIC module IOs description. . . . .	46
4.5	Microchip PIC18F4680 main features. . . . .	47
4.6	PIC18F4680 IO ports distribution. . . . .	47
4.7	IO signals description of the microcontroller module. . . . .	50
5.1	TRIS registers configuration. . . . .	65
5.2	Timer 0 control register. . . . .	66
5.3	Microcontroller SPI pins. . . . .	68
5.4	Microcontroller USART pins. . . . .	68
5.5	Simatic STE7 IO data. . . . .	76
A.1	PROFIS price estimation (Unit size: 71.86x145.29mm) . . . . .	85
A.2	PROFIS price estimation (Unit size: 71.86x145.29mm) . . . . .	85



# List of Abbreviations

ADC	Analogue to Digital Converter, p. 46
ASIC	Application-Specific Integrated Circuit, p. 33
BIS	Beam Interlock System, p. 10
CPLD	Complex Programmable Logic Device, p. 12
DI	Digital Input, p. 12
DO	Digital Output, p. 12
DS	Data Server, p. 3
EEPROM	Electrically Erasable Programmable Read-Only Memory, p. 47
EMI	Electromagnetic Environment, p. 7
HMI	Human Machine Interface, p. 3
JTAG	Joint Test Action Group, p. 50
LED	Light-Emitting Diode, p. 13
LEIR	Low Energy Ion Ring, p. 2
LHC	Large Hadron Collider, p. 1
LSB	Least Significant Bit, p. 54
MISO	Master Input Slave Output, p. 43
MOSI	Master Output Slave Input, p. 43
MSSP	Master Synchronous Serial Port, p. 46
PCB	Printed Circuit Board, p. 14
PLC	Programmable Logic Controller, p. 3

PLC	Programmable logic controller, p. 12
PLD	Programmable Logic Devices, p. 3
PROFIS	Profibus Intelligent Slave, p. 7
PSB	Proton Synchrotron Booster, p. 1
PS	Proton Synchrotron, p. 1
RAM	Random Access Memory, p. 33
RTS	Request To Send, p. 35
R&D	Research and Development, p. 3
SCADA	Supervisory Control And Data Acquisition, p. 3
SCK	Serial Clock, p. 43
SDI	Serial Data In, p. 43
SDO	Serial Data Out, p. 43
SPI	Serial Peripheral Interface, p. 38
SPS	Super Proton Synchrotron, p. 1
SS	Slave Select, p. 43
SVCU	Sector Valve Control Unit, p. 9
TMP	Turbo Molecular Pumps, p. 4
USART	Universal Synchronous/Asynchronous Receiver/Transmitter, p. 46
USB	Universal Serial Bus, p. 50
VCU	Valve Control Unit, p. 12
VGF	Full-Range Gauge, p. 4
VGI	Bayard-Alpert gauges, p. 3
VGM	Membrane Gauges, p. 4
VGP	Penning Gauges, p. 3
VGR	Pirani Gauges, p. 3

VHDL	Very-High-Speed-Integrated-Circuits Hardware Description Language, p. 6
VIES	DC Power Supplies for Solenoids, p. 3
VPGF	Fixed Pumping Groups, p. 3
VPGM	Mobile Trolley, p. 4
VPI	Sputter Ion Pump, p. 5
VPS	Sublimation Pumps, p. 4
VVR	Roughing Valves, p. 5
VVS	Sector Gate Valves, p. 3



# Chapter 1

## Introduction

This chapter presents a brief description of the CERN's accelerator chain. The Large Hadron Collider vacuum system, its architecture and equipments are also presented and described. The controller of the sputter ion pump is explained since it is an essential equipment of the LHC vacuum system and part of the work presented in this this document. This chapter presents the motivation and initial objectives of the developed work. Finally, the chapter ends with an overview of the structure of this document.

### 1.1 The CERN Accelerator Chain

CERN [1] is a particle physics laboratory based in Geneva, Switzerland, being the world's largest scientific research institute to study the basic constituents of matter, the fundamental particles. It is well known for its largest accelerator, the Large Hadron Collider (LHC). A series of accelerators work together to push particles to near the speed of light. The accelerator chain of CERN (Figure 1.1) is a succession of machines that accelerate particles to increasingly higher energies. Each machine boosts the energy of a beam of particles, before injecting the beam into the next machine in the sequence. In the LHC, the last element in this chain, particle beams are accelerated up to the record energy of 4 TeV per beam. Most of the other accelerators in the chain have their own experimental halls where beams are used for experiments at lower energies.

The proton source is a simple bottle of hydrogen gas. An electric field is used to strip hydrogen atoms of their electrons to yield protons. Linac 2, the first accelerator in the chain, accelerates the protons to the energy of 50 MeV. The beam is then injected into the Proton Synchrotron Booster (PSB), which accelerates the protons to 1.4 GeV, followed by the Proton Synchrotron (PS), which pushes the beam to 25 GeV. Protons are then sent to the Super Proton Synchrotron (SPS) where they are accelerated to 450 GeV. The protons are finally transferred to the two beam pipes of the LHC. The beam in one pipe

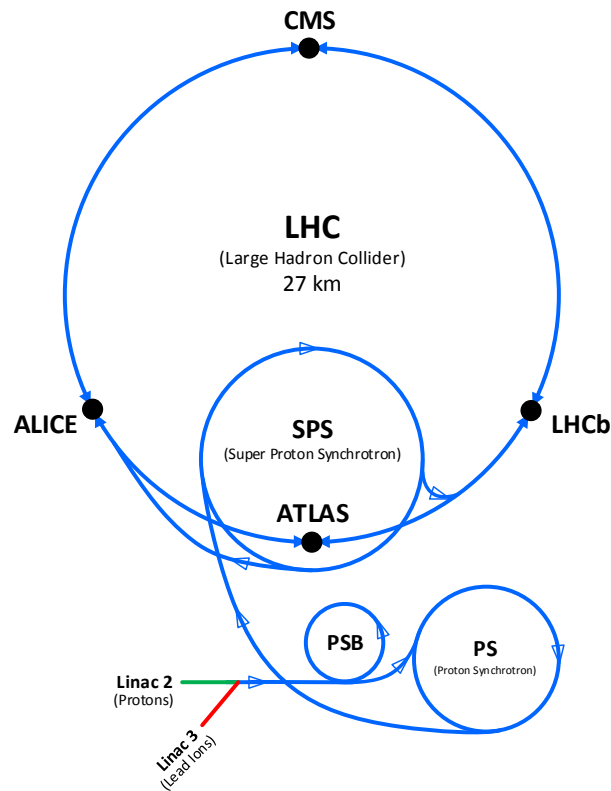


Figure 1.1: Accelerator chain of CERN.

circulates clockwise while the beam in the other pipe circulates anticlockwise. It takes 4 minutes and 20 seconds to fill each LHC ring, and 20 minutes for the protons to reach their maximum energy of 4 TeV. Beams circulate for many hours inside the LHC beam pipes under normal operating conditions. The two beams are brought into collision inside four detectors: ALICE, ATLAS, CMS and LHCb, where the total energy at the collision point is equal to 8 TeV [1] .

Protons are not the only particles accelerated in the LHC. Lead ions start from a source of vaporized lead and enter Linac 3 before being collected and accelerated in the Low Energy Ion Ring (LEIR). Then, the lead ions follow the same route as the protons in order to maximize the beam energy.

## 1.2 The LHC Vacuum System

To minimize the interactions between the accelerating beams and the residual gases, and thus maximize the beam lifetime, the beam pipes of all accelerators must be pumped down to a suitable vacuum level. The LHC at CERN is a 27 km proton accelerator-collider, with two separate beam pipes that merge at the interaction points, as stated before. The static vacuum pressure is of about  $10^{-11}$  mbar, although it can rise up to  $10^{-8}$  mbar due

to beam dynamic effects. Feeding the LHC, the SPS and its transfer lines add 16 km of beam pipe vacuum in the range of  $10^{-7} - 10^{-9}$  mbar. Going back through the chain, the Proton Synchrotron (PS), its Booster and the Linacs (L2, L3) add a further 2 km. To guarantee the best vacuum level in all the CERN accelerator complex, it is necessary to use an automation system capable to monitor and control all the vacuum field equipments and detect any possible anomaly [2].

### 1.2.1 Vacuum Controls Architecture

Automation plays a crucial role in many complex industrial and Research and Development (R&D) installations. It has attracted substantial effort from researchers in both academic and industrial communities to provide efficient scientific and engineering solutions to turn the management, monitoring and control of complex systems simpler. One of the R&D institutions that has a large scale automation solution is the European Organization for Nuclear Research. The Profibus-DP protocol [3,4] is largely used at CERN in the vacuum controls architecture. It is a fieldbus included in the EN50254 European Standard [5] and in the IEC 61158 International Standard [6].

The vacuum controls used in the CERN accelerator complex have been upgraded and homogenized to a Programmable Logic Controller (PLC) based architecture using the Siemens S7-400 series. In Figure 1.2, the LHC vacuum controls architecture is depicted. The Human Machine Interface (HMI) is a Supervisory Control And Data Acquisition (SCADA), built with WinCC-OA (e.g. PVSS). Geographically limited accelerators or installations are controlled by a single PLC. Wider machines have one PLC at each underground service area (e.g. 28 for LHC). Independently of the number of PLCs used, there is only one PVSS Data-Server (DS) per accelerator complex. The PLCs and the DS communicate through Ethernet in a protected and restricted “Technical Network” [2].

In the underground service area, a PLC master accesses the field equipments through their controllers. Often they are intelligent since they have an embarked microprocessor or other Programmable Logic Devices (PLDs). When equipped with the corresponding interface, they can communicate with the PLC via Profibus; this minimizes the complexity and price of cabling and also allows a wider exchange of information and configuration parameters. Bayard-Alpert gauges (VGI) controller (Volotek), Pirani gauges (VGR) and Penning gauges (VGP) controller (TPG300), DC power supplies for solenoids (VIES) and also fixed pumping groups (VPGF) controller (PLC slave and its turbo pump controller) are already equipped with Profibus connection. On the other hand, the Sector Gate Valves (VVS) controller is directly connected to digital IOs on the PLC using a dedicated bus and protocol. The sputter ion Pump controller is controlled through a remote IO station from Siemens, connected to the Profibus network, the same with the sublimation pumps

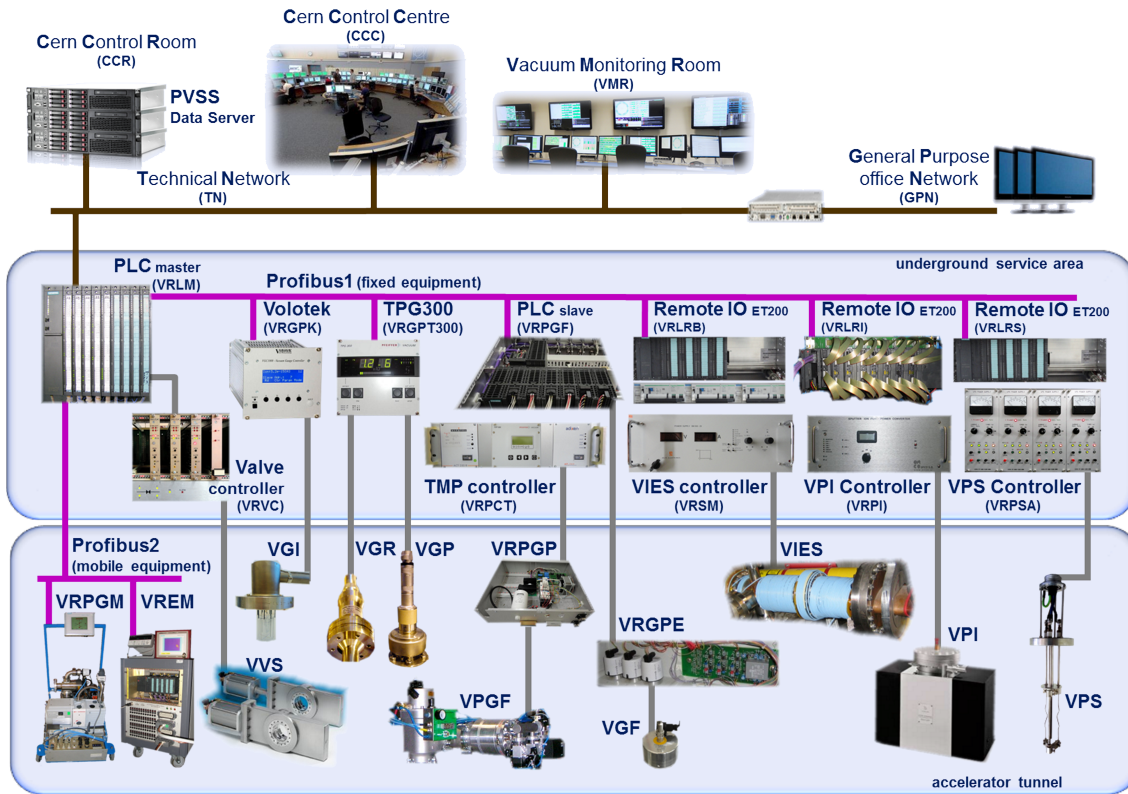


Figure 1.2: LHC vacuum controls architecture [2].

(VPS) controller [2].

### 1.2.2 Vacuum Instrumentation

To generate, maintain and measure the best level of vacuum, several instruments are required: active and passive gauges, pumps and valves. All these equipments are located in the accelerator area (tunnel), as shown in Figure 1.2. Various types of vacuum gauges are used in function of the vacuum pressure range to be measured. Some of these gauges are: membrane gauges (VGM), thermal conductivity or VGR, cold cathode ionization or VGP, full-range gauge (VGF) and hot cathode ionization or GI. These vacuum gauges can be used to measure pressures in the range of 1 to  $10^{-12}$  mbar [2].

The vacuum pumps are used to evacuate the internal gases of the accelerator ring pipes. Primary pumps are employed to create pressures of the order of  $10^{-3}$  mbar. They are also used as a backing pump for turbo molecular pumps (TMP), who are effective in the range of  $10^{-3}$  to  $10^{-10}$  mbar. Often they are both assembled and controlled together as a pumping group. If their usage is temporary, they can be embarked in a mobile trolley (VPGM), together with the powering and control electronics. For long-term pumping (VPGF), only the pumps and powering circuitry are left close the accelerator, while all

the radiation-sensitive electronics is kept in safe areas. When needed to reach ultra-high vacuum, Sputter Ion Pumps (VPIs) must be used and are effective in the range of ( $10^{-5}$  to  $10^{-11}$  mbar).

To isolate a pumping group from the pumped volume (vacuum sector), roughing valves are used (VVR). To prevent leak propagation or to allow mechanical intervention, VVS are employed [2].

### 1.2.3 Sputter Ion Pump Controller

The VPIs are mainly used when ultra-high levels of vacuum (i.e. ultra-low pressure) need to be reached in the accelerators pipes. The vacuum is achieved through an ionization process that submits the gases in the pipes to a 6 kV potential difference (between the anode and cathode plates of the VPI). Under normal operation conditions, the ionization current of the VPI, which is equal to its consumption current, is proportional to the pressure in the pipes. This operation characteristic of the VPIs turns the pressure measurement procedure very simple since the internal pressure can be determined by

$$P = K \cdot I_{ion}, \quad (1.1)$$

where  $P$  is the pressure in [mbar],  $K$  is the constant of proportionality given in [mbar/A] and  $I_{ion}$  is the ionization current expressed in [A]. The coefficient  $K$  is a characteristic parameter of each type of ion pump [7].

The block diagram of the VPI controller is shown in Figure 1.3. As can be seen, the VPI is composed by three essential modules: the high voltage module, the ionization current acquisition and signal conditioning module and the module that generates the status and thresholds information. The VPI current consumption flowing through a shunt resistor produces a voltage drop, which is the input of a logarithmic amplifier that aims to confine the five decades of the ionization current range (i.e  $1.0^{-6}$  to  $1.0^{-1}$  [A]) to a 0-6 [V] voltage output.

The status information and the voltage related to the ionization current are combined in the output voltage signal  $U$ . Each VPI status information is coded with a given voltage range. For example, when the high voltage module is off, the output voltage is between 7.5 and 8 V. To accommodate all the status information, a voltage range of 4 V is used. Thus, the output voltage  $U$  can vary between 0 and 10 V, as indicated in Table 3.2. This voltage is acquired by the remote IO module and sent to the Profibus master. The threshold levels inform the user whether the trimmed levels related to the value of the ionization current have been reached or not. Note that these data are not provided to the PLC master in the actual implementation.

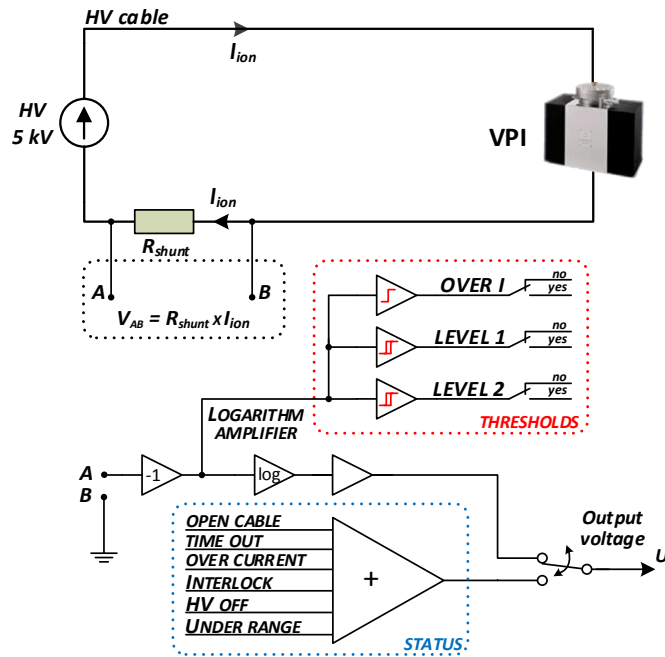


Figure 1.3: Block diagram of the VPI Controller.

Table 1.1: Output voltage from the VPI controller.

Output Voltage [V]	VPI Status
$0.0 \leq U < 6.0$	Ionization current
$6.0 \leq U < 7.5$	Under range
$7.5 \leq U < 8.0$	High voltage off
$8.0 \leq U < 8.5$	Interlock
$8.5 \leq U < 9.0$	Over current
$9.0 \leq U < 9.5$	Time out
$9.5 \leq U \leq 10$	Open cable

### 1.3 Motivation and Objectives

The vacuum sector valves of CERN's accelerators are controlled by digital electronic cards designed in 2006. The controller firmware is a combinational schematic description, which was developed using an obsoleted software version running only in Window XP. For the first project, the aim is to assure the portability from a schematic description designed with an old software version (Quartus v6.0) into a Very-High-Speed-Integrated-Circuits Hardware Description Language (VHDL) with the last version of Quartus running on windows 7 while keeping the same functionalities. This is the first step for further improvements of the SVCU cards: 1) Understand the existing control logic for vacuum sector valves in LHC, SPS and CPS; 2) Rewrite the functionalities in VHDL and perform digital simulations; 3) Implement them in the CPLD and test; 4) Add functionalities if necessary

and perform simulation/test; 5) Write detailed technical reports and documentation.

One of the CERN's systems that uses an automation network with masters and slaves interconnected by a Profibus-DP network is the sputter ion pumps sub-system, which is very important to the correct operation of the accelerators. Currently, the cabling of the VPI sub-system is complex and is routed through a rich electromagnetic environment, which is an important issue for the correct acquisition of the analog signals transmitted over the cables. To mitigate the electromagnetic interference (EMI) problems, an improved architecture is proposed. To implement it, it was necessary to develop a dedicated Profibus intelligent slave (PROFIS) for the VPI controllers, replacing the existent remote IO station. Therefore, the following goals have been presented for the second project: 1) Market survey of commercial chips allowing Profibus-DP communication; 2) Design the electronic circuit around the selected component (ADC converter, IO signals, level matching, isolation); 3) Analysis/simulation (SPICE) parts of the circuit; 4) Schematic/PCB design with Altium Designer; 5) Establish the bill of material, and produce one prototype (PCB and component mounting); 6) Establish the test procedure, test the prototype circuit functionalities with LabVIEW and PLC Profibus-DP connection; 7) Write detailed technical report and documentation.

## 1.4 Thesis Outline

The present report is organized in six chapters and three appendices. Chapter 1 introduces CERN, its accelerators and the vacuum systems. The LHC controls architecture is described and all vacuum equipments are presented. The sputter ion pump sub-system, its control architecture and related signals are also described.

In Chapter 2, the control architecture for vacuum sectors of LHC, SPS and CPS is presented. The developed upgrade of the firmware to improve the control of the vacuum sectors are described and digital simulations are explained.

Chapter 3 presents the current and proposed architecture of the VPI controllers. The advantages of the new architecture are described and the new acquired signals are explained. In Chapter 4, the design and hardware development of the Profibus-DP slave for sputter ion pumps is presented. All the selected components and their features are described. The schematics and PCB development are also presented and explained. The developed software with all the state machines and configuration registers of the microcontroller and Profibus-DP ASIC are explained in Chapter 5.

Finally, Chapter 6 presents some conclusions about the achieved results and future work.

## 1.5 Scientific Contributions

A paper has been submitted to the 10<sup>th</sup> IEEE International Symposium on Industrial Embedded Systems (SIES'2015).

# Chapter 2

## Firmware Upgrade on Vacuum Sector Valve Control Unit

In this chapter, the control architecture of the vacuum sector valves of the LHC, SPS and CPS is presented. The PLC communication and how it multiplexes the different equipments that are distributed and placed near the accelerators is explained. The interlock system for each machine, the upgraded firmware of the sector valve control unit (SVCU) are described and analysed. Finally, the control logic is explained and the performed simulations are depicted.

### 2.1 Vacuum Sectors of Vacuum

The beam pipes of the CERN's accelerators complex are divided by sectors. The aim is to provide isolation between sequential sectors, preventing leak propagation in case of a failure or to allow mechanical interventions. For this purpose, and depending on the accelerator, the interlock system and its logic control is different, however their topology and equipment are similar. In Figure 2.1, one vacuum sector of the LHC is shown.

Each vacuum sector of the LHC, SPS and CPS is organized as depicted in Figure 2.1. There are always two interlocks systems for each sector gate valve (VVS). The interlock system is composed by several sensors (e.g. pressure gauges and sputter ion pumps) and depending on their state an interlock signal might be generated informing that the pressure level in a given accelerator is not under the desired vacuum level. The sector valve control unit performs logic control upon the received interlock signals. It is responsible for receiving the interlocks signals and actuate on the VVS, closing the valve and thus isolating a vacuum sector from the upstream and downstream ones.

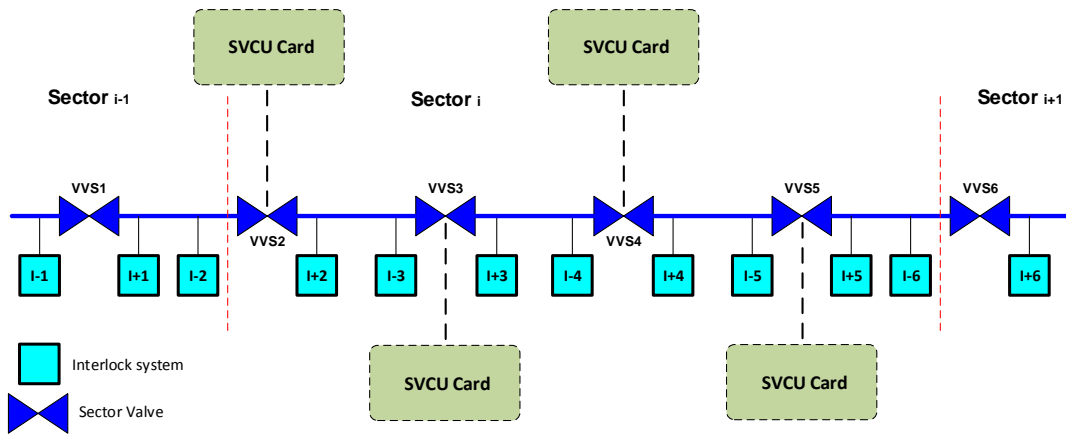


Figure 2.1: Vacuum sector of LHC.

## 2.2 Interlocks

In engineering field, an interlock system is used to prevent an activity of being initiated unless preceded by certain conditions. In every electronic or mechanical system, interlocks are used to protect the different sub-systems and operators from harmful actions by stopping the machine. For example, a beam dump in LHC is requested and triggered by an interlock sub-system in case of failure. When the vacuum sub-systems are operating correctly, the beam interlock system receives a status flags indicating that the vacuum is at the good level, and beam can be injected into LHC.

The block diagram of the used interlocks in the LHC machine can be seen in Figure 2.2. As mentioned before, each VVS is controlled by one SVCU card and there are two interlock system associated to each SVCU. Each SVCU card receives four signals (I-1, I-2, I-3 and I-4) from the interlock crate, which decodes the vacuum level given by the vacuum gauges around the controlled sector valve, two pressure interlocks from the previous and next SVCU cards and one signal given by the Beam Interlock System (BIS), which indicates if all the conditions are good for the beam to circulate (beam permit). The SVCU sends a pressure interlock to the previous and next SVCU, a beam dump is requested in case of a local pressure interlock. When a beam dump is required, the corresponding sector valve and also the adjacent sector valves are closed.

In the SPS accelerator, like in LHC, pressure interlocks from vacuum gauges are used. Local interlock signals (I-1 and I-2), from left and right card, are generated by ion-pumps. Additionally, global interlock signals are added to close all the valves in one sector (Figure 2.1). Global interlocks are triggered either there is local interlocks in the current SVCU or when a SVCU receives global interlock signals from other SVCU (left or right). Delta pressure interlock signals are also used coming from TPG300 controllers (VGR and VGP gauges of Figure 1.2). The diagram of the used interlocks for SPS is shown in Figure 2.3.

The complex PS machine uses only local pressure interlock signals from ion-pumps, as depicted in Figure 2.4. These interlocks might come from the left or right card.

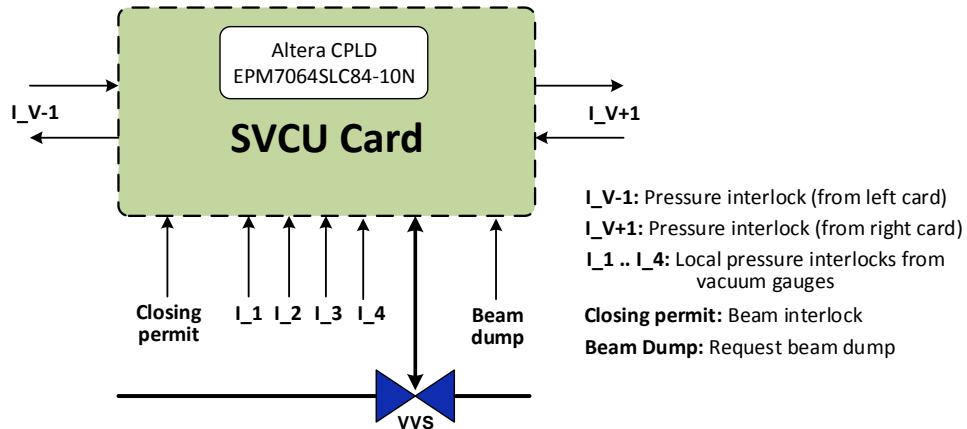


Figure 2.2: Interlock system of LHC.

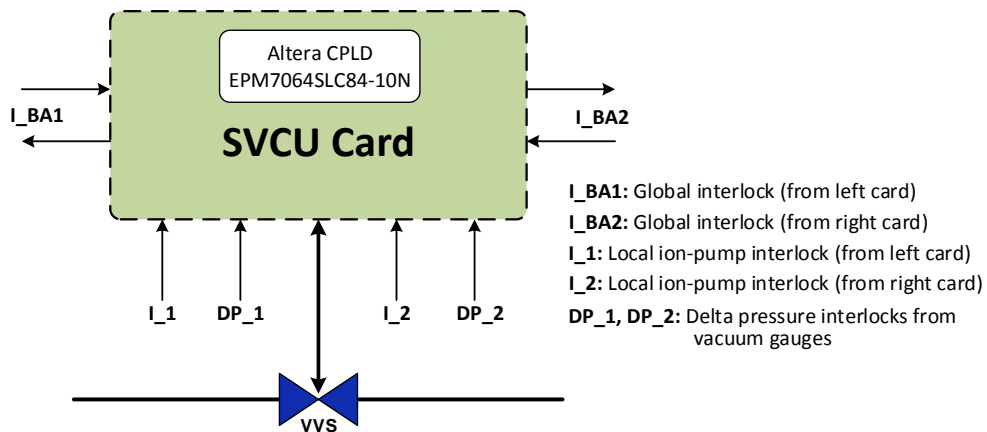


Figure 2.3: Interlock system of SPS.

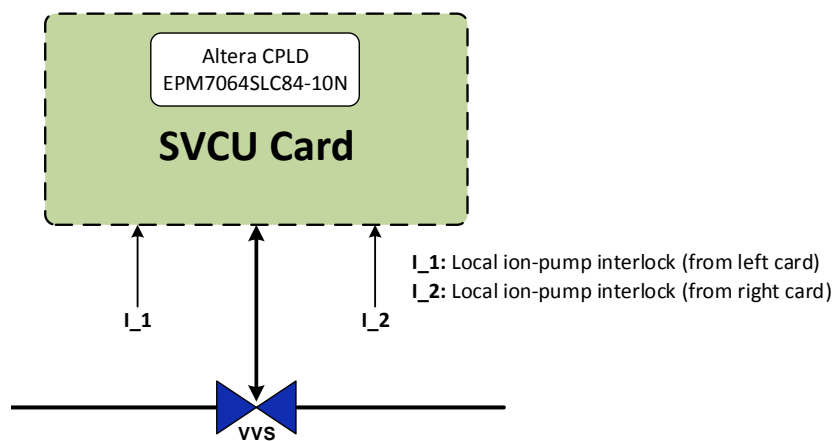


Figure 2.4: Interlock system of SPS.

## 2.3 Control System Architecture

The control system of the vacuum sectors is depicted in Figure 2.5. The system comprises four main modules: Programmable logic controller (PLC) master, SVCU card, Valve Control Unit (VCU) and VVS.

A PLC master communication allows the remote control of the SVCU card, which actuates on the VVS through the control rooms. A multiplexer module addresses up to 32 SVCU cards per PLC bus. This bus contains up to 6 digital inputs (DI) and up to 8 digital outputs (DO). Each VVS is controlled by one SVCU card and one VCU. A Complex Programmable Logic Device (CPLD) is the logic controller of the SVCU card. It receives the interlock signals coming from the vacuum gauges and other sensors, actuating on the VCU, and thus on the VVS itself. The MUX card allows the data exchange between the PLC master and the SVCU cards, in which digital inputs are used to inform the vacuum users about the status information of each SVCU and digital outputs from the PLC master are used to address the desired SVCU and to send open/close commands to the VVS.

The VCU connects the SVCU to the VVS (interface driver) receiving open/close commands from SVCU, and actuating on the valve itself. Finally, the VVS is the sector vacuum valve that closes in case of bad vacuum level.

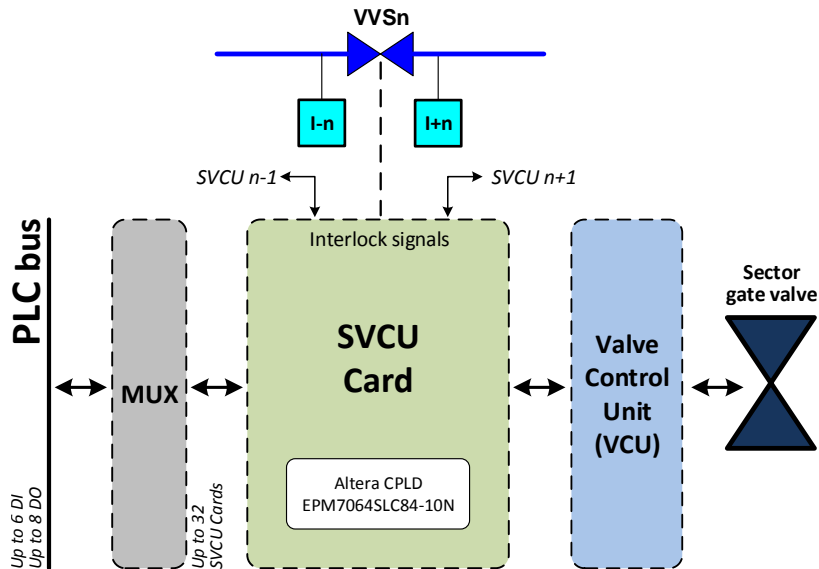


Figure 2.5: SVCUs general architecture.

The SVCU cards are remotely monitored and controlled by a PLC master. For the communication, the bus has several digital IOs. The PLC bus can address up to 4 chassis, where which one contains 8 SVCUs. In total, one PLC bus is able to access up to 32 SVCUs

cards. In Figure 2.6, the architecture of the PLC communication is depicted.

To access the desired SVCU card from the control room, 6 digital outputs are provided in one PLC bus and connected to a multiplexer (MUX). Using the first 2 lines ( $2^2 = 4$ ) and the next 4 lines ( $2^4 = 8$ ), the multiplexer decodes the chassis and the corresponding SVCU, respectively. Once the SVCU card is addressed, the communication starts with 6 digital outputs from the PLC and 8 digital inputs from the SVCU card. One PLC interrupt input is added to the system for interrupt handling. Besides the remote communication, SVCU cards can also be controlled locally through buttons in the SVCU front panel. The SVCU card is going to be explained further in this chapter.

The architecture of the PLC-SVCUs communication system is the same for all the accelerators here documented (LHC, SPS and CPS). Although, the digital IO line function of the PLC bus might change from one accelerator to another. For further explanation and precise information about each digital IO, please refer to [8].

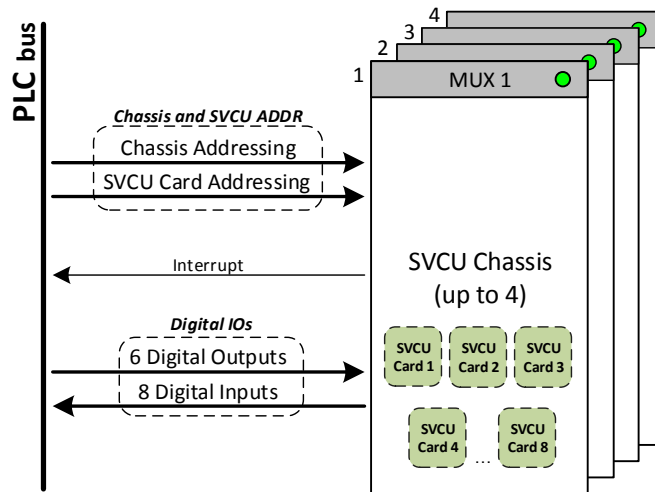


Figure 2.6: PLC communication and MUX.

## 2.4 SVCU Card

The SVCU electronic card is the logic control device for the vacuum sectors of LHC, SPS and CPS. The SVCU card comprises three main hardware modules: front panel, electronic circuits and the backplane connector, as can be seen in Figure 2.7. In the front panel of the SVCU card there is a button that controls the valve state (open/close). The VVS can only be controlled locally if the local control mode is enabled. Status LEDs (Light-Emitting Diode) are also in the front panel, informing the users about its status.

Each SVCU card is controlled by one CPLD. The EPM7064SLC84-10N from Altera [9], belongs to the MAX7000S series and contains 64 macrocells, 68 IO ports and 84 pins,

with 5 V core operating voltage, 3.3 V or 5 V as input voltages and 3.3 V or 5 V as output voltages. Regarding the control logic, it is completely combinational, meaning that no clock is used. The output state is a digital function build by logic gates, which depends on the present value of the inputs (interlocks and open/close commands). The control logic is going to be explained in the following section.

The backplane connector contains all the interlock signals, power supplies and digital outputs to the valve control unit (VCU). A IDC-32 connector allows the interface between the CPLD logic controller with the interlocks systems and PLC master. The SVCU cards are plugged into chassis and addressed by the PLC master through a multiplexer. Therefore, every SVCU card is accessible from the control room, where the vacuum users are able to check its status and close or open the corresponding VVS.

To control the vacuum sectors of LHC, SPS and CPS, one single PCB (Printed Circuit Board) was developed, the SVCU card. However, the interlock system and thus, the control logic of each machine are different. In order to be able to use the same electronic card, dip-switches were added to control its behaviour in function of in which machine it is being used. Since the logic controller is a CPLD, the firmware inside can easily be changed and re-programmed. The on-board LEDs have different functions and the metallic front panel of each machine is different from one to another [8].

The front panel of the SVCU card for LHC is depicted in Figure 2.8. There are several LEDs from the interlock signals informing the user about their state. A 5 mm LED shows whether the valve is opened or closed. As stated before, the SVCU can be controlled remotely (PLC master) or locally. To enable the on-site control, a hidden push-button must be pressed and the LOC LED turns on.

The front panel of the SVCU card used in SPS is depicted in Figure 2.9. It is similar to the LHC one, having though some differences in the interlock signals LEDs. Global interlock signals were added to this card and are generated by means of local interlocks. If a local interlock is received from left or right card, two global interlocks are triggered and sent to all valves in the current sector closing them.

In the CPS machine only two interlock signals are used, left and right local interlocks. To open and close the valve locally, two buttons are available. The local control mode is enabled by pressing the hidden button. The front panel, as well as its description, can be seen in Figure 2.10.

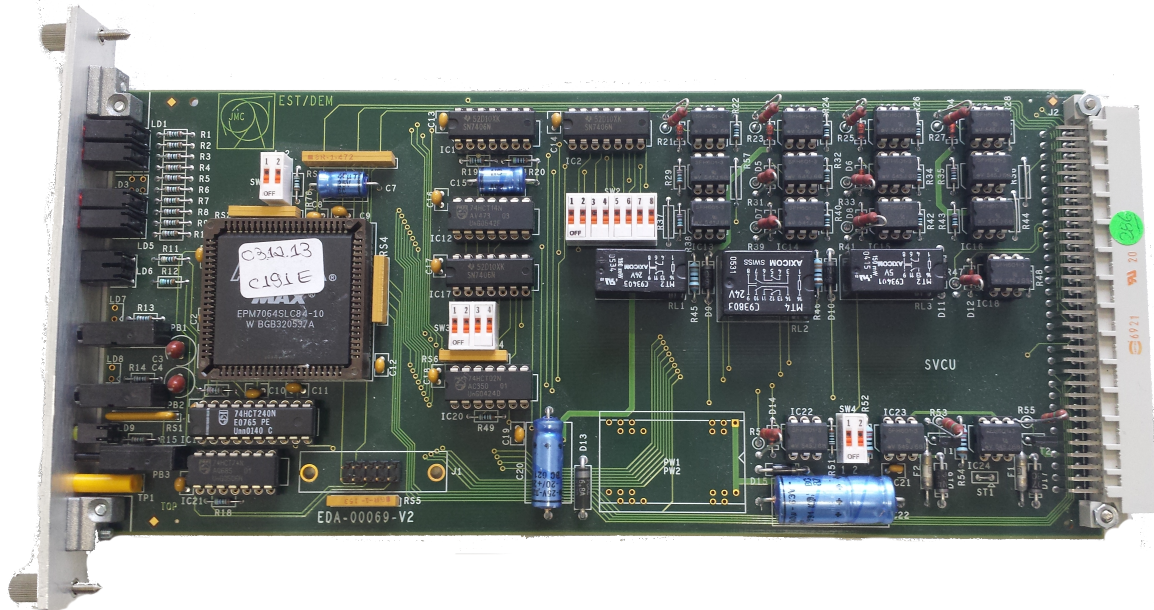


Figure 2.7: SVCU card.

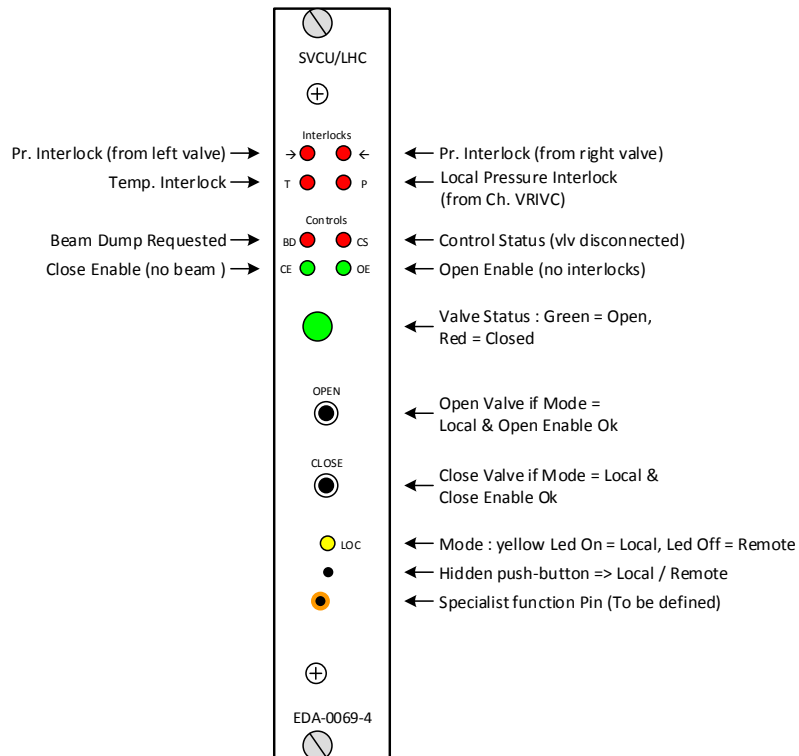


Figure 2.8: SVCU card front panel of LHC.

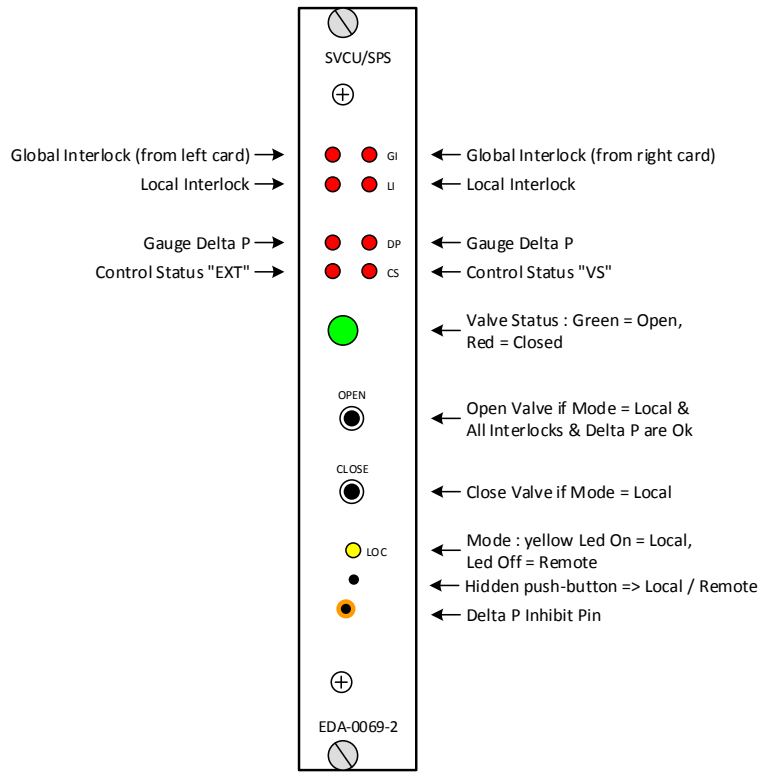


Figure 2.9: SVCU card front panel of SPS.

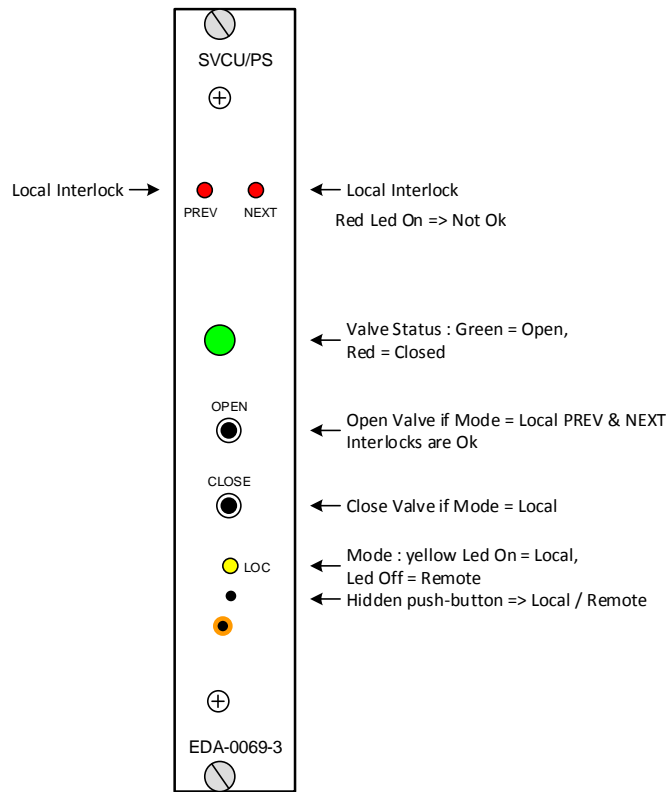


Figure 2.10: SVCU card front panel of CPS.

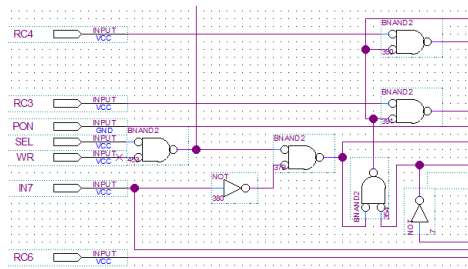
## 2.5 Control Logic

The on-board CPLD controls the SVCU card, receiving the interlock signals and actuating, locally or remotely the VVS. The control logic, as stated before, is pure combinational logic, thus the actuation is a function of the actual inputs affected to each machine. Although the same SVCU card is used in all the vacuum sectors of LHC, SPS and CPS, the interlock system is different and, therefore the control logic inside the CPLD is also different.

The control logic for the SVCUs was developed few years ago, using only logic gates (e.g. AND, NOR or XOR) building an output function, which is a combinational result of the inputs (Figure 2.11(a)). At the time of the control logic development and, since the vacuum sectors of SPS and CPS machines are similar, the developed firmware for the SVCU card used in SPS was being used also in the SVCU cards for CPS. However, in the SPS machine there are more interlock signals than in the CPS one, which leads to unconnected pins inside the CPLD.

In order to make the control logic more robust and easier of understanding, VHDL code was developed (Figure 2.11(b)) and programmed into the SVCU cards. The aim of this firmware development is to assure the portability from a schematic description designed with an old version of Quartus (v6.0) running only on windows XP into a VHDL description with the last version of Quartus running on windows 7 while keeping the same functionalities. This is the first step for further improvements of the SVCU cards. The schematic based control logic was literally translated into VHDL language. Buffers of the outputs were used in order to be able to use them as internal inputs. The whole commented VHDL code for each machine can be seen in [8].

An upgrade was done in the CPS machine, deleting the logic that was not being used and keeping only the needed logic for the present interlock signals. Two SVCU cards for the CPS machine were programmed and tested on-site near the accelerators.



(a)

```
-- valve control outputs (table 7 of the current report) --
OP_buf <= (((((NOT((NOT((NOT(BOP OR REM12_buf)) OR NOT(RC2 OR ((SEL OR
CL_buf <= (((NOT((NOT(SW3_3 AND VS)) AND NOT((IN3 OR IN6) AND NOT(NIF
--Debug LEDs--
LED1 <= NOT(IN8);      --'0' > LED1_1 ON (Global Interlock from next VVS)
LED2_buf <= NOT(IN1);  --'0' > LED1_2 ON (Global Interlock from VVS-1)
LED3 <= NOT(IN6);     --'0' > LED2_3 ON (Local Interlock from VVS-1)
LED4 <= NOT(IN3);     --'0' > LED2_4 ON (Local Interlock from VVS+1)
--LED5 (not used)
--LED6 (not used)
LED7 <= NOT(IN5);     --'0' > LED4_7 ON (Dp from VVS+1) not OK
LED8 <= NOT(IN4);     --'0' > LED4_8 ON (Dp from VVS-1) not OK
LED9_buf <= NOT(VS);  --'0' > LED5_9 ON (Control status VS) not OK
LED10_buf <= NOT(EXT); --'0' > LED5_10 ON (Control status EXT) not OK
```

(b)

Figure 2.11: Control logic of SVCU cards (a) logic gates; (b) VHDL.

In the following items, the logic that is being used in LHC machine is presented.

1. The valve can be opened, either locally or remotely, if the LED "OE" in the front panel (Figure 2.8) is on;
2. The valve can be closed, either locally or remotely, if the LED "CE" is on;
3. Local pressure interlocks close the VVS and send an interlock to the previous and next SVCU card;
4. An interlock from VVS-1 or VVS+1 close the current valve;
5. The beam dump is requested automatically whenever the command to close the valve is triggered;
6. If there is beam circulating, the valve can not be closed. First the beam must be dumped and only after the valve can be closed.
7. Local temperature interlocks do not cause the valve to be closed, but once the valve is closed, it can not be opened in presence of temperature interlocks;
8. To open the valve, all interlocks must be ok;
9. There is one key, on-site, which enables/disables the remote control communication.

The following items describe the behavior of the SVCU card used in SPS.

1. Global interlocks from previous or next cards, close the current valve;
2. Local interlocks from previous or next cards, close the current valve and are generated global interlocks to all left and right cards;
3. Delta pressure interlocks do not close the valve. However if the valve is closed and if there is one delta pressure interlock not ok, the valve can not be opened;
4. To open the valve, all interlocks (local and delta pressure interlocks) must be ok;
5. There is one key, on-site, which enables/disables the remote control communication.

The following items describe the behavior of the SVCU card used in CPS.

1. Local interlocks from both sides, left or right, close the valve;
2. To open the valve, local interlocks must be ok;
3. There is one key, on-site, which enables/disables the remote control communication.

## 2.6 Modelsim Simulations

To test the new control logic, Modelsim simulations were conducted. A VHDL testbench file was created for each machine and tests were performed in order to validate the control behavior stated above. Knowing in advance how each SVCU responds to the inputs, a VHDL test bench file was written simulating certain conditions (e.g. interlocks not ok or open button pressed). The Modelsim simulation for SPS is depicted in Figure 2.12.

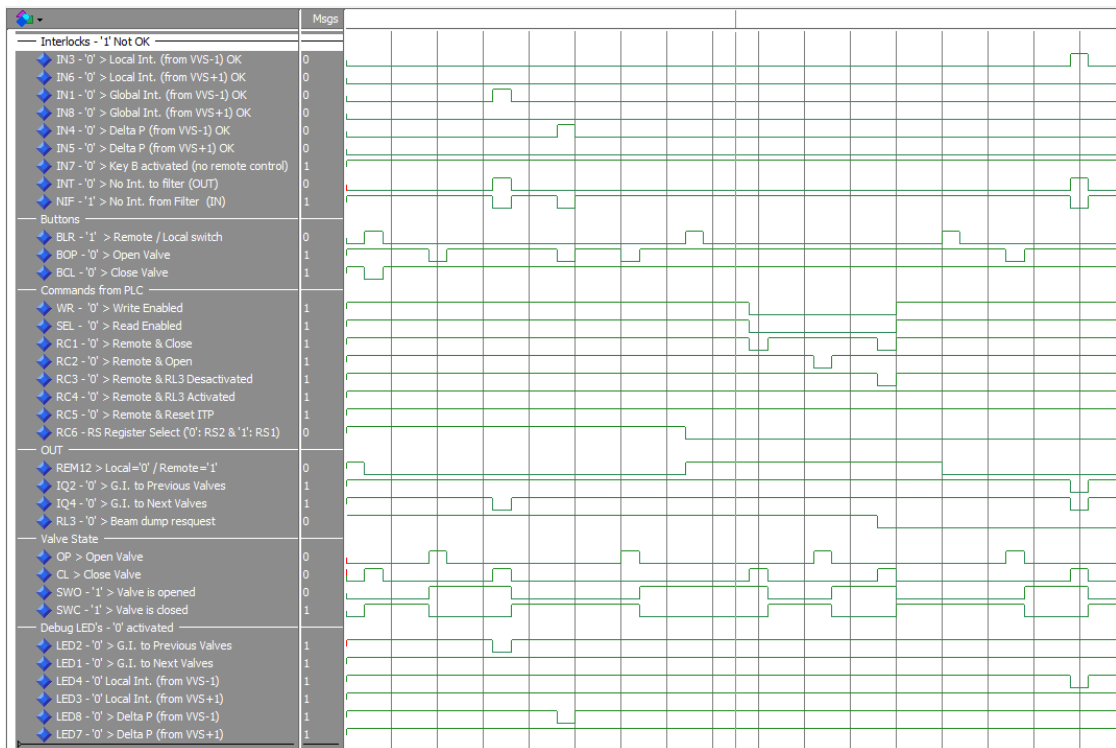


Figure 2.12: SVCU card front panel of CPS.

In total, twelve tests were performed for the SPS control firmware. In each test, digital values were assumed simulating malfunctions (presence of interlocks) or open valve orders (open button pressed). The new controller was also tested in laboratory where there is a specific test bench for each machine.



# Chapter 3

## Sputter Ion Pump Controller Architecture

In this chapter, the present and the proposed architecture of the sputter ion pump controllers are presented. The existent control system, the VPI controller modules and its signals are explained. Finally, the advantages of the proposed architecture and the new signals are described.

### 3.1 Present VPI Controller Architecture

Among the CERN's accelerators, several systems (e.g. magnets, accelerating cavities, cryogenic and vacuum) are needed to work together to put the proton or heavy ions beams circulating. The Vacuum system contributes to minimize the interaction between the accelerating beams and the residual gases, thus maximizing the beam lifetime. When needed to reach ultra-high vacuum, sputter ion pumps are used [ $10^{-5}$ .. $10^{-11}$  mbar] [2].

In Figure 3.1 the VPI architecture of the LHC is depicted. It is comprised by three main parts: the Remote IO station, the VPI controller and the pump itself. As can be seen, a Remote IO station can connect up to 40 VPI controller to the network. The Remote IO station is composed of several analog or digital IO modules and a Profibus slave module (ET200). Each module can be used as input, output or mixed and are connected in one hand to the ET200 internal bus, and in the other hand to the back panel for the VPI controller connections. The VPI controller, located in the underground service areas takes the control of the pumps and provides the high-voltage supply. Finally, the pump itself, situated in the accelerator tunnel, is mounted directly on the beam pipe.

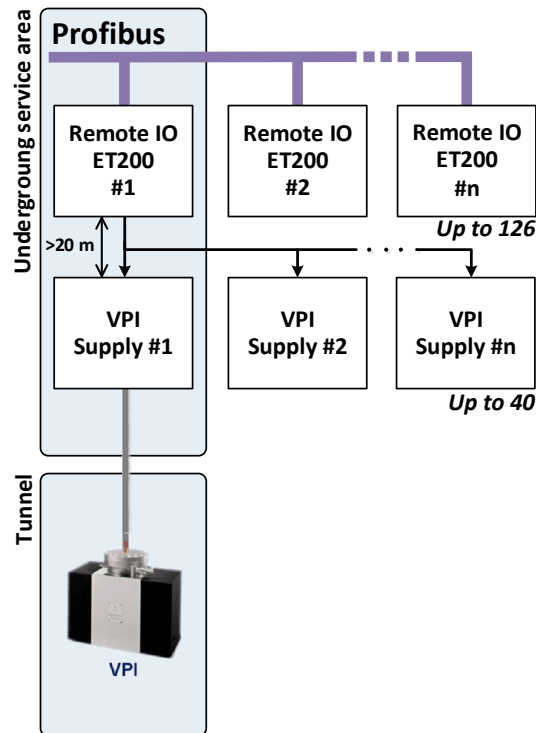


Figure 3.1: VPI architecture of the LHC.

### 3.1.1 VPI Controller Modules

The VPI controller is formed by four different modules: the back panel, the control cards, the high-voltage actuation and the front panel. The back panel is where all the control signals are connected and the cards plugged. The control card implements the conditioning electronics of the ionization current (refer to section 3.1.4), the control of the high-voltage relays and the threshold level circuits. It informs the VPI status either directly on the front panel, or on the back panel for remote control.

The high-voltage module contains the supply and switches to turn on and off the 6 kV applied to the pump. It includes also the high-voltage connector and cable connected to the pump located in the tunnel. The front panel contains the status LEDs, as well as the potentiometers to set the threshold levels.

### 3.1.2 Present Control Architecture

The industrial Profibus network is widely used at CERN to control the field equipments, as well as the existing sub-systems. The present VPI controller is controlled through remote IO station also through a hand-unit device for local control, as shown in Figure 3.2.

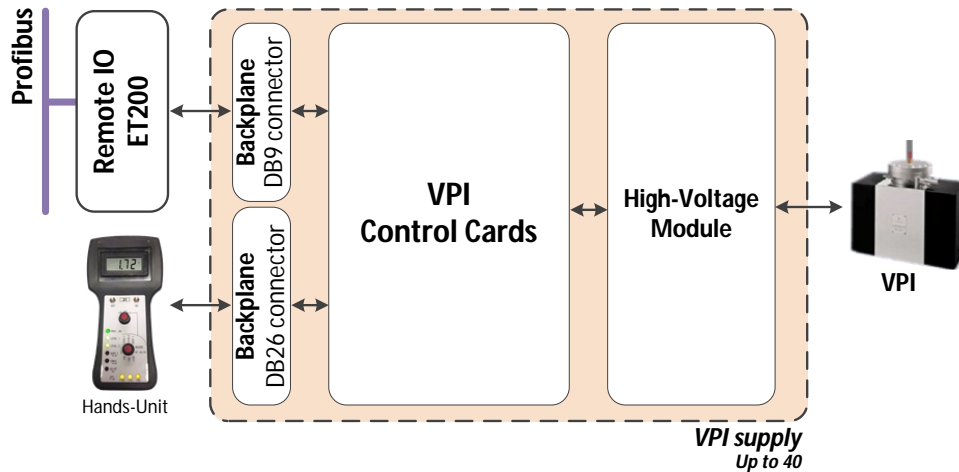


Figure 3.2: VPI control architecture.

The Remote IO station is the control interface between the Profibus master and the VPI controller. A DB-9 connector in the back panel of the controller, allows the connection from the remote IO station where the signals are sent or received. A high-voltage isolated block contains the switches and relays to turn on and off the 6 kV for the pump. The hand-unit is used to control the VPI locally and also to check its status information, the programmed threshold levels and the ionization current measurement. According to the standards, the Profibus network supports up to 127 slaves connected to the same bus, therefore up to 127 Remote IO station can be connected, and up to 40 VPI controllers can be multiplexed using the same remote IO station.

### 3.1.3 Present Control Signals

The control of the VPI controller is carried out by the Profibus master through the remote IO station. Depending on the ionization current, the threshold levels and the VPI status information, the controller takes the decision whether to turn on or off the high-voltage supply for the pump. To establish this bidirectional communication, three signals are used, as can be seen in Figure 3.3.

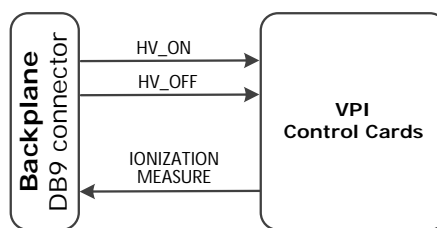


Figure 3.3: Signals of the current VPI control architecture.

To turn on and off the 6 kV supply, "HV\_ON" and "HV\_OFF" are used. These digital (24 V) outputs coming from the Remote IO station are responsible to get the pump powered up with high-voltage, and thus reach ultra-high vacuum. As feedback control, a 0-10 V analogue signal is returned. One part corresponds to the consumption current, being a good measurement of the pressure. Another part corresponds the VPI status. The resulting signal, is transmitted to the remote IO station and then to the Profibus master.

### 3.1.4 Ionization Current Measurement and Status

The ionization current and the status information of the VPI are both merged in the same signal. The graphical representation of this signal is presented in Figure 3.4. The x-axis represents the ionization current, comprised between 1  $\mu\text{A}$  to 100 mA (desired range), and the y-axis shows the result voltage, according to the ionization current and the status information.

From 0V to 6V is represented the VPI ionization current with a logarithm scale, and from 6V to 10V this signal traduces 6 status of the VPI.

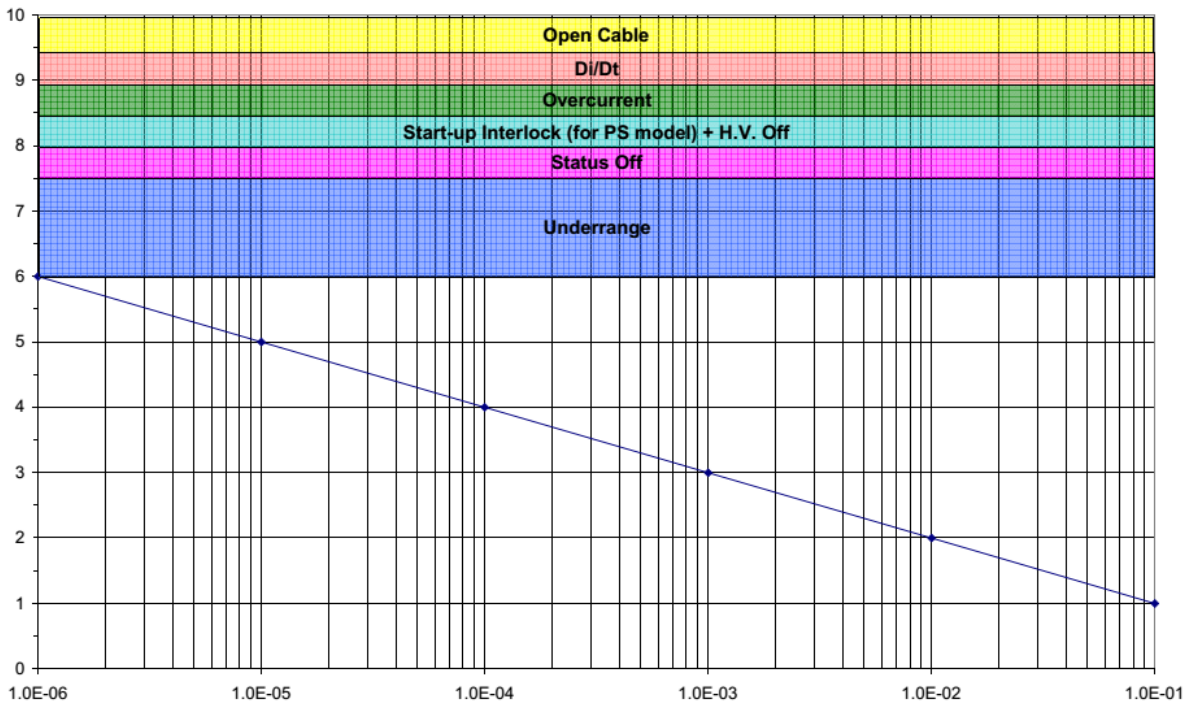


Figure 3.4: 0-10 V feedback control signal [2].

## 3.2 Proposed VPI Supply Architecture

In this project, an enhanced control architecture is proposed, according to Figure 3.5. As can be seen, when compared to the existing control architecture (Figure 3.1), the Remote IO station has been removed. This component, as mentioned before, is the control interface between the Profibus master and the VPI controller. However, using this kind of remote IO station results in some disadvantages; 1) Instead of using a unique Profibus cable, each equipment connected to the remote IO station has a different and proprietary cable; 2) The equipment controlled through this remote IO station is not an independent Profibus slave with a specific address in the network.

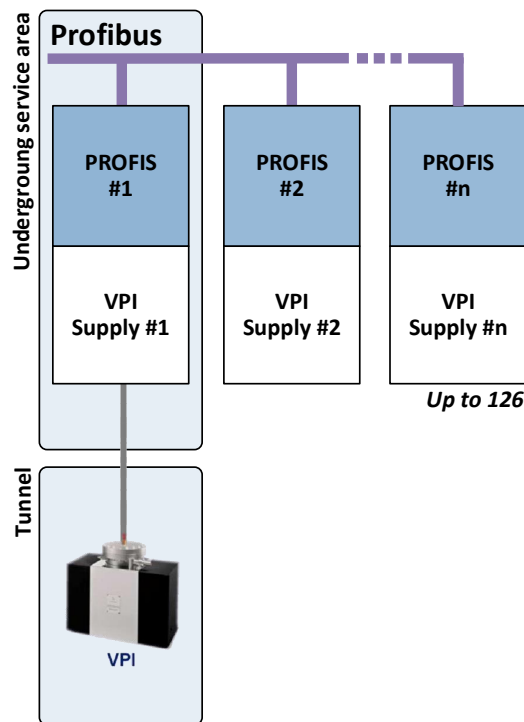


Figure 3.5: New VPI control architecture.

The usage of an integrated Profibus Intelligent Slave (PROFIS) in each VPI controller is the proposed solution. An Independent Profibus slave dedicated for VPIs was designed making this vacuum pump a common Profibus slave in the network and not an equipment connected to a remote IO station. This feature presents some advantages: 1) Each VPI controller is no longer interfaced with the remote IO station, and thus only the standard Profibus cable is needed; 2) Each VPI controller bears a unique station address, which is the Profibus network identifier; 3) The number of VPI signals that are currently accessed is significantly improved; 4) The programmed threshold levels are now available in the Profibus master.

### 3.2.1 Proposed Control System

The new control system is depicted in Figure 3.6. The interface between the Profibus master and the new VPI controller is established using the new designed board, PROFIS, which is going to be explained in the following sub-sections. As can be noticed, instead of having the remote IO station as control interface, the new architecture has a specific intelligent slave for each VPI controller. The new controller is attached on the VPI controller requiring only a standard Profibus cable to establish the communication with the master.

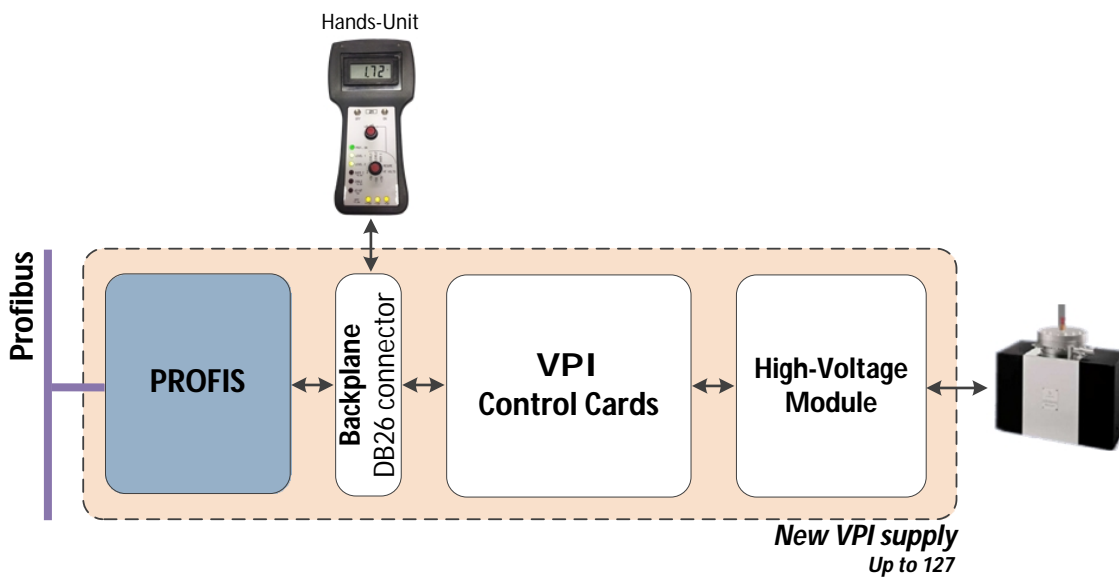


Figure 3.6: Proposed VPI control architecture.

### 3.2.2 Hand-Unit

The control of the VPI controller is carried out by the Profibus master, but it can also be controlled through the hand-unit, depicted in Figure 3.6. This local control device is plugged in the back panel of the VPI controller through a DB26 connector, and is used to check the programmed threshold levels, the status information and to control the high-voltage power supply in the pump. The new PROFIS is going to be connected in parallel with the hand-unit, and thus has access to all its signals presented in Table 3.1).

The connection of the new VPI controller is depicted in Figure 3.7. The VPI control card splits the signals over the PROFIS, the DB26 back panel connector and the hand-unit. In the PROFIS there is an IO module where all the signals in Table 3.1 are acquired and conditioned, in order to send them to the microcontroller with the good voltage level for digital or analog acquisitions.

Table 3.1: Hands-unit signals.

Hands-unit IDC-26	I/O	Signal	Range	Description
1	O	HV_ON	0V;12V (ON-OFF)	Turn-on the high-voltage
2	O	HV_OFF	0V;12V (ON-OFF)	Turn-off the high-voltage
3	I	OVER I	0V;5V (ON-OFF)	Over current warning
4	I	TIME OUT	0V;5V (ON-OFF)	Time out warning
5	I	CABLE	0V;5V (ON-OFF)	Cable indication
6	I	PROTECTION	0V;5V (ON-OFF)	Protection indication
7	I	LEVEL 1A	Relay contact	Level 1 threshold
8	I	LEVEL 1A'		
9	I	LEVEL 1B	Not used	
10	I	LEVEL 1B'		
7	I	LEVEL 2A	Relay contact	Level 2 threshold
8	I	LEVEL 2A'		
13	I	LEVEL 2 THRESHOLD	[0-5V]	Front panel potentiometer
14	I	LEVEL 1 THRESHOLD	[0-5V]	Front panel potentiometer
15	I	HV READING	[0-5V]	High-voltage measure
16	I	OVER CURRENT THRESHOLD	[0-5V]	Front panel potentiometer
17	I	CURRENT INPUT		Not used
18	I	CURRENT COM		Not used
19	I	GND		
20	I	IONIZATION MEASURE	[0-10V]	Ionization measure (graph of figure 3.4)
21	I	GND		
22	I	+5V		
23	I	+15V		
24	I	-15V		
25	I	GND		
26			Not used	

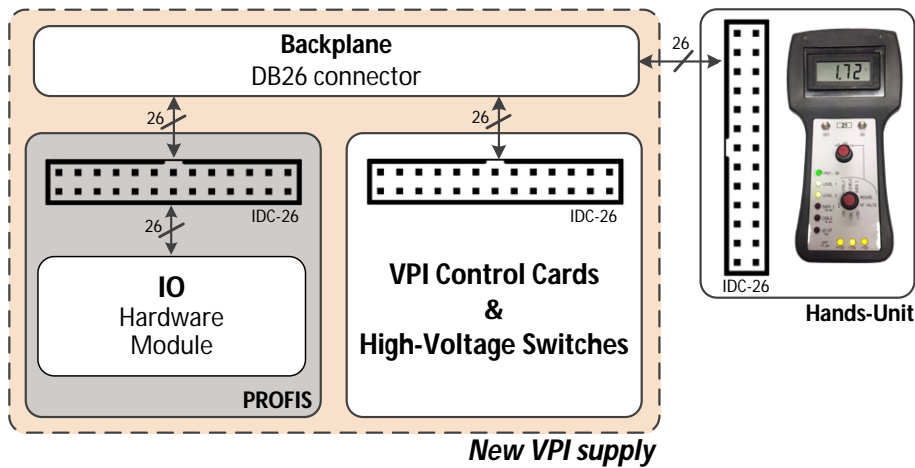


Figure 3.7: PROFIS IO connection.

Among the hand-unit signals, almost all of them are used in the new PROFIS, but first they have to be conditioned to fit the microcontroller’s IO specifications. In the next sections, the used signals are described.

As mentioned before, with this architecture and introducing the PROFIS in the control system of the VPI controller, the Profibus master is going to have more access into the VPI control signals. The programmed threshold levels (level 1, level 2 and over current) are now available in the remote control system, as well as the status information. This upgrade makes the troubleshooting of each VPI easier, because it is no longer necessary going to the tunnel to check the threshold levels, essential part of the VPI’s behavior.

### 3.2.3 PROFIS Signals

The new PROFIS is the proposed control interface for the VPI controller, replacing the remote IO station in the current architecture. The number of accessed signals increases, and the control communication with the pump is therefore enhanced. As stated in the previous section, the PROFIS signals are acquired through an IDC-26 connector and conditioned by an on-board hardware module. The used signals, among the hand-unit signals (Table 3.1), are presented in Table 3.2.

Figure 3.8 depicts the signals diagram of the new VPI controller control architecture. Inputs are signals coming from the VPI controller (Hand-Unit DB26 connector) and outputs are the signals coming from the PROFIS or from the Profibus master. As can be seen, before passing through the PROFIS IO module, 13 input signals are acquired. As outputs, 2 signals control the high-voltage supply to the pump.

Table 3.2: Hands-unit used signals in PROFIS.

<b>Hands-unit IDC-26</b>	<b>Signal</b>	<b>Description</b>
1	HV_ON	Turn on the high-voltage supply (0;12V digital signal).
2	HV_OFF	Turn off the high-voltage supply (0;12V digital signal).
3	OVER I	ON if the ionization current of the pump has reached the programmed threshold level. (digital signal which is connected to a front panel LED)
4	TIME OUT	ON if short circuit has occurred (measured after 30 minutes). (digital signal which is connected to a front panel LED)
5	CABLE	ON if the high-voltage cable is disconnected in the VPI controller back panel. (digital signal which is connected to a front panel LED)
6	PROTECTION	On if the VPI ionization current is between the admitted values. (digital signal which is connected to a front panel LED)
7	LEVEL 1A	Level 1 relay contact. This contact closes if the ionization current
8	LEVEL 1A'	has reached the programmed threshold value for level 1.
11	LEVEL 2A	Level 2 relay contact. This contact closes if the ionization current
12	LEVEL 2A'	has reached the programmed threshold value for level 2.
13	LEVEL 2 THRESHOLD	Reading of the programmed threshold for level 2. (Analog voltage from front panel potentiometer [0V-5V])
14	LEVEL 1 THRESHOLD	Reading of the programmed threshold for level 2. (Analog voltage from front panel potentiometer [0V-5V])
15	HV READING	Reading of the high-voltage value comprised in [0V-5V] (1/10000) factor scale
16	OVER CURRENT THRESHOLD	Reading of the programmed threshold for over current.
20	IONIZATION MEASURE	Analog signal that traduces the ionization measure. (graph of Figure 3.4)
21	GND	PROFIS 0V
23	15V	PROFIS +15V power supply
25	GND	PROFIS 0V

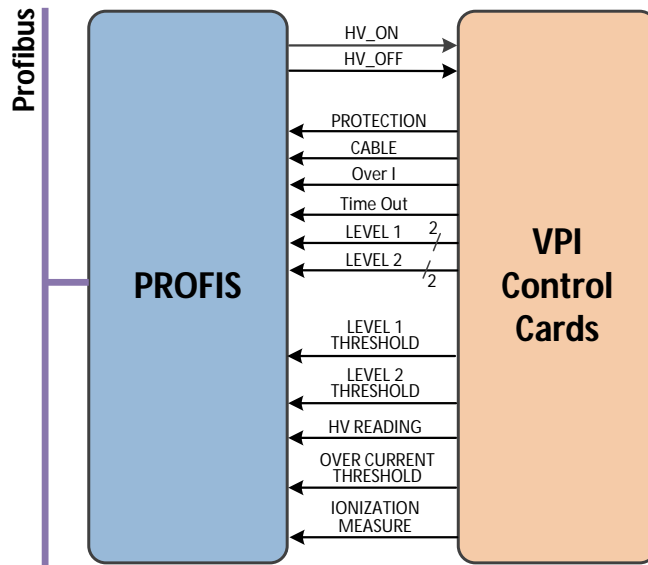


Figure 3.8: Signals of the proposed architecture.

### PROFIS to VPI Supply

Signals from the PROFIS to the VPI controller are considered as outputs and are shown in Table 3.3. The control of the high-voltage to the pump is performed through these 2 digital 12 V outputs, which are connected to the coil of the high-voltage relays.

Table 3.3: PROFIS to VPI controller signals.

Hands-Unit IDC-26	I/O	Signal	Description
1	O	HV_ON	Turn-on the high-voltage (0V;12V digital relay output)
2	O	HV_OFF	Turn-off the high-voltage (0V;12V digital relay output)

### VPI Supply to PROFIS

Signals from the VPI controller to the PROFIS, considered as inputs are described in Table 3.4. The VPI status information, the programmed threshold levels and the ionization measure are acquired through the PROFIS internal IO module.

The status information (OVER I, TIME OUT, CABLE, PROTECTION, LEVEL 1, LEVEL 2) are collected from LED pins, therefore they need to be instrumented and their range converted in order to use them in the controller. The programmed threshold levels (LEVEL 2 THRESHOLD, LEVEL 1 THRESHOLD and OVER CURRENT THRESH-

OLD) are analog voltage inputs ([0V-5V] range) that are set through potentiometers in the front panel. The HV READING is an analog voltage that traduces the high-voltage value with a factor scale of 1/1000. Finally, the IONIZATION MEASURE is the analog voltage containing the ionization current and the status information as shown in Figure 3.4.

Table 3.4: VPI controller to PROFIS signals.

<b>Hands-Unit IDC-26</b>	<b>I/O</b>	<b>Signal</b>	<b>Description</b>
3	I	OVER I	ON if the ionization current of the pump has reached the programmed threshold level.
4	I	TIME OUT	ON if short circuit has occurred (measured after 30 minutes).
5	I	CABLE	ON if the high-voltage cable is disconnected in the VPI controller back panel.
6	I	PROTECTION	On if the VPI ionization current is between the admitted values.
7	I	LEVEL 1	Level 1 relay contact. This contact closes if the ionization current has reached the programmed threshold value for level 1.
8			
11	I	LEVEL 2	Level 2 relay contact. This contact closes if the ionization current has reached the programmed threshold value for level 2.
12			
13	I	LEVEL 2 THRESHOLD	Reading of the programmed threshold for level 2.
14	I	LEVEL 1 THRESHOLD	Reading of the programmed threshold for level 1.
15	I	HV READING	Reading of the high-voltage value. 1/10000 factor scale.
16	I	OVER CURRENT THRESHOLD	Reading of the programmed threshold for over current.
20	I	IONIZATION MEASURE	Analog signal that traduces the ionization measure (graph of Figure 3.4).



# Chapter 4

## Board Design and Development

This chapter presents the hardware design and development of an intelligent Profibus slave specific for sputter ion pumps. The board architecture is first presented and a description of the hardware modules is given. The circuits and the used schematics are depicted and explained one by one. Finally, a prototype of the designed PCB is shown.

### 4.1 Board Architecture

The new PROFIS board can be divided into four main hardware modules: Profibus interface, controller (Profibus ASIC + microcontroller), input-output module and power-supply, as depicted in Figure 4.1. The Profibus interface module performs the signal conditioning between the Profibus industrial network and the Profibus ASIC. It is responsible for the reception and transmission of the Profibus telegrams and perform a galvanic isolation between the differential input Profibus network lines and the electronic signals of the PROFIS.

The controller comprises two different modules working together: the Profibus ASIC and the microcontroller. The Application-Specific Integrated Circuit (ASIC) IC handles the physical layer (layer 1) and the data link layer (layer 2) of the Profibus-DP network according to the OSI reference model [4]. The ASIC integrated Random Access Memory (RAM) is used as an interface between the ASIC and the microcontroller module. The control of the serial communications, the signals acquisition, the actuation of the high-voltage and the Profibus state machine is carried out by the microcontroller.

The interface of the PROFIS with the VPI supply is accomplished by the internal IO module. This module performs the needed instrumentation over the desired signals in order to adjust their range (analog and digital) to be compatible with the microcontroller.

Finally, in the power supply-module, several linear DC regulators provide all required

voltage levels to power the different hardware modules. All PROFIS board modules, will be explained in the following sections.

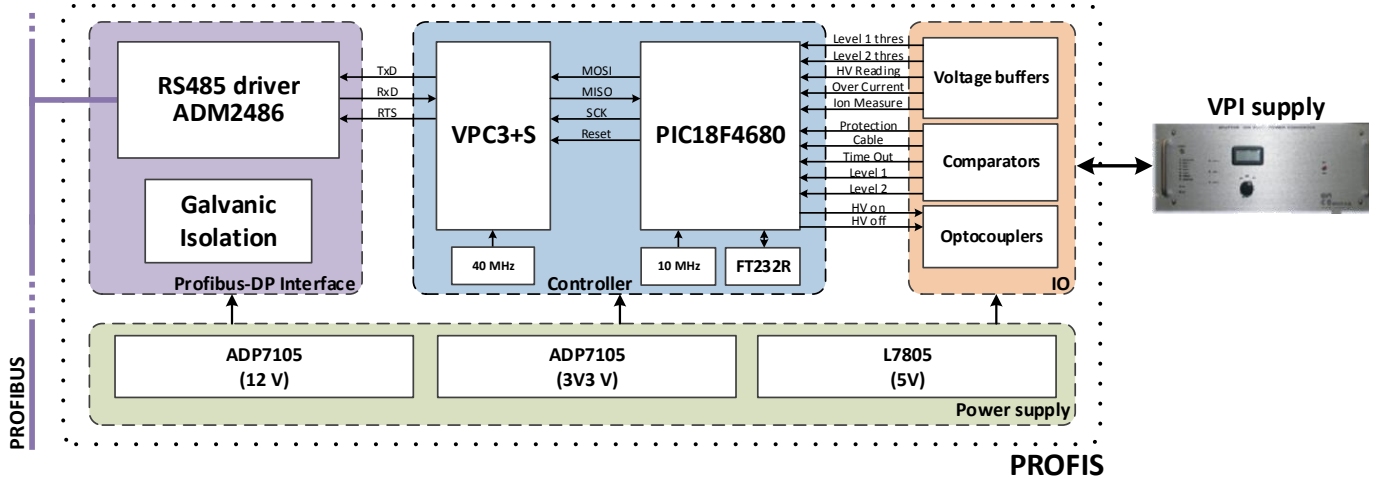


Figure 4.1: Block diagram of the new PROFIS board.

## 4.2 Profibus-DP Interface

The Profibus bus network uses the RS485 standard protocol as the physical layer for communication. The Profibus-DP interface module connects the differential bus network into the PROFIS board, which is galvanically isolated. The schematic of this module is represented in Figure 4.2.

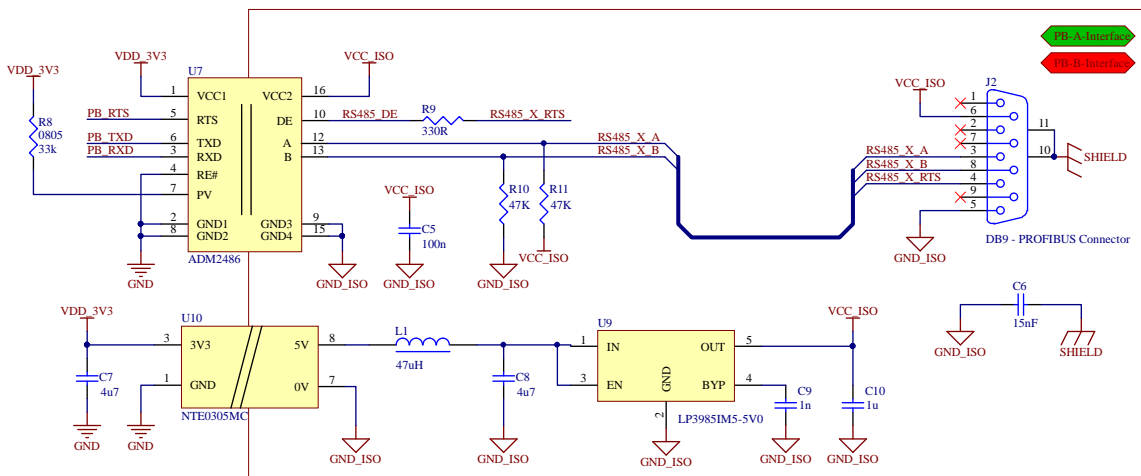


Figure 4.2: Schematic of the Profibus-DP interface module.

### 4.2.1 Profibus Transceiver

The RS485 transceiver was implemented with the ADM2486 circuit from Analog Devices. This circuit is an integrated differential bus transceiver with galvanic isolation, which was specifically designed for bidirectional data communication used in multi-point bus transmission lines [10]. A DB9 Profibus connector interfaces the bus network. It connects the A and B RS485 differential lines into the bus side of the transceiver, as shown in Figure 4.2.

The interface bus with the Profibus ASIC has three lines that are used for the communication, TX, RX and RTS (Request To Send). Assuming as reference the Profibus master, TX (pin 6 of U7) is used to transmit data and RX (pin 3 of U7) is used to receive data by PROFIS. The RTS signal (pin 5 of U7) is used by the slave to answer, informing the master that is ready to exchange data. In Figure 4.3 two oscilloscope screenshots of these three signal are depicted. From the top to the bottom, the first wave is TX, the second one is RTS and the last one is RX.

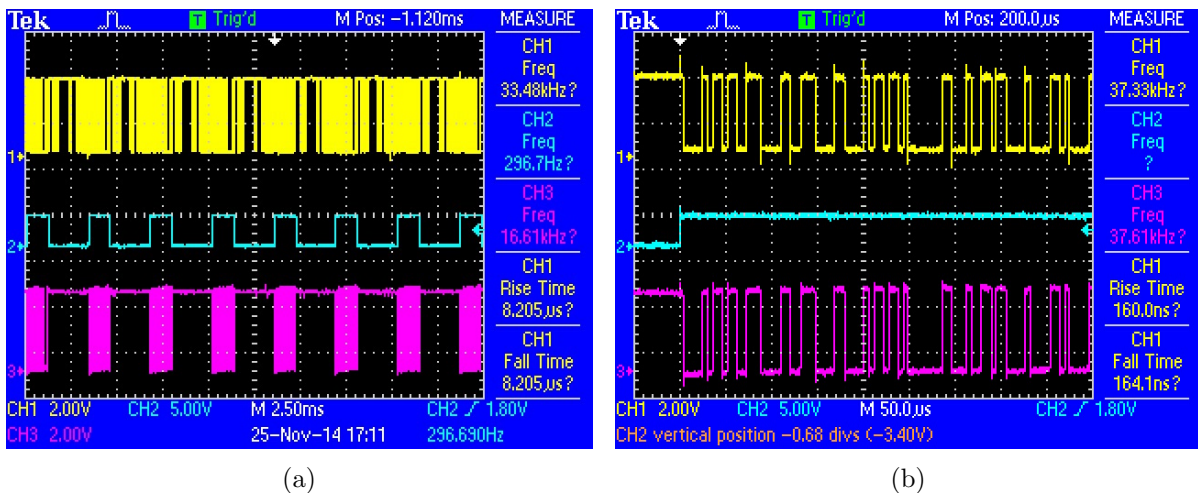


Figure 4.3: Oscilloscope screenshots of the Profibus communication (a) 2.55 ms/div and (b) 50  $\mu$ s/div.

As can be seen in Figure 4.3(a), the Profibus communications are cyclic and occur with a frequency of approximately 296 cycles per second (given by the RTS signal). First, the Profibus master request a data exchange with the slave; then, if the slave parametrization and configuration have been accomplished (refer to Chapter 5), the slave informs the master that is ready to send data and the communications starts.

### 4.2.2 Galvanic Isolation

To isolate galvanically the electronic internal modules of the PROFIS from the Profibus network, and simultaneously power the transceiver with the internal PROFIS power supply, it is necessary to use a DC-DC converter with galvanic isolation. For this purpose, the NTE0305 [11] (U10 of Figure 4.2) DC-DC converter is used. The NTE0305 from Murata is a 1 W, step-up DC-DC converter, galvanically isolated that provides an output voltage of approximately 5.6 V with 3.3 V as input voltage.

The IC output is connected to a low-pass filter formed by the inductance (L1) and the capacitor (C8). This filter is used to eliminate some high frequencies that might appear in the signal caused by external influences. The isolated 5 V power supply (VCC\_ISO/GND\_ISO) is fixed through a 150 mA ultra low-dropout voltage regulator LP3985 [12] (U9).

## 4.3 Controller - Profibus ASIC Module

The controller module of the PROFIS comprises two parts: the Profibus ASIC and the microcontroller, as can be seen in Figure 4.1. These two modules control the entire application, from the acquisition/actuation in the VPI supply to the Profibus state machine.

### 4.3.1 Profibus ASIC

In order to implement this module several programmable electronic devices can be used (e.g. FPGAs or microcontrollers). Alternatively, this module may be implemented using ASICs, which are devices customized for a particular use. In embedded systems, this type of IC is largely used to release the microcontroller to other tasks related with the application user demands, and to implement specific sub-modules with a high level of complexity, reducing the design time. A Profibus ASIC is therefore used in this application to handle the physical layer (layer 1) and the data link layer (layer 2). In order to select the most suitable ASIC for the current application, a market survey was performed.

### 4.3.2 Market Survey

In the market there are 2 main manufacturers of ASICs for Profibus-DP, Siemens and Profichip. These manufacturers present various ASICs for different functional needs and applications: master applications; intelligent slave applications and simple slave applications, as shown in Figure 4.4.

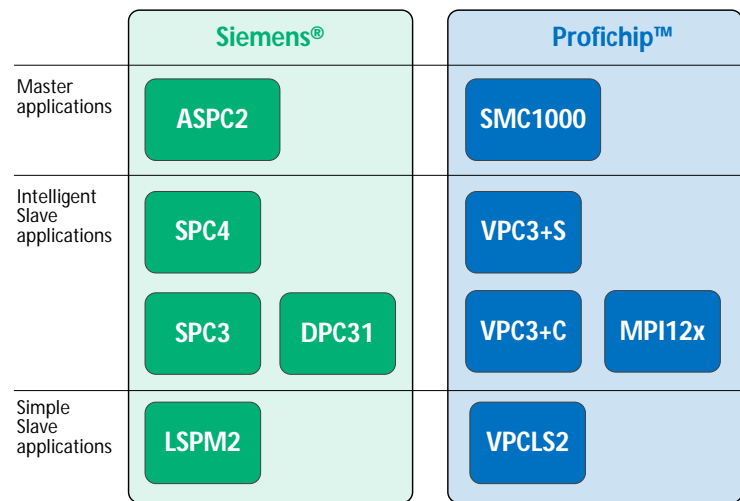


Figure 4.4: Profibus-DP ASIC market survey [13] [14].

The PROFIS board is intended to be a slave in the Profibus network, therefore only the intelligent and the simple slaves of Figure 4.5 are taken into account. The Siemens SPC4 is an intelligent Profibus-DP slave but also a Profibus-PA (Process Automation) ASIC. Profibus-PA is used to monitor measuring equipment via a process control system in automation applications and not in a centralized controller [4]. The Profichip MPI12x combines a MPI (Message Passing Interface) communication IC and a Profibus slave core VPC3+C. This type of ASIC is used for applications that require selective MPI or Profibus slave functionality sharing the same hardware. Thus for the current application the SPC4 and the MPI12x are excluded since it is only necessary an ASIC that implements the Profibus-DP protocol.

The considered ASICs for this application are the VPC3+S, SPC3, VPC3+S, DPC31, VPCLS2 and LSPM2, which can be grouped in to four types of solutions, as illustrated in Figure 4.5.

Solutions 1, 2 and 3 belong to the intelligent slave applications, since they have an internal or external microcontroller. On the other hand, solution 4 is only suitable for simple slave applications, where there is no need of data processing. In solution 1, the microcontroller and ASIC interface is achieved through a parallel data and address bus. The RAM of the Profichip VPC3+C and the Siemens SPC3 is addressed through 11 parallel bits and has 8 bits of data, which is also a parallel bus. This would lead to the use of 19 IO pins only for the ASIC interface. As a consequence, since 19 pins of the microcontroller are already assigned, an IC with a large number of IO pins would have to be used.

The second solution uses the VPC3+S, which is an evolution of the VPC3+C (Solution 1). The VPC3+S has a parallel and a serial bus interface (I2C or SPI) for data and address.

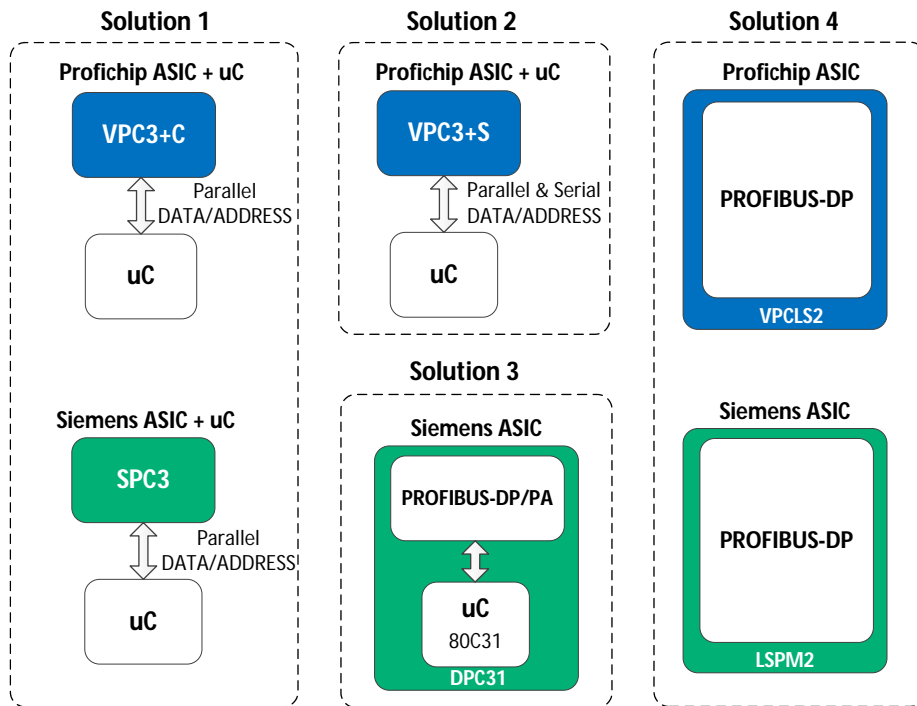


Figure 4.5: Solutions for the Profibus ASIC module.

In case of using Serial Peripheral Interface (SPI), the ASIC interface (address+data) is established using four wires, saving exactly 15 IO pins of the microcontroller, which can be used for other tasks. This is the main advantage of using the VPC3+S over the VPC3+C.

Solution 3 uses only a single IC from Siemens that comprises the Profibus-DP communication module and the processing unit (microcontroller). Thus, there is no need of an external microcontroller as in the previous solutions. This Siemens ASIC also supports the Profibus-PA, which is over-engineered for this application.

The solution 4 is the simplest one and is used only for simple slave applications. In this solution two ICs, the Profichip VPCLS2 and the Siemens LSPM2 can be used. These ICs are used when the application only require digital IOs and no processing is needed. The current application has to acquire, not only digital inputs but also analog ones, which makes this this solution impracticable.

Regarding to the amount of pins that are necessary to interface with the VPI supply, how the different signals are going to be acquired, the microcontroller-ASIC interface, the complexity of the circuit and PCB and, finally, the price estimation of using an internal or external microcontroller, the solution 2 presented in Figure 4.5 was chosen. The VPC3+S, having a serial interface, allows a significantly reduction of the required microcontroller's IO pins, and thus the circuit complexity and the price of the microcontroller are lower than the other solutions. The possibility of choosing a better suitable microcontroller, instead of using the DPC31 internal one, was also important to choose the solution 2.

The VPC3+S is therefore the Profibus ASIC used in the PROFIS board.

### 4.3.3 VPC3+S Overview

The VPC3+S [15] from Profichip is a Profibus-DP communication ASIC with a processor interface for intelligent slave applications. The VPC3+S handles the message and address identification, the data security sequences and the Profibus-DP protocol processing. All the protocol timers and monitoring functions are integrated in the IC. Therefore, almost the entire performance capability of the external controller is available for the application itself. In Figure 4.6, the block diagram of the VPC3+S is presented. As illustrated, the main internal blocks of the VPC3+S are the physical unit, 4kB RAM, microsequencer, idle timer, watchdog, interrupt controller and the bus interface interface unit.

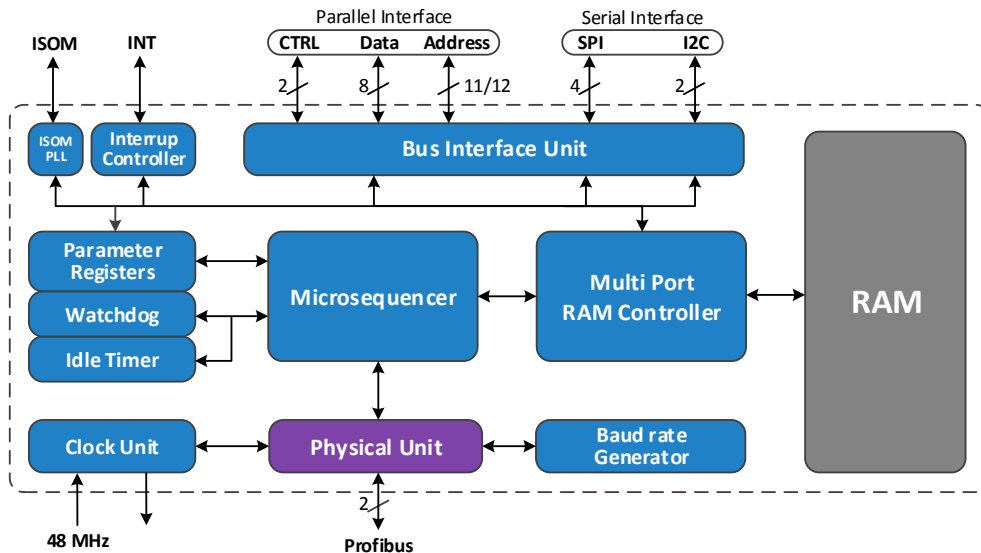


Figure 4.6: Internal modules of the VPC3+S [15].

The physical unit of the VPC3+S converts the asynchronous serial Profibus data stream into internal parallel data or vice-versa. Data is synchronized with the ASIC clock and processed by the microsequencer (MS). The VPC3+S is capable of automatically identifying and controlling transmission rates up to 12 Mbps. The baud rate generator obtains the transmission clock from the ASIC clock unit.

The integrated 2/4 kByte RAM operates as a dual port memory, which serves as an interface between the VPC3+S and the firmware application. It contains the procedure-specific parameters, such as: buffers lengths, modes of operation, status register and station address [15]. The Microsequencer controls the entire operation of the VPC3+S. It contains the DP-Slave state machine. The Idle Timer controls the correct timing of the DP-telegrams according to the Profibus-DP standards and especially controls the idle time before the next request telegram may occur. The watchdog timer observes the entire com-

munication and operates in three different states: BAUD\_SERACH, BAUD\_CONTROL and DP\_CONTROL (Profibus-DP state machine). If the watchdog is not re-triggered within the parameterized time (e.g. if the master application fails), a fail safe mode is activated and the output buffers are switched automatically off [15].

The Interrupt Controller of the VPC3+S informs the user firmware application about indication messages and various error events. Up to a total of 16 events are stored in the interrupt controller. The Bus Interface Unit (Parallel Interface, SPI and I2C) is a configurable synchronous/asynchronous 8-bit interface module compatible with several types of microcontrollers and processors. The user can directly access the internal RAM or the parameter registers via the 11/12 bit address bus (2k/4k RAM) [15].

#### 4.3.4 VPC3+S RAM

The 2/4 kByte internal RAM serves as a communication interface between the VPC3+S and the firmware application, which controls the Profibus-DP state machine. In the 2 kByte mode, the RAM is divided in 256 segments of memory with 8 bytes each, resulting in 2048 bytes of memory space. If 4 kByte RAM mode is used, the RAM comprises 256 segments, with 16 bytes instead of 8. The ASIC RAM can be divided into three different parts: the control parameters, the organizational parameters and the DP-VO, V1 and V2 buffers, as shown in Table 4.1.

Table 4.1: 2k RAM organization [15].

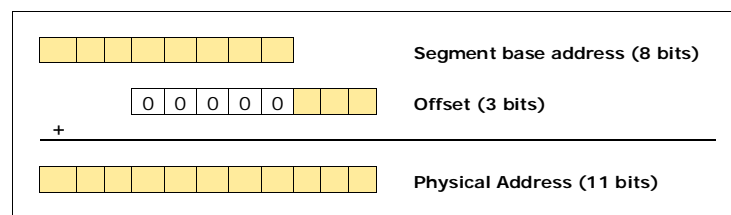
N° of Cells	Addr HEX	Function
22	000X	VPC3+S Control Parameters
	....	
	015X	
42	016X	Profibus-DP Organizational Parameters
	....	
	03FX	
1984	040X	DP-V0, V1 and V2 Buffers
	....	
	7FFX	

The Profibus control parameters are located in the first 22 memory cells (addresses 00hex - 15hex). These cells can be either read-only or write-only and they control the operation of the interrupt controller register and also the operation of the Profibus mode. The VPC3+S status register can also be checked in these cells. The organizational parameters comprise 42 cells of memory starting in 16hex. The entire buffer structure for the DP service access point (DP-SAPs) is based on these parameters. In addition,

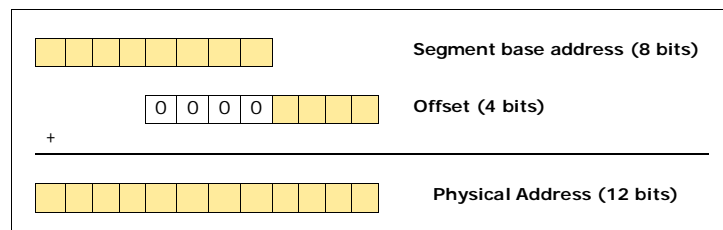
general parameter data, Profibus station address and identification number are also stored in these cells. These parameters can be written and read [15].

In the DP-V0, V1 and V2 buffers (addresses 40hex - 7FFhex), the configuration and parametrization of the DP-master is stored. The input and output data, the diagnosis and the auxiliary buffer are also located in these cells.

To access the RAM in 2k mode, a segment base address plus 3 bits of an offset byte are used, resulting in 11 bits ( $2^{11} = 2048$ ), as depicted in Figure 4.7(a). If the 4k mode is used, 12 bits of physical buffer address are necessary ( $2^{12} = 4096$ ), as shown in Figure 4.7(b).



(a)



(b)

Figure 4.7: Physical buffer address of the RAM (a) 2kB mode and (b) 4 kB mode.

### 4.3.5 ASIC Mode Registers

The VPC3+S control parameters are combined into 4 mode registers. The microcontroller accesses and configures these registers through SPI to set up the Profibus mode of operation. In mode register 0, the parameters should be defined only in the offline state (before the start order). The VPC3+S must not exit the offline state until all the control and organization parameters are loaded into the ASIC. During the ASIC operation (Profibus-Dp state machine) some control bits must be changed in the mode register 1. These control bits are divided into two bytes. Separate addresses are used for setting (008Xhex) and resetting (009Xhex). For example, to start the VPC3+S, a '1' should be written in the cell with address 008Xhex and to reset this bit, a '1' should be written in the cell with address 009Xhex. In mode register 2 and 3, parameters can only be edited in offline state [15].

### 4.3.6 Interrupt Controller

The microcontroller manages the Profibus communication. In order to perform it, the microcontroller must be informed about indication messages and various error events that might occur during the communication with the Profibus master. The interrupt controller is an internal module in the VPC3+S that informs the microcontroller about these events. The controller consists of an interrupt acknowledge register (IAR), an interrupt request register (IRR), an interrupt mask register (IMR) and an interrupt register (IR), as depicted in the block diagram of Figure 4.8.

The IRR has the capability to store up to 16 events, which are internally generated by the VPC3+S. The microcontroller can either access the generated interrupts that are stored in the IRR or generate them for debug purposes. The interrupt events can be suppressed via the IMR, which is programmed by the microcontroller with a certain mask. Events that are not masked in the IMR, their corresponding bits are automatically set in the IR and thus, an external interrupt can be generated through a VPC3+S external pin. Each event that was processed by the microcontroller must be acknowledged by the IAR. The interrupt controller does not have priority level and does not provide an interrupt vector. The interrupt controller of the VPC3+S works as a poll system, which means that the application firmware needs to cyclic read out the interrupt events [15].

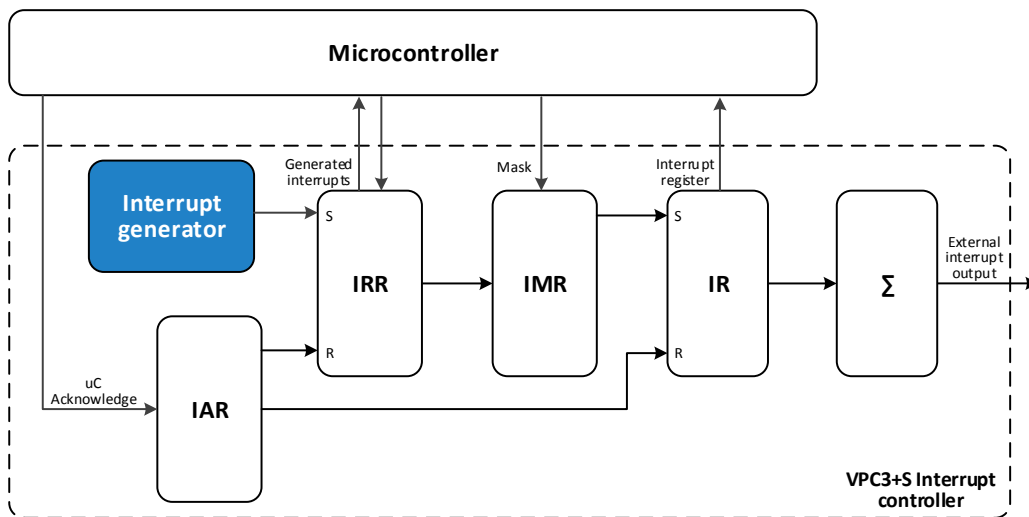


Figure 4.8: Block diagram of the VPC3+S interrupt controller.

### 4.3.7 SPI Interface

The VPC3+S is designed to be interfaced directly with the Serial Peripheral Interface (SPI) protocol. Depending on the VPC3+S inputs status of CPOL (Clock Polarity) and CPHA (Clock Phase), four different SPI modes can be selected. Since the SPI interface

is used in this project, all the unused VPC3+S pins including the parallel data/address bits must be connected to the ground [15]. In Figure 4.9 the interface between the microcontroller and the Profibus ASIC is depicted.

As illustrated, the serial communication bus requires four lines: MOSI (Master Output Slave Input), MISO (Master Input Slave Output), SCK (Serial Clock) and SS (Slave Select). Another line is necessary to reset the ASIC. The interface between the ASIC and the microcontroller is therefore achieved using the SPI bus, instead of the parallel bus that would lead to a huge usage of the microcontroller's IO pins. The microcontroller and the VPC3+S used pins are presented in Tables 4.2 and 4.3, respectively.

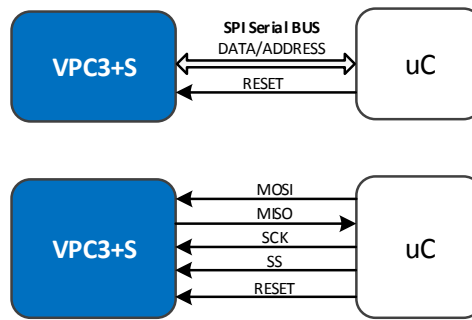


Figure 4.9: Interface between the microcontroller and the VPC3+S.

Table 4.2: Microcontroller SPI pins.

Signal	uC Pin
Serial Data Out (SDO/MOSI)	43 - RC5
Serial Data In (SDI/MISO)	42 - RC4
Serial Clock (SCK)	37 - RC3
Slave Select (SS)	36 - RC2

Table 4.3: VPC3+S pins for the SPI interface.

VPC3+Spin	Signal	I/O	Description	Connect to
9	SERMODE	I	'1' - serial interface	VCC
28	MOT/XINT	I	'0' - not used	GND
33	MODE	I	'0' - SPI mode	GND
2	SPLCPOL	I	clock polarity	VCC/GND
44	SPLCPHA	I	clock phase	VCC/GND
3	SPLXSS	I	slave select	uC (36-RC2)
48	SPLSCK	I	serial clock	uC (37-RC3)
1	SPLMOSI	I	Master-Out-Slave-In	uC (43-RC5)
47	SPLMISO	O	Master-In-Slave-Out	uC (42-RC2)

As shown in Table 4.2, port C of the microcontroller handles the SPI communication. The SPI pins of the VPC3+S are connected to the microcontroller through pin 3 (SPL\_XSS), 48 (SPL\_SCK), 1 (SPL\_MOSI) and 47 (SPL\_MISO). The SPI mode of operation is configured with pin 2 (SPL\_CPOL) and 44 (SPL\_CPHA). Additionally, pin 33 (MODE) selects the serial communication (SPI or I2C), pin 28 (MOT/XINT) is not used in this application, since it is used for the parallel interface configuration and pin 9 (SERMODE) sets the communication as serial interface.

The SPI communication follows the next procedure: first the microcontroller informs the ASIC about the type of operation, 13hex for read byte mode, 03hex for read array of bytes mode, 12hex for write byte mode and 02hex for write array of bytes mode; then the microcontroller sends 2 bytes that correspond to the address of the ASIC RAM's cell (Figure 4.7) and finally, 8 bits of data or more (depending on the type of operation) are written/read into/from the RAM. In Figures 4.10 and 4.11 are depicted four oscilloscope screenshots of two clock speeds, 156.25 kHz and 2.5 MHz, respectively. In each screenshot there are 3 signals: SPI serial clock (top), MOSI (middle) and MISO (bottom).

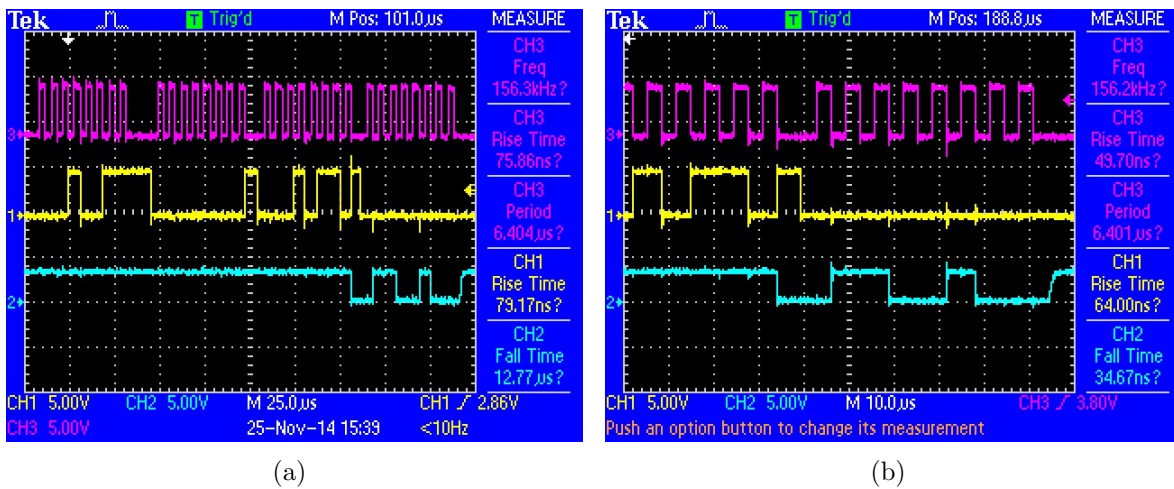


Figure 4.10: Oscilloscope screenshots of the SPI communication at 156 kHz (a) 25 μs/div: (b) 10 μs/div.

To test the SPI communication, a known value (64hex) was written into the Profibus station address cell (16hex). In order to check if the value was correctly saved, the memory cell was read and the result is depicted in Figure 4.10. As can be seen, 3 signals are visible: SPI serial clock, MOSI and MISO. The communication lasts 32 cycles of clock: 8 cycles for the type of operation (read byte), 16 cycles for the RAM's address (16hex) and, finally, 8 cycles to read the station address value that was written into the ASIC memory. The same process was repeated using the maximum speed for the SPI communication imposed by the microcontroller (2.5 MHz). The signals waveform are lightly superimposed with noise, as illustrated in Figure 4.11. Nevertheless the communication still worked perfectly.

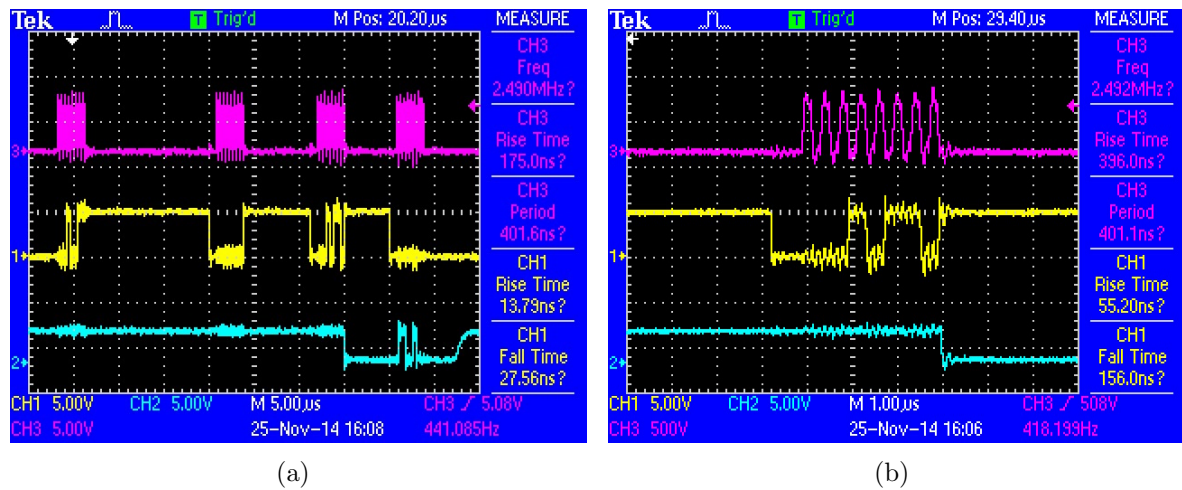


Figure 4.11: Oscilloscope screenshots of the SPI communication at 2.5 MHz (a)  $5 \mu\text{s}/\text{div}$ ; (b)  $1 \mu\text{s}/\text{div}$ .

### 4.3.8 IOs of the Profibus ASIC Module

Various modules of schematics were generated in order to make the PCB connections and the understanding of the schematics easier. As the Profibus-DP interface and the Profibus ASIC modules of Figure 4.1 are part of the Profibus communication, they were merged into one single block schematic, depicted in Figure 4.12. The schematic behind this block can be seen in the Appendix B, Figure B.2.

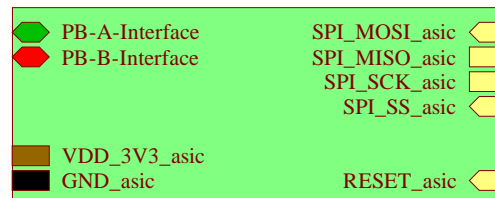


Figure 4.12: ASIC module IOs.

It contains the Profibus-DP interface and the VPC3+S ASIC. On the left side is plugged the Profibus cable with the differential lines A and B, and the 3.3 V power supply. On the right side, there is the SPI interface that connects the ASIC to the microcontroller. In table 4.4 the IO signals are described.

Table 4.4: ASIC module IOs description.

Signal	Direction	Description
PB-A-Interface	I/O	Profibus A line
PB-B-Interface	I/O	Profibus B line
SPL_MOSI_asic	I	Master-Out-Slave-Input
SPL_MISO_asic	O	Master-In-Slave-Output
SPL_SCK_asic	I	SPI clock
SPL_SS_asic	I	SPI slave-select
RESET_asic	I	ASIC reset
VDD_3V3_asic	I	ASIC
GND_asic		power supply

## 4.4 Controller - Microcontroller Module

The controller of the PROFIS board (in Figure 4.1) comprises two blocks: the Profibus ASIC and the microcontroller. The microcontroller block is based on the PIC18F4680 from Microchip [16]. In the next sub-sections its main characteristics and configurations for this application are presented.

### 4.4.1 General Overview of PIC18F4680

The microcontroller is an essential part of the signals acquisition and actuation, and also in the control of the entire application. It runs the firmware application that controls the Profibus-DP state machine, which is implemented inside the VPC3+S ASIC. The choice of using the microcontroller was based on the needed IO ports and integrated modules. The chosen microcontroller was the PIC18F4680. It has a program memory of 64 kB, 20 interrupt sources, 5 ports IO, 1 MSSP (Master Synchronous Serial Port), 1 USART (Universal Synchronous/Asynchronous Receiver/Transmitter) and 1 Analogue to Digital Converter (ADC) module with 11 multiplexed channels, as shown in Table 4.5.

The PIC18F4680 is a 8-bit microcontroller that belongs to the 18F family of Microchip. The manufacturer provides a specific library (XC8) for these microcontrollers. The library and the development software (MPLABX<sup>®</sup>) can be downloaded free of charge from Microchip's website.

### 4.4.2 Ports Distribution

The IO ports of the microcontroller provide the interface with the external circuits. They can be configured as digital IOs, ADC channels, interrupt outputs, and SPI/I2C com-

Table 4.5: Microchip PIC18F4680 main features.

<b>Features</b>	<b>PIC18F4680</b>
Operating Frequency	Up to 40 MHz
Program Memory (Bytes)	65536
Data Memory (Bytes)	3328
EEPROM (Bytes)	1024
Interrupt Sources	20
IO Ports	A, B, C, D and E
Timers	4
Serial Communicatuos	MSSP, USART
10-bit ADC	11 Input Channels
Comparators	2

munication lines, depending on the available hardware modules integrated in the microcontroller. For this project, the distribution of the microcontroller's IO ports is shown in Table 4.6.

Table 4.6: PIC18F4680 IO ports distribution.

<b>Microcontroller IO Port</b>	<b>Function</b>
A	ADC - VPI's threshold levels
B	VPI's status information
C	SPI and USART serial communications
D	Profibus station address
E	High voltage turn ON/OFF

The used microcontroller contains 5 IO ports, Port A, B, C, D and E. Port A is essentially used for analog signals, acquiring the analog VPI's threshold levels. An external 10 MHz crystal is used to provide a higher clock rate. To read out the digital information about the status of the VPI controller, Port B is used. This port has a digital output connected to a debug LED. Port C is dedicated to the serial communications: SPI for the ASIC interface and USART for LabVIEW. Port D reads the Profibus station address, which is configured by the user through a 8-DIP-switch. Finally Port E is used to command the high-voltage supply, a spare SPI slave-select and an on-board reset button. In Figure 4.13 is depicted part of the PROFIS schematic relative to the microcontroller, where the connections of each port can be seen. From the chosen net names it can be easily identified the function of each IO.

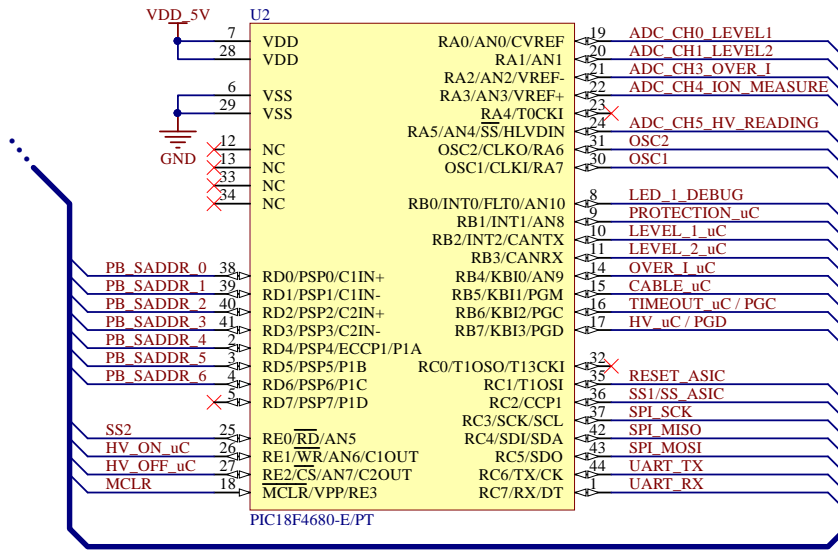


Figure 4.13: Microcontroller IO ports.

The analog and digital signals from the VPI controller (port A and B respectively) must first be conditioned in order to make their ranges and levels compatible with the microcontroller IO specifications. For example, the VPI status information, which are on-off states, are acquired through the voltage drop of the corresponding LEDs in the VPI’s front panel. For the red LEDs, the voltage drop is typically 1.8 V when it is on. This 1.8 V signal is connected to a comparator, in the PCB IO module, that will switch its output depending on a given threshold value, producing 0 V (LED off) and 5 V (LED on).

### 4.4.3 IOs of the Microcontroller Module

The high-level schematic of the microcontroller module is depicted in Figure 4.14. All the IOS related to the serial communications, SPI for VPC3+S and USART for LabVIEW, are placed on the left side. The 5 V power supply and one digital output, which is the SPI spare slave select (SS2) that is connected to an on-board SPI bus header, are also on the left side.

On the right side of the hierarchical block there are seven digital inputs that are related to the VPI status information; two digital outputs to turn on and off the high-voltage; five analog inputs with the threshold levels, the ionization measure, the high-voltage level and seven digital inputs with the Profibus station address, which is set through a 7-bit on-board DIP-switch. The whole schematic of the microcontroller can be see in Appendix B (Figure B.3).

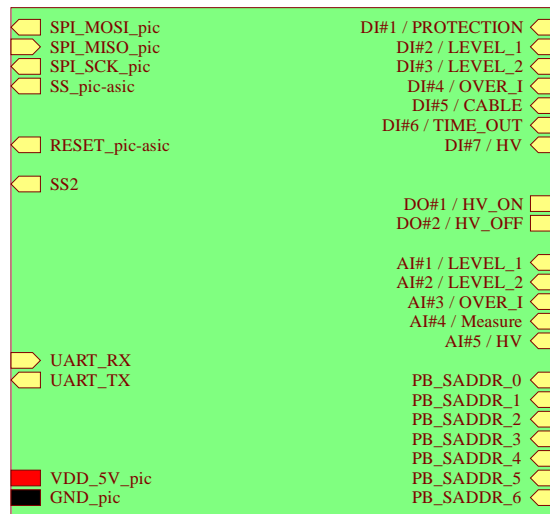


Figure 4.14: Microcontroller module IOs.

The description of the IO signals of the microcontroller module is presented in Table 4.7. The digital inputs [DI#1 - DI#7] and the analog inputs [AI#1 - AI#5] are signals acquired from the hands-unit connector by the IO module. Those are actually the ones that are going to be sent into the Profibus network and thus, will be available in the Profibus master.

Table 4.7: IO signals description of the microcontroller module.

Signal	IO	Description
SPI.MOSI_pic	O	Master-Out-Slave-Input
SPI.MISO_pic	I	Master-In-Slave-Output
SPI.SCK_asic	O	SPI serial clock
SS_pic-asic	O	SPI slave select
RESET_pic-asic	I	ASIC reset
SS2	O	Additional slave select
UART_RX	I	USART RX
UART_TX	O	USART TX
DI#1/PROTECTION	I	Protection indicator
DI#2/LEVEL.1	I	Level 1 threshold
DI#3/LEVEL.2	I	Level 2 threshold
DI#4/OVER.I	I	Over current threshold
DI#5/CABLE	I	Cable indicator
DI#6/TIME.OUT	I	Time out indicator
DI#7/HV	I	High voltage indicator
AI#1/LEVEL.1	I	LEVEL 1 set potentiometer
AI#2/LEVEL.2	I	LEVEL 2 set potentiometer
AI#3/OVER.I	I	Over current set potentiometer
AI#4/Measure	I	Ionization measure
AI#5/HV		High voltage value
DO#1 / HV_ON	O	Turn-on the high-voltage
DO#2 / HV_OFF	O	Turn-off the high-voltage
PB.SADDR [7..0]	I	DIP-switch for Profibus station address
VDD_5V_pic		Microcontroller power supply
GND_pic	I	

## 4.5 USART

A USART module was added to the board to serve as a debug tool and for the LabVIEW interface. It contains a mini Universal Serial Bus (USB) connector, a USB to RS232 converter and two debug LEDs for the TX and RX communication lines. The FT232R from FTDI [17] emulates a general-purpose serial port when connected to a PC through a USB cable. Operating at USB Hi-Speed (480 Mbps) rate, this IC can be configured for a wide variety of asynchronous and synchronous serial standards, such as JTAG, SPI, I2C and USART. In Figure 4.16, part of the schematic of the USART module can be seen.

In this application the FT232R is being used as a USART communication module. The on-board USART module allows the data exchange between the LabVIEW software and the PROFIS board. This communication can also be used as another type of interface

to the VPI controller. The whole schematic of the USART module can be seen in the Appendix B (Figure B.4).

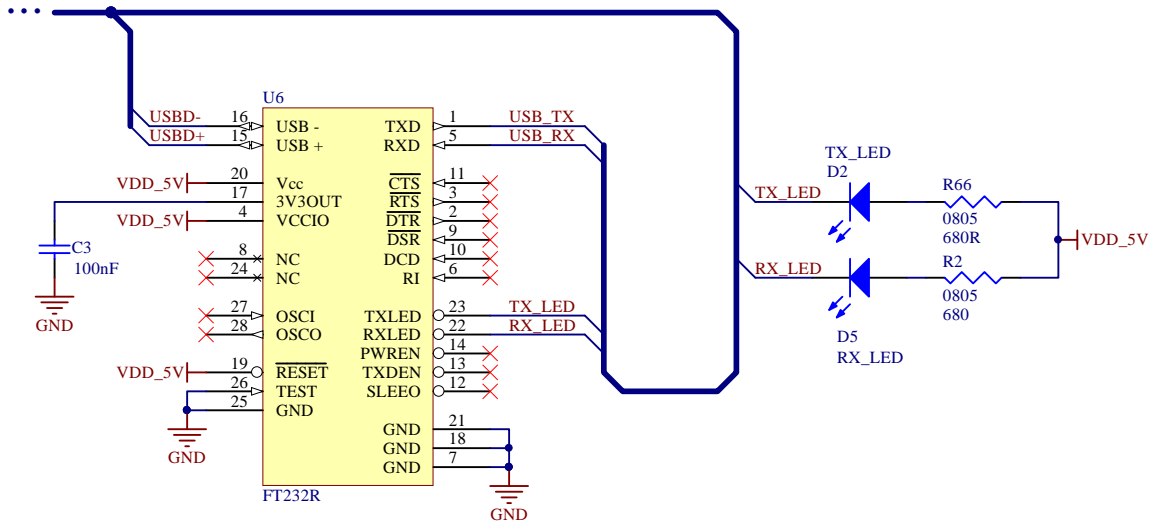


Figure 4.15: Schematic of the USB-RS232 IC converter.

### IOs of the USART Module

The block diagram (from the hierarchical schematic) of the USART module can be seen in Figure 4.16. On the left side, the D- and D+ of the USB protocol are connected. Also, the 5 V power supply is provided on the same side. On the right side, the TX and RX IOS related to the RS232 communication are shown.

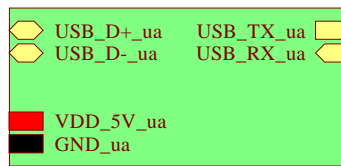


Figure 4.16: USART module IOs.

## 4.6 IO Module

The IO module of the PROFIS is the block that does the interface between the microcontroller and the VPI supply. It acquires the desired hands-unit signals, converting their ranges according with the microcontroller’s IO limits. This block is mostly composed by rail-to-rail operation amplifiers (OpAmps) and optocouplers. As stated before, the PROFIS board has several digital and analog inputs and two digital outputs. In the IO

module, the digital inputs (VPI status information) are connected to comparators in order to generate 0 V or 5 V when the corresponding LED is off or on, respectively.

The analog signals are connected to OpAmps working as followers (voltage buffers) to make the correct voltage transfer to the ADC channels of the microcontroller. The interposed voltage buffer prevents the second circuit (IO module) from loading the first circuit (VPI supply) unacceptably and thus, interfering with its desired operation.

To turn on and off the high-voltage supply of the vacuum pump, high-voltage relays are required. To isolate the PROFIS from the relays, two optocouplers are used. These, are actuated by signals received from the microcontroller. The signals can be seen in Tables 3.3 (outputs) and 3.4 (inputs). As mentioned before, inputs are considered signals to be sent to the Profibus master and outputs the control signals to the VPI supply. The whole schematic of IO module can be seen in Appendix B, Figure B.6.

### 4.6.1 Digital Inputs

The VPI status information are digital inputs, which are acquired by the PROFIS board and sent to the Profibus master. The signals PROTECTION, CABLE, OVER I and TIME OUT are acquired using the same principle: through the voltage drop of the LEDs in the VPI's front panel. As long as the LED is on, the output of the comparator is positively saturated. When the LED is off, there is no voltage drop in the LED and the comparator output is low. In Figure 4.17 the acquisition schematic of one of the VPI status information signals is shown. This schematic stands for the PROTECTION signal but, as mentioned before, the other schematics of the CABLE, OVER I and TIME OUT signals are exactly the same.

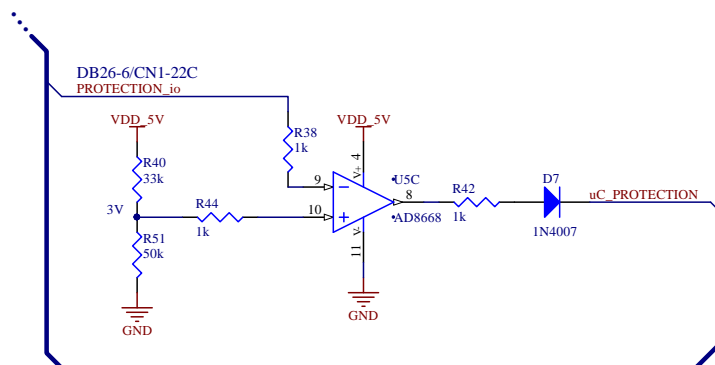


Figure 4.17: Circuit used to acquire the digital input PROTECTION.

The two remaining digital inputs, LEVEL 1 and LEVEL 2, indicate whether the related levels have been reached or not. The only way to access these signals is through the LEDs in the hands-unit, which leads to one restriction: the signals are acquired with

the LED voltage drop, so if the hands-unit is unplugged, there is no voltage drop and the signal is unreachable. To be able to work with or without the hands-unit plugged in the VPI controller, a pair of relay contacts are used (see Table 3.2). Figure 4.18 represents the used circuit for the acquisition of LEVEL 1. Two resistors were added (R34 and R35), and two situations are faced: hands-unit connected or hands-unit disconnected from the VPI controller.

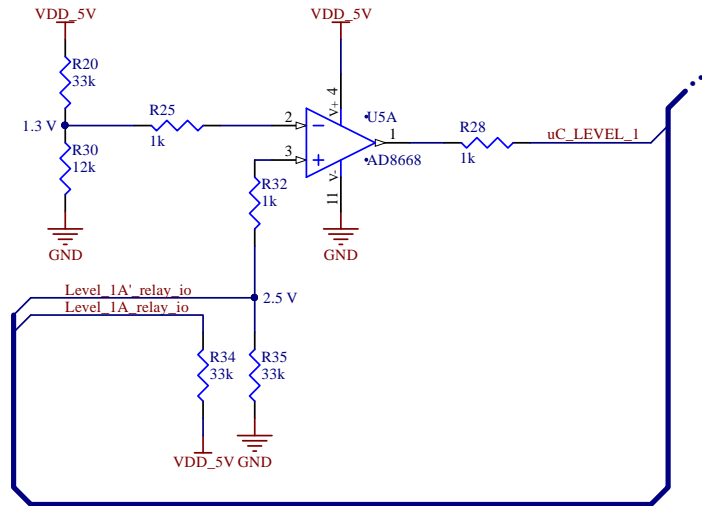


Figure 4.18: Circuit used to acquire the digital input LEVEL 1.

As mentioned before, two different situations can occur: hands-unit connected or disconnected. When the hands-unit is connected and if the threshold was reached, the relay contact is closed and there is around 2 V (voltage drop of the LED) between R34 and R35. Therefore,  $I_{n+}$  is greater than  $I_{n-}$  and the comparator is positively saturated (5 V to the microcontroller). If the threshold was not reached, the relay contact is open and there is around 0,12 V between R34 and R35, caused by the internal hands-Unit hardware.

If the hands-unit is disconnected, there is no voltage drop to control the OpAmps, so alternatively the relay contacts are used. When the coil of the relay is powered, the position of its contacts change and a voltage divider is formed with R34 and R35, which applies 2.5 V in the  $I_{n+}$  and the OpAmp saturates positively.

## 4.6.2 Digital Outputs

The digital outputs of the PROFIS board control the high-voltage supply for the VPI pump. To turn on and off the high-voltage, two signals from the microcontroller are connected to optocouplers that provide an output voltage of 0 V or 12 V to be delivered in the coil of the high-voltage relay, as can be seen in Figures 4.19 and 4.20.

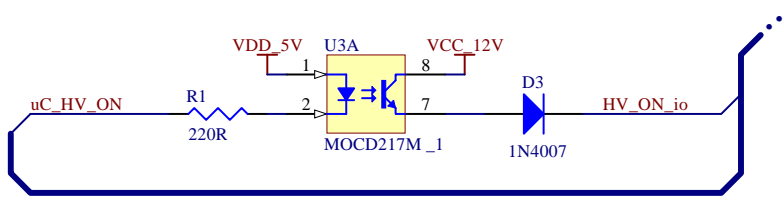


Figure 4.19: Turn on the high-voltage.

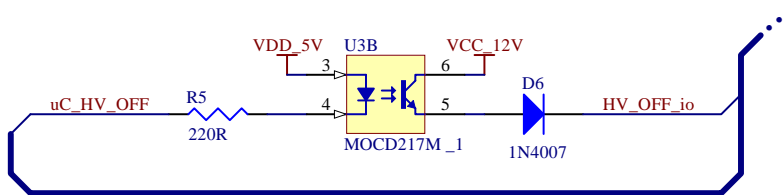


Figure 4.20: Turn off the high-voltage.

The digital outputs of the microcontroller are connected to resistors R1 and R5. For instance, in Figure 4.19 when the signal `uC_HV_ON` is 0 V, the internal LED of the optocoupler is on, saturating the phototransistor. In this situation the optocoupler provides 12 V to the high-voltage relay. The PROFIS board is controlling the VPI supply in parallel with the hands-unit and also with the buttons of the VPI supply front panel. In total, three different devices can control the high-voltage supply. The 12 V digital outputs are connected in the same point as the other controls, so 12 V might appear on the cathode of D3 and D6. To prevent a possible damage of the optocouplers, two protection diodes were added.

### 4.6.3 Analog Signals

The analog input signals from the VPI supply are connected to the 10-bit ADC of the microcontroller. The maximum input voltage range of the ADC ( $V_{REF+} - V_{REF-}$ ) is equal to the voltage of the power supply of the microcontroller (5 V). The IONIZATION MEASURE, which is a 10 V range signal, is the only that needs to be divided by 2 before being connected in the microcontroller's ADC. To make the correct voltage transfer, buffers were added. The minimum step of the ADC, known as Least Significant Bit (LSB) can be calculated using the Eq. 4.1.

$$LSB = \frac{V_{REF+} - V_{REF-}}{2^n} \quad (4.1)$$

If  $V_{REF+} = 5$  V,  $V_{REF-} = 0$  V and resolution 10 bits ( $n = 10$ ), the minimum step of the ADC is 4.88 mV.

The schematics for the IONIZATION MEASURE and for the LEVEL 2 THRESHOLD are depicted in Figures 4.21 and 4.22. For the LEVEL 1 THRESHOLD, HV READING and OVER CURRENT of Table 3.4, the circuit is the same as Figure 4.22.

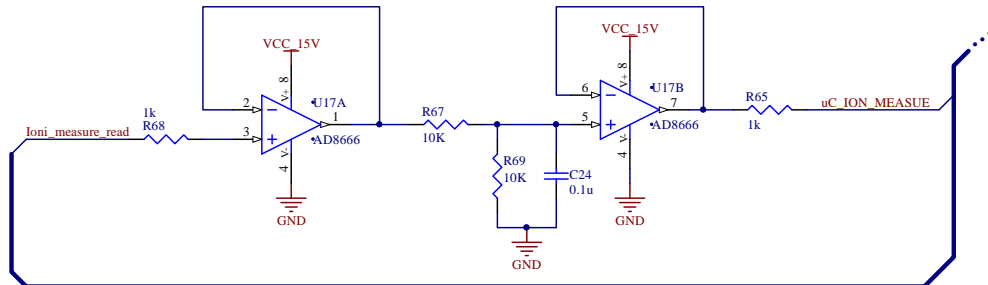


Figure 4.21: Conditioning circuit of the analog input IONIZATION MEASURE.

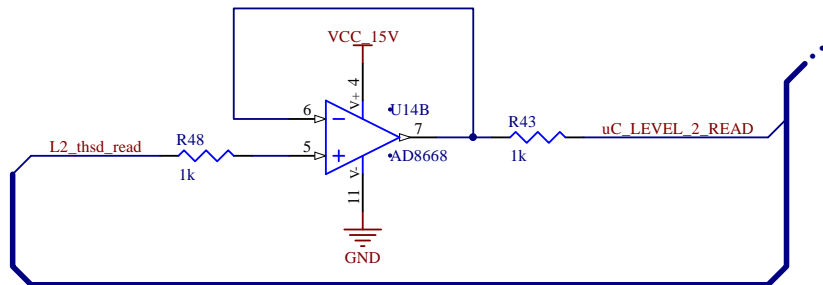


Figure 4.22: Circuit used to acquire the analog input LEVEL 2 THRESHOLD.

#### 4.6.4 IOs of the IO Module

The inputs and outputs of the IO module can be seen in Figure 4.23. The IO module implements the interface between the microcontroller and the on-board IDC-26 connector, where all the PROFIS signals are connected.

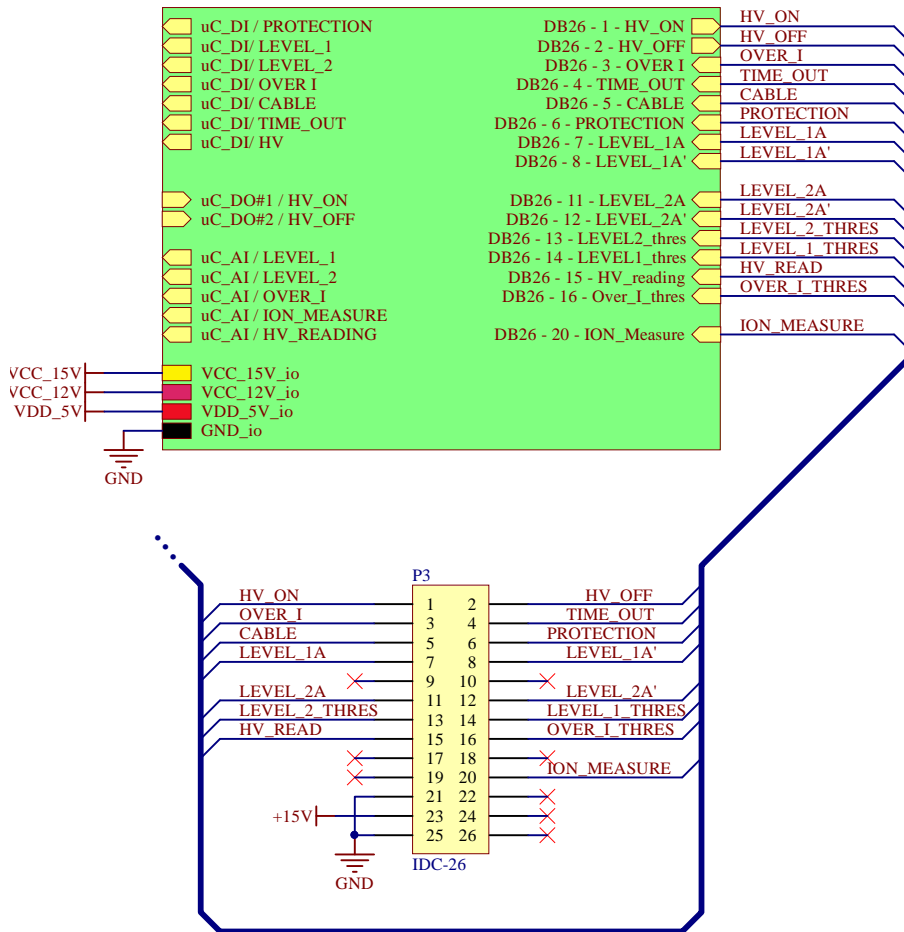


Figure 4.23: Inputs and outputs of the IO module.

## 4.7 Power Supply Module

The PROFIS board requires four different voltages sources (15 V, 12 V, 5 V and 3.3 V) to power all the electronic modules. These voltages are generated by the power supply module of the PROFIS from a 15 V supply, which is provided by the VPI controller. Three on-board voltage linear regulators, which are connected in cascade, provide 12 V, 5 V and 3.3 V, with 15 V as input. In Figure 4.24 the schematics of the three linear regulators is presented.

The ADP7105 [18] from Analog Devices is used to generate the 12 V. The adjustable output value can be calculated through Eq. (4.2) [18]. The 5 V and the 3.3 V are provided by the fixed linear regulators, L7805 [19] from Texas Instruments and ADP1706 [20] from Analog Devices, respectively.

$$V_{OUT} = 1.22 \times \left(1 + \frac{R3}{R46}\right) \quad (4.2)$$

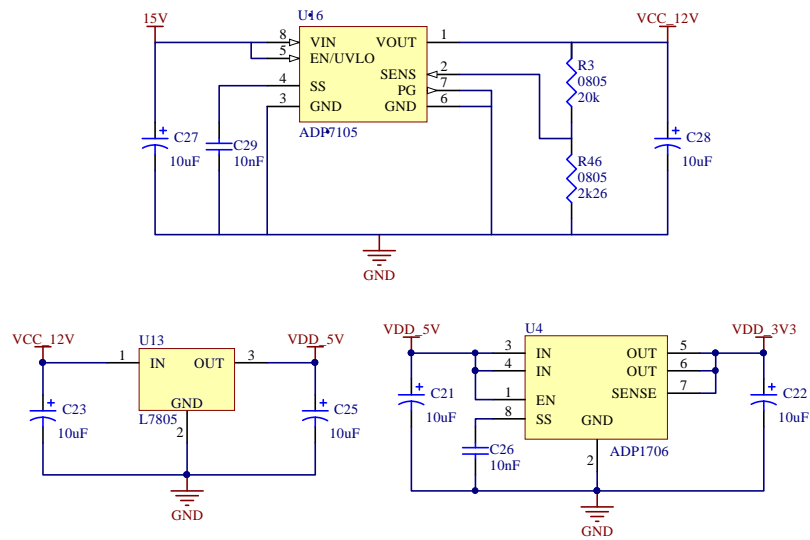


Figure 4.24: DC-DC converters.

### 4.7.1 IOs of the Power Supply Module

The power supply module of the PROFIS generates the four voltages from a single 15 V power supply. As stated before, the 15 V voltage source is the only input of this module and the output voltages are 12 V for the high-voltage relays, 5 V for the microcontroller and USART modules and 3.3 V for the ASIC and Profibus-DP interface modules. In Figure 4.25 the high-level power supply block diagram is illustrated. On the left side there is the 15 V input power supply and on the right side there are the output voltages to power all the PROFIS modules.

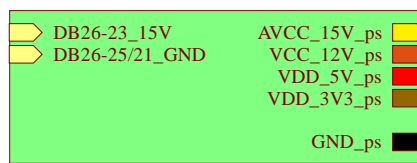


Figure 4.25: Power-supply module IOs.

## 4.8 PROFIS

### 4.8.1 Top Level Schematic

Due to the huge amount of hardware components and the number of connections of the PROFIS board, would not be possible to merge all of them in one single schematic. The understanding of the schematics would be more complex. Therefore, a main schematic that connects all the described modules was created, as shown in Figure 4.26.

The high-level hierarchical schematic comprises five modules: Profibus-DP interface + ASIC, microcontroller, IO module, USART and power supply. Also a SPI spare bus, two LEDs for power supply indication, Profibus dip-switch and an IDC-26 connector, which can be seen in the schematic. All these modules were explained in previous sections.

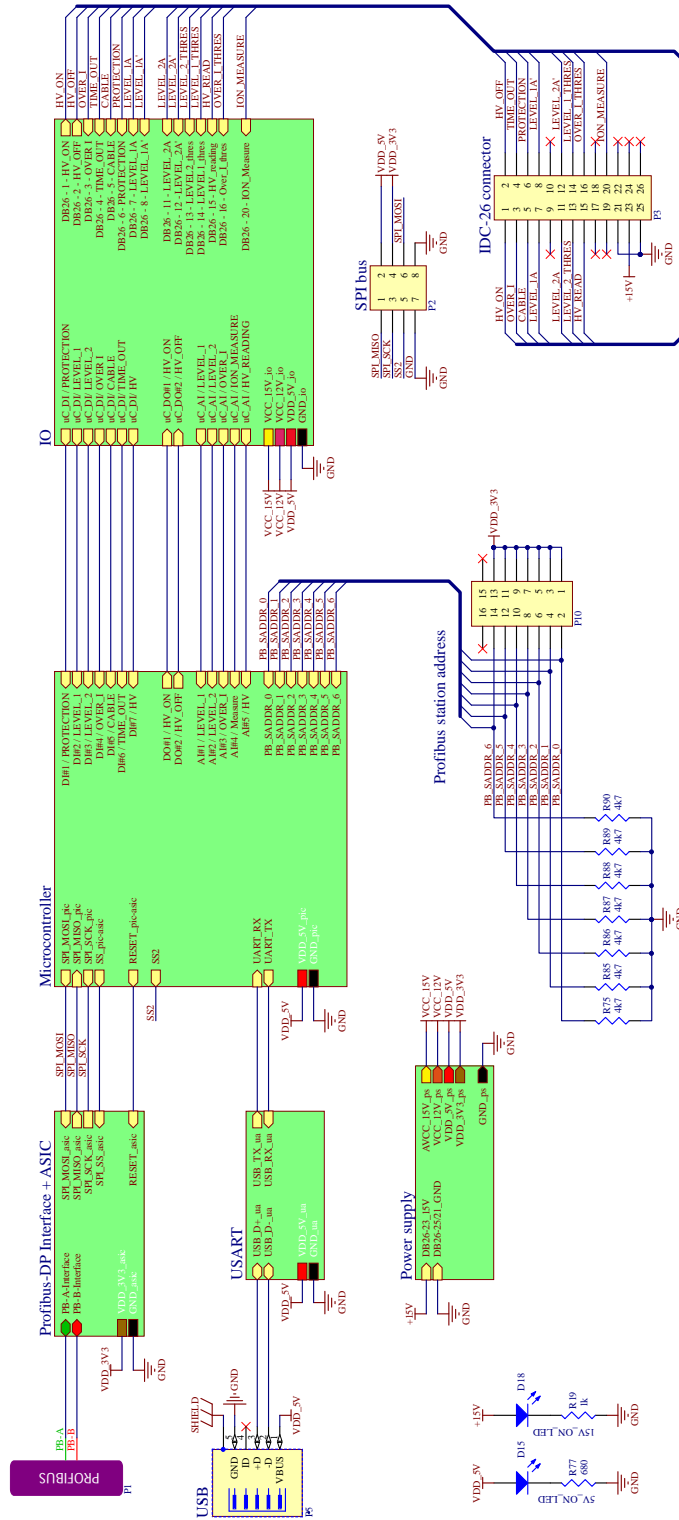


Figure 4.26: Main schematic of the PROFIS board.

## 4.8.2 PCB and Enclosure Box

At the beginning of the project, the PROFIS board was intended to be a general Profibus slave with a certain number of digital and analog IOs, instead of a dedicated Profibus slave for VPIs. A first prototype was developed and after some adjustments in the components, the PCB was working as desired. However, it has been decided to make a single dedicated PCB for VPIs, as the Profibus interface was working perfectly with the firmware application and thus, would be easy to replicate this board with more peripherals and digital/analog IOs. The first developed prototype can be seen in Appendix C (Figure C.1).

The final PCB prototype was totally produced and assembled at CERN. It has 4 layers and uses the standard FR-4 glass epoxy dielectric. The thickness of the copper is  $36\ \mu\text{m}$  and the total height of the PCB is 1.6 mm. The PCB has 4 layers, 2 internal layers for power and ground planes, and other two layers, top and bottom for routing signals, as shown in Figure 4.27. Several power supplies are needed for the different PCB modules, therefore 3 isolated power planes (+12 V, +5 V and +3.3 V) were built in the power plane layer. This layer is illustrated in Appendix C (Section C.2).

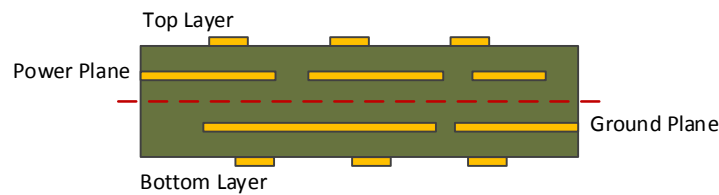


Figure 4.27: PCB layer stack.

The PCB is organized as shown in Figure 4.28. The Profibus interface contains the DB-9 Profibus connector as well as the RS485 driver. The controller is formed by two modules, the ASIC VPC3+S and the microcontroller PIC18F4680. The microcontroller is programmed through a RJ11 female connector (microcontroller programmer). A reset module with a tactile button was added to the PCB. The IO module serves as an interface between the IDC-26 connector and the microcontroller. The DC linear regulators are located on the top of the PCB, next to the optocouplers. Finally the USART module with the mini-USB connector is on the right side of the IO module.

A first prototype was developed and tested. After some modifications in the routing of the PCB and the firmware application working as desired, a final PCB prototype was designed. The final version of the PROFIS board (Figure 4.29) is an intelligent Profibus slave dedicated for sputter ion pumps. The PROFIS is dedicated for CERN's VPI controllers, however depending on the type of the application, the microcontroller can easily be replaced by other and a new specific Profibus slave with different number of IOs can be designed .

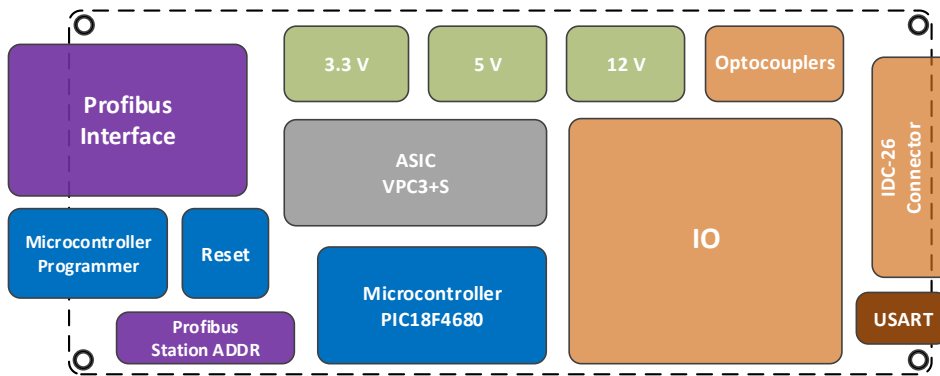
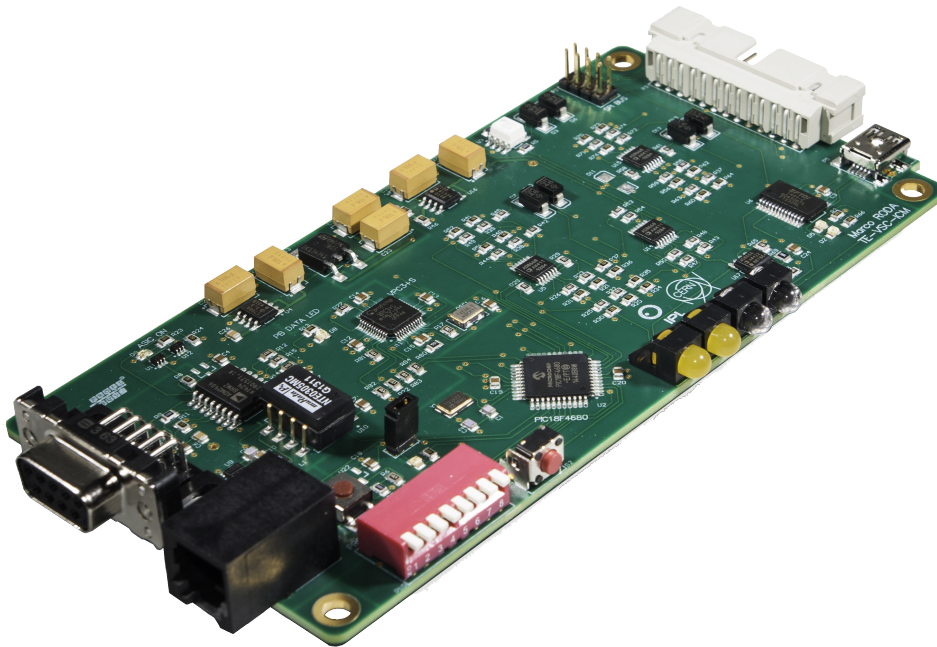
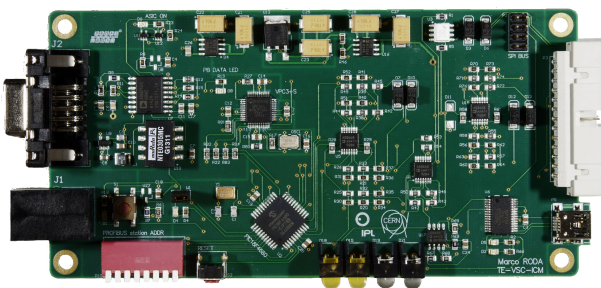


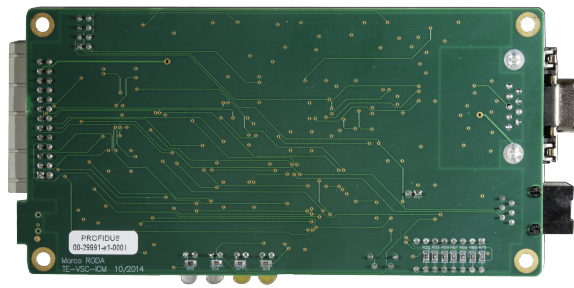
Figure 4.28: PCB organization.



(a)



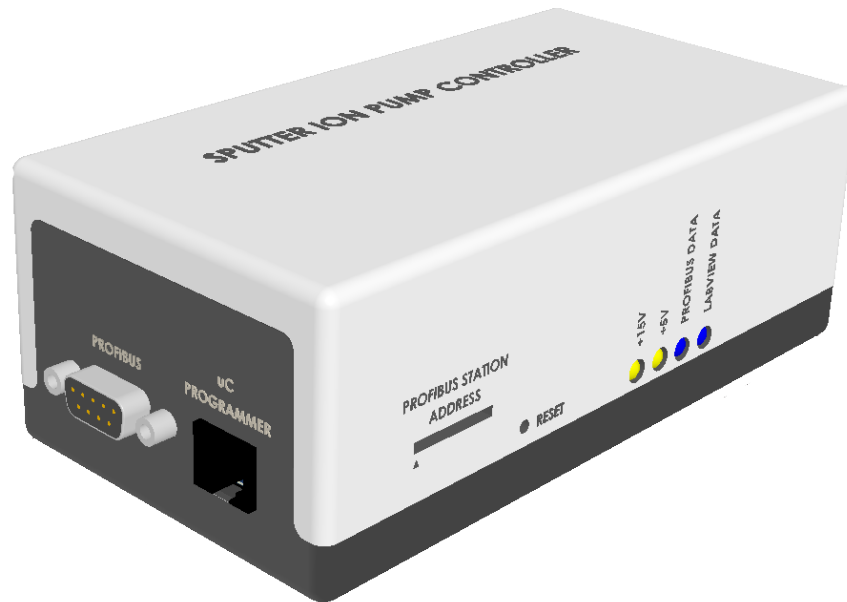
(b)



(c)

Figure 4.29: PROFIS final PCB prototype.

To enclosure the PCB and make it more robust, a plastic box was designed using a 3D CAD software. The final result is shown in figure 4.30. The aim of producing such box is to attach the PROFIS board on the VPI supply.



(a)



(b)

Figure 4.30: Designed plastic box for the PROFIS board.



# Chapter 5

## Firmware Implementation

This chapter describes the application firmware, the state machines behind it and the software interface (Simatic STEP7 and LabVIEW). The microcontroller is the core of the electronic card running the firmware code that allows the communication with the ASIC, and thus achieving a Profibus connection. It also allows the acquisition of the analog and digital signals of the VPI supply and the control of the high-voltage to the pump.

The user interface is established through two software applications developed for this project. A Simatic STEP7 project and a simple hardware configuration (one master and one slave) were created in order to test the correct remote communication using Profibus-DP. Additionally a LabVIEW panel was developed to control and monitor the pump locally.

### 5.1 Application Firmware of the PROFIS

To handle the Profibus-DP state machine, a microcontroller is needed. The communication process can be seen at two different stages (Figure 5.1). First the microcontroller (SPI master) configures the ASIC (SPI slave) with the desired mode of operation under the Profibus-DP standards [ref to Profibus]. Once the Profibus ASIC is configured, the PLC (Profibus-DP master) sends a request to communicate with the PROFIS (Profibus-DP Intelligent Slave) testing its configuration, and if there is a match between both configurations, the data exchange begins. The core of the Profibus source code was provided by Profichip and was modified to fit the current application. Functions of IO write/read, RAM address and SPI write and read were all written in C code language.

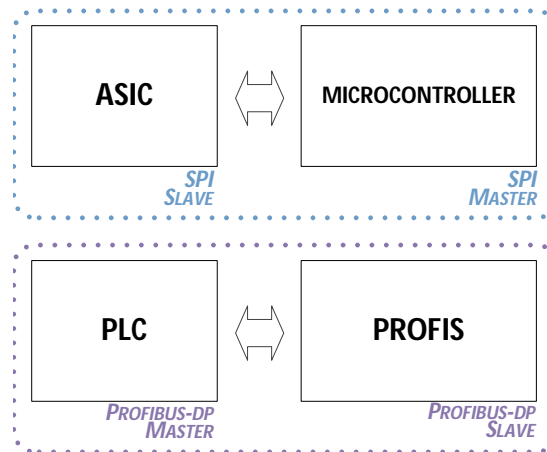


Figure 5.1: Master-Slave communication principle.

### 5.1.1 Main Program Flowchart

The main program flowchart of this application is presented in Figure 5.2. Before entering in data exchange mode, the microcontroller and the VPC3+S must be initialized. In the microcontroller, the hardware modules: ADC, USART, timers, interrupts and IO ports must be configured. In the VPC3+S, the Profibus modes of operation, as well as the station address are loaded. The high-voltage power supply is turned off, as default configuration, and the controller enters in the Profibus-DP state machine.

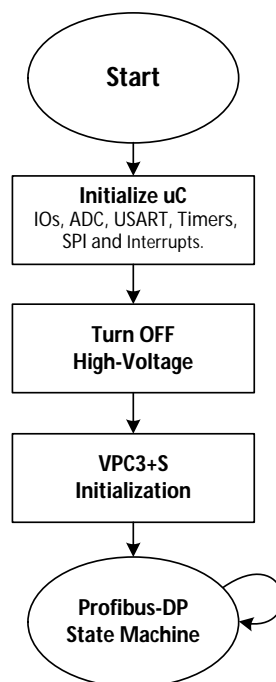


Figure 5.2: Main program flowchart of the software application.

## 5.1.2 Microcontroller Initialization

The microcontroller acts as the bridge between the Profibus-DP communication and the VPI supply (and the VPI pump). Through its internal modules, an IO interface is established achieving a bidirectional remote communication. In the next sub-sections, the initialization of the microcontroller's internal modules is presented.

### IO Ports

The used microcontroller, PIC18F4680 from Microchip, has 5 ports IO (PORT A, B, C, D and E). The IO configuration of these ports is performed using the TRIS registers, according to Table 5.1.

Table 5.1: TRIS registers configuration.

PORT	Register	Configuration bits
A	TRISA	0b11111111;
B	TRISB	0b11111110;
C	TRISC	0b10011001;
D	TRISD	0b11111111;
E	TRISE	0b000;

Knowing that '1' means the pin is configured as input and '0' as output, all the pins of PORTA are configured as input (ADC channels); most of the pins of PORTB are used as inputs (VPI status information) and only one is configured as output (debug LED); TRISC is set up for the serial communications (SPI and USART); PORTD is all configured as input since it is responsible for acquiring the DIP-Switch that defines the Profibus station address; finally PORTE is used to turn on/off the high-voltage, therefore are configured as output.

### Software Interrupts

As mentioned before, the PROFIS board can be interfaced, simultaneously, through two different software applications, Simatic STEP7 and also LabVIEW. Two software interrupts with priority level are used to keep the LabVIEW data updated without disturbing the Profibus state machine, which is always being executed. A high-priority interrupt caused by overflow of microcontroller's Timer 0, is used to send via serial port the analog and digital information of the VPI to LabVIEW. In the opposite direction (LabVIEW to PROFIS) a low-priority interrupt detects upcoming characters from the serial port (USART) actuating on the high voltage, turning on and off.

The high-priority level interrupt is a function that is executed whenever an overflow of Timer 0 occurs. Basically this function converts the digital and analog information from the VPI into a string and send it to LabVIEW. The frequency of the overflow is programmed using the control register of Table 5.2. The control bits are configured as "10000011", meaning that is working as a 16-bit timer, FOSC/4 (internal clock) as increment source and a prescaler of 1:16. Taking into account the configuration of Timer 0, the time between successive overflows is given by Eq. (5.1).

$$T_{0IF} = \frac{1}{F_{CLK}} \times PRESCALER \times 2^n \quad (5.1)$$

With  $F_{CLK} = F_{OSC}/4 = 2.5$  MHz,  $PRESCALER = 16$  and  $n = 16$ , there is an overflow every 0.42 seconds. Whenever a timer overflow is triggered, the high-priority code is executed and the Write.LabVIEW() function is called, sending the information to LabVIEW through USART.

Table 5.2: Timer 0 control register.

<b>TMR0ON: Timer 0 ON/OFF Control bit</b>	
<b>bit 7</b>	1 = Enables Timer 0. 0 = Stops Timer 0.
<b>T08BIT: Timer 0 8-bit/16-bit Control bit</b>	
<b>bit 6</b>	1 = Timer 0 is configured as an 8-bit timer. 0 = Timer 0 is configured as an 16-bit timer.
<b>T0CS: Timer 0 Clock Source Select bit</b>	
<b>bit 5</b>	1 = Transition on T0CKI pin. 0 = Internal instruction cycle clock (CLKO).
<b>T0SE: Timer 0 Source Edge Select bit</b>	
<b>bit 4</b>	1 = Increment on high-to-low transition on T0CKI pin. 0 = Increment on low-to-high transition on T0CKI pin.
<b>PSA: Timer0 Prescaler Assignment bit</b>	
<b>bit 3</b>	1 = Timer 0 prescaler is NOT assigned. 0 = Timer 0 prescaler is assigned.
<b>T0PS2:T0PS0: Timer0 Prescaler Select bits</b>	
	111 = 1:256
	110 = 1:128
	101 = 1:64
<b>bit 2-0</b>	100 = 1:32
	011 = 1:16
	010 = 1:8
	001 = 1:4
	000 = 1:2

The low-priority interrupt is activated when characters from Labview arrive in the input buffer of the USART module. The low-priority function is then executed, turning on or off the high-voltage power supply into the VPI pump.

## 10-bit ADC

The analog-to-digital converter is part of the internal modules of the microcontroller and contains 11 multiplexed inputs distributed along the IO pins. This module allows conversion of an analog input (maximum 5V) to a corresponding 10-bit digital number. The module is configured through five registers:

- ADC Result High Register (ADRESH)
- ADC Result Low Register (ADRESL)
- Control Register 0 (ADCON0= "00000011"): controls the operation of the ADC module (channel selection, enable/disable);
- Control Register 1 (ADCON1= "00001010"): configures the functions of the PORT pins (external ADC voltage references, ADC port configuration);
- Control Register 2 (ADCON2= "10010100"): configures the ADC clock source (programmed acquisition time and digital value justification).

The ADC module is configured with an input clock of  $F_{OSC}/4$  (2.5 MHz), the ADC voltage reference is the one provided by the micro-controller's power-supply (5 V), the result format is right-justified and the conversion time of each word ( $TAD$ ) is  $17.6 \mu s$  ( $TAD = 4TOSC$ ). For further information about the ADC configuration register, refer to PIC18F4680 data sheet [16].

## Serial Peripheral Interface

The Master Synchronous Serial Port (MSSP) module of the microcontroller, configured as SPI mode, allows 8 bits of data to be synchronously transmitted or received. To accomplish communication, typically three pin are used. Additionally, a fourth pin may be used to select the particular SPI slave device. In Table 5.3 there are the microcontroller's pins used for the SPI serial communication. The SPI module is configured with a clock speed of  $F_{OSC}/4$  and mode of operation "00", which means that CPHA (Clock Phase) = '0' and CPOL (Clock Polarity)= '0'.

## USART

USART is one of the two serial communication modules of this microcontroller. It can be configured as a full-duplex asynchronous system to communicate with peripheral devices, such as personal computers or as a half-duplex synchronous system that can communicate with peripheral devices, such as ADC or DAC and serial EEPROMs. The IO pins of the microcontroller for the USART module are available in PORT C (Table 5.4). The operation mode of the USART is controlled through three registers: Transmit Status and Control (TXSTA), Receive Status and Control (RCSTA) and Baud Rate Control (BAUDCON).

Table 5.3: Microcontroller SPI pins.

Signal	uC Pin
Serial Data Out (SDO/MOSI)	43 - RC5
Serial Data In (SDI/MISO)	42 - RC4
Serial Clock (SCK)	37 - RC3
Slave Select (SS)	36 - RC2

Table 5.4: Microcontroller USART pins.

Signal	uC Pin
USART TX	44 - RC6
USART RX	1 - RC7

The USART module contains an internal circuit that sets the baud rate of the communication for both modes, synchronous and asynchronous. The BRG can operate in 8 or 16 bits mode being configured through bit BRG16 of BAUDCON register. The Baud Rate Generator (BRG) is programmed through two registers, SPBRGH and SPBRG [16].

Assuming the microcontroller has a 10 MHz as clock frequency ( $F_{OSC}$ ), a desired baud rate (BRD) of 38400 bps, 16 bits mode operation for BRG and an asynchronous communication, the expression that defines the baud rate can be seen in Eq. (5.2).

$$BaudRate = \frac{F_{OSC}}{[16 \times (n + 1)]} \quad (5.2)$$

The unique unknown value of (5.2) is  $n$  that represents the programmed value in

SPBRGH and SPBRG setting the communication speed.

$$n = \frac{F_{OSC}}{16 \times BaudRate} - 1 = \frac{10 \times 10^6}{16 \times 38400} - 1 = 15.27 \simeq 15 \quad (5.3)$$

The  $n$  value, calculated by Eq. (5.3), is inserted in the SPBRGH and SPBRG register as an integer, thereby an error is introduced which can be calculated through Eq. (5.4).

$$error = \frac{BaudRate_{calculated} - BaudRate}{BaudRate} \times 100 = \frac{39062.5 - 38400}{38400} \times 100 = 1.7\% \quad (5.4)$$

The USART module is therefore configured as asynchronous mode, eight bit of data plus start-bit and stop-bit, and a baud rate of 38400 bps.

### 5.1.3 ASIC Initialization

The VPC3+S is initialized as shown in the flowchart of Figure 5.3. The initialization starts clearing organizational parameters and user-buffers in the ASIC RAM. A memory test is then performed, sending and reading back random information. In the next step the controller accomplishes several tasks, such as: sends the global and application structures for the Profibus communication, sets the slave address, configures the interrupt mask, suppresses useless interrupt events and allocates user buffers length/areas. Once the initialization is finished, the controller gives a start order to the VPC3+S.

The designed board is an independent Profibus slave, having an unique address on the Profibus-DP network. An on-board dip-switch set the address, between 0 and 127 (decimal), which is configured by the user.

### 5.1.4 Profibus-DP State Machine

The Profibus-DP protocol is totally integrated in the VPC3+S ASIC, therefore the controller is not involved in the processing of the Profibus-DP state machine. However, the controller has a crucial function processing the data sent by the master and, depending on the slave configuration, the communication may get through its different states, as shown in Figure 5.4. The Profibus-DP state machine comprises four main states:

1. **Power ON/Reset state:** Initial state following the power up of the controller.

The VPC3+S is initialized and the slave might receive a telegram from a class 2

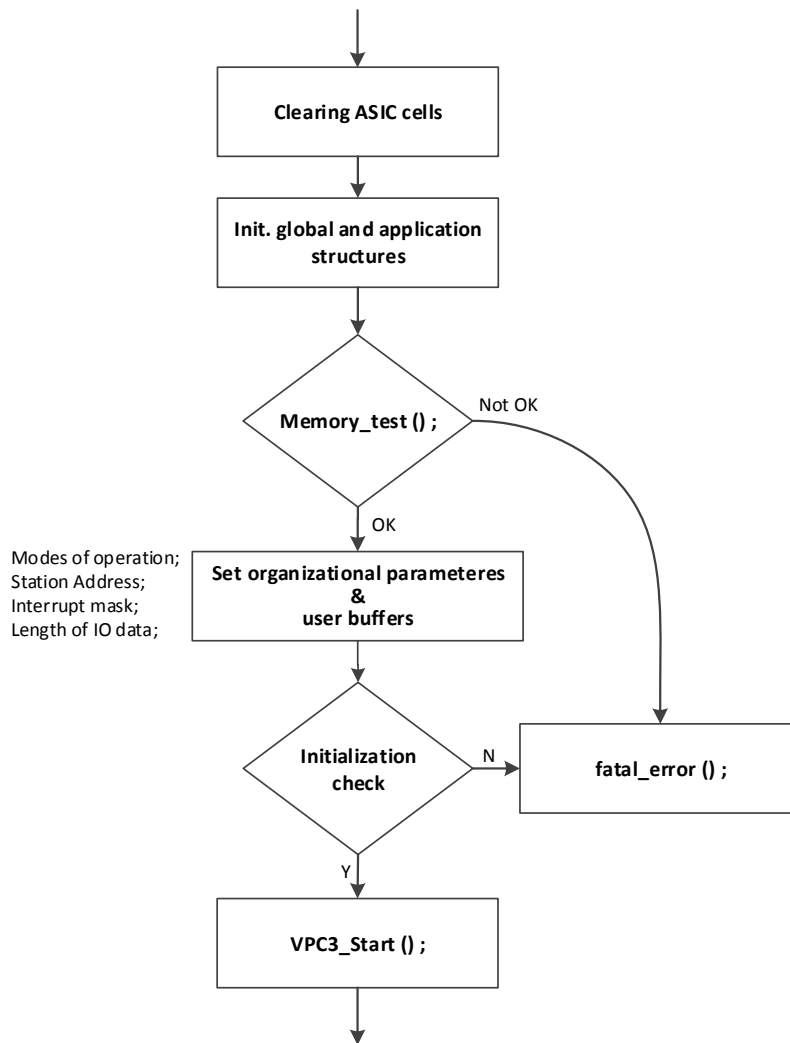


Figure 5.3: ASIC VPC3+S initialization.

master to change its station address. If the slave has already a valid address, the communication will proceed to the Wait Prm state.

2. **Parametrization state:** In this state, the slave waits for a parametrization message from the master, which sets the Profibus modes to operate in. The controller will reject all other telegrams, except a request telegram for diagnostics or configurations. After these parameters have been set, the controller will start with the IO configuration.
3. **Configuration state:** The controller expects to receive a configuration telegram which specifies the number of input and output bytes that are going to be exchanged. Besides the configuration itself, the telegram also causes the slave to check the configuration that was sent against the stored configuration. The slave in this state will accept a request telegram for diagnostics, configuration or a set parameters telegram.

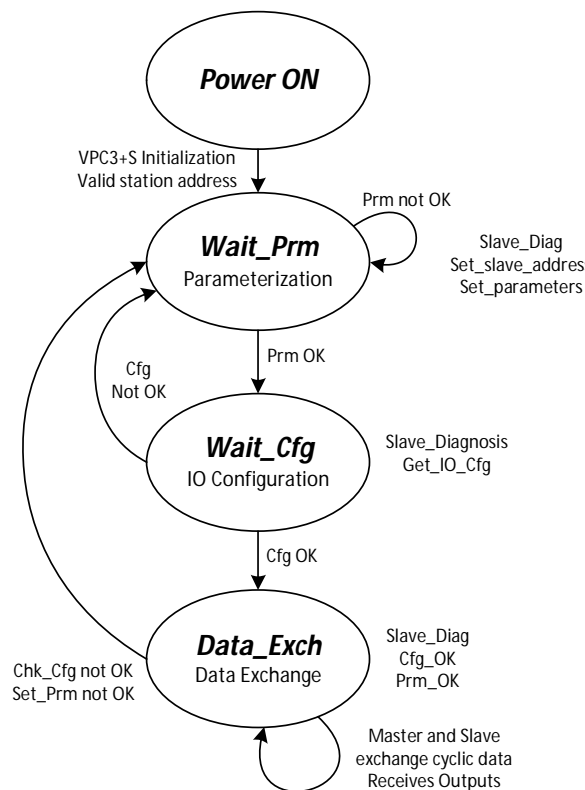


Figure 5.4: The state machine of Profibus-DP slave [3].

4. **Data exchange state:** After all the parametrizations and configurations have been accomplished, the slave cyclically exchanges IO data with the master. In data exchange state, the master communicate with the slave going through another state machine The data processing state (Figure 5.5) is the key state of whole state machine, passing most of the time in this state.

### 5.1.5 Data Exchange State Machine

The flowchart of the data exchange state machine is shown in Figure 5.5. Once the parametrization and IO configuration of the Profibus-DP slave have been accomplished, the controller enters in the data exchange state. In this state the microcontroller keeps checking if there is new parameter or configuration data, polling all the interrupt events in the VPC3+S interrupt register. The microcontroller does the corresponding process and notifies the ASIC that the interrupt event has been acknowledge through the interrupt acknowledge register. The microcontroller processes/sends the input data (Figure 5.6(a)) and receives the output data (Figure 5.6(b)) from the PLC master. The watchdog timer is then re-triggered and the the process starts again.

The Profibus slave may assume a fail-safe state if its watchdog timer expires without having received a message from its assigned master. Normally, this timer is triggered

whenever the master talks to the slave. If a timer overflow has occurred, this means that the master has not communicated with the slave recently, and thus is not being controlled. The slave will then leave the data exchange state and its outputs will go to a default state. This state is usually set via user-defined parameters of the parametrization telegram.

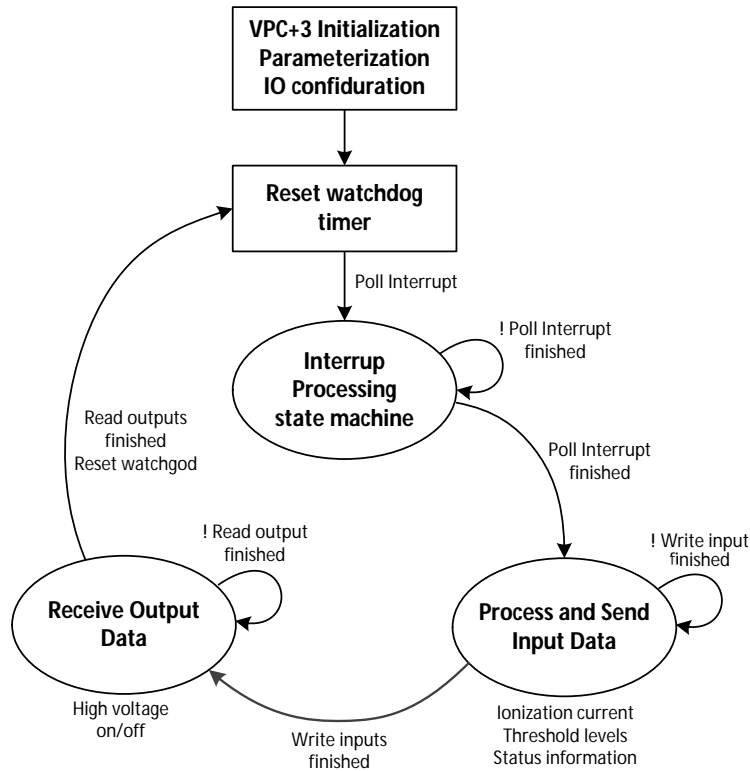


Figure 5.5: Data exchange state.

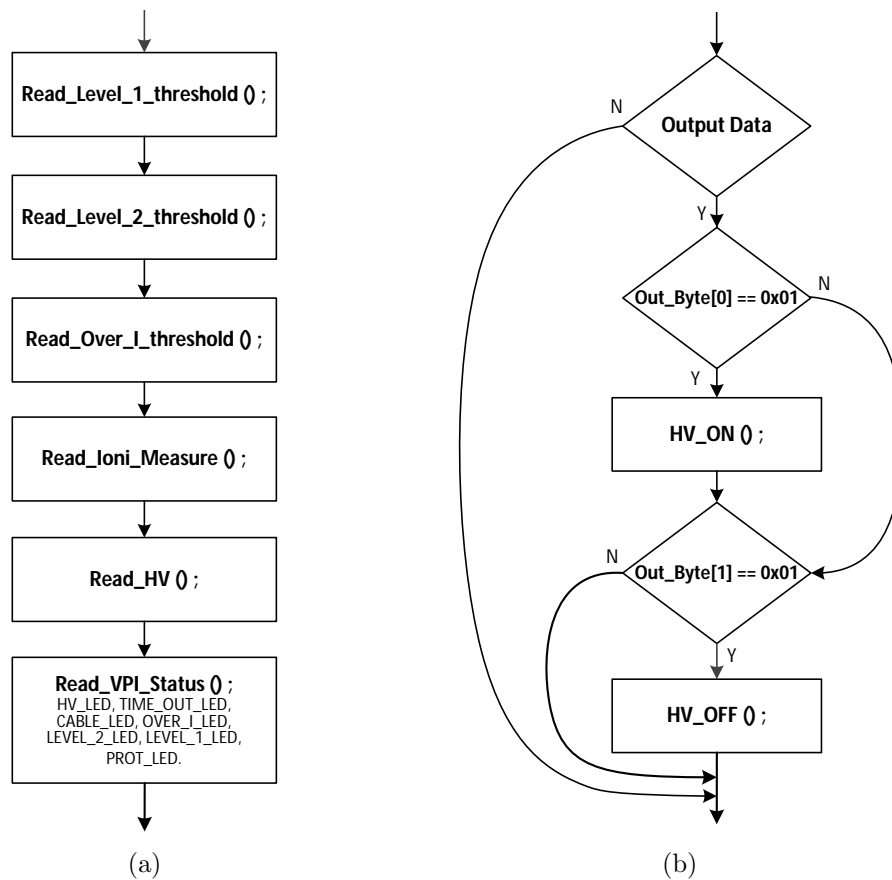


Figure 5.6: Data processing (a) read and process input data; (b) receive output data.

## 5.2 SIMATIC STEP 7 Interface

In the vacuum system at CERN, the PLC controllers are provided by Siemens. Every equipment connected to the Profibus network is therefore controlled by Simatic STEP 7 software [21]. To test and evaluate the correct behaviour of the PROFIS designed board, a hardware configuration on Simatic was set up and data was exchanged between Profibus master and slave.

### 5.2.1 Hardware Configuration

To test the Profibus communication, a PLC master CPU314C-2 [22] was used. This CPU contains a microprocessor that achieves a processing time of 60 ns per binary instruction. The extensive memory (max. 8 MB) allows the project to be stored in the CPU memory, and keeping it every time the CPU is turned-off. A multi-port interface (MPI) is used to establish the communication with a personal computer (PC). The MPI makes it possible to set up a simple network with a maximum of 16 CPUs. A Profibus-DP interface is also

included in this CPU. It allows a distributed automation configuration offering high speed and easy of use. An integrated 24 digital inputs, 16 digital outputs, 4 analog inputs and 2 analog outputs are also provided in this CPU, although they are not used. The hardware and the network configurations are depicted in Figures 5.7 and 5.8, respectively.

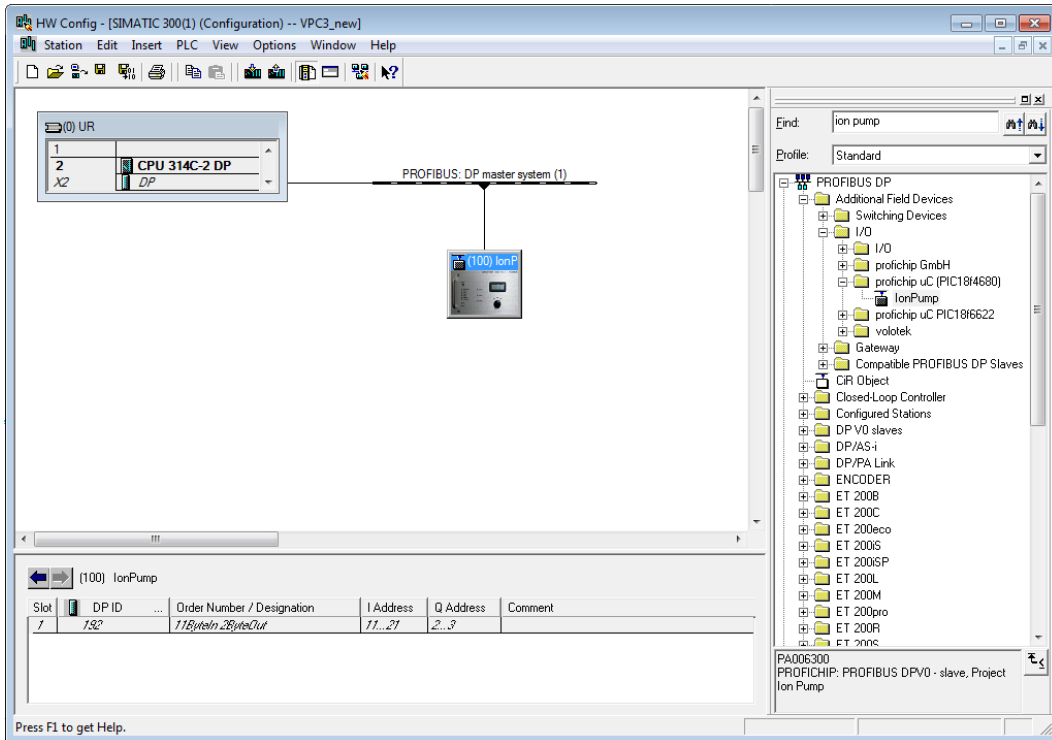


Figure 5.7: Simatic STEP7 hardware configuration.

As can be seen in Figure 5.7, the used CPU contains a DP interface which connects the VPI controller to the Profibus network. The Profibus slave (VPI controller) is configured with 100 decimal out of 127. Figure 5.8 illustrates the network configuration of the designed application. A MPI bus interface connects the CPU (Profibus master) to a PC, where Simatic STEP7 is installed. A Profibus-DP network connects the Profibus master to the Profibus slave.

## 5.2.2 GSD file

General Station Description (GSD) files contain the information about the basic capabilities of the device. All the Profibus devices are shipped with a GSD file that includes informations about the length of the input/output data, the modes of operation and the available diagnostics. Every Profibus slave is therefore described through its specific file, which is uploaded into the Simatic STEP 7 software. For this project, a specific GSD file was written and is illustrated in Figure 5.9.

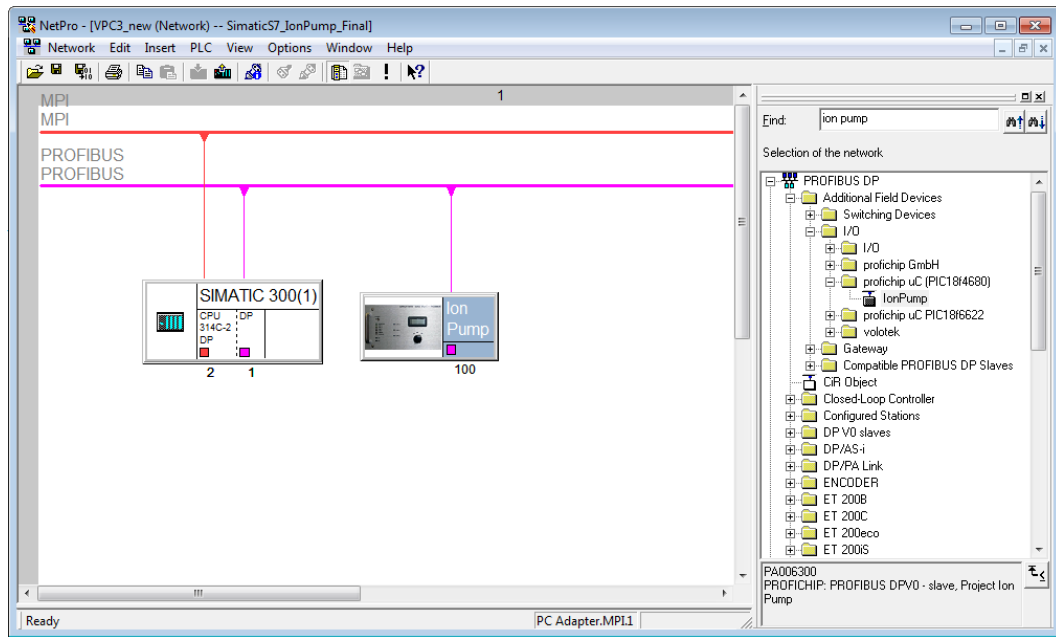


Figure 5.8: Simatic STEP7 network configuration.

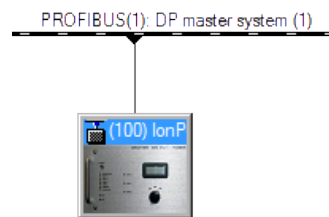


Figure 5.9: GSD symbol of the VPI controller.

### 5.2.3 Simatic STEP7 I/O Data

As mentioned in the previous chapter, this system has 11 input bytes (VPI controller to Profibus master) and 2 output bytes (Profibus master to VPI controller). As the microcontroller's ADC is 10-bit word and the data telegrams sent to Profibus are 8-bit word, to represent an ADC output value 2 bytes are required. Therefore, each VPI threshold level is represented with two byte of data. Table 5.5 shows the equivalent data in the Simatic STEP 7 software.

Table 5.5: Simatic STE7 IO data.

Length	STEP7 Signal	Description
2 Byte	LEVEL 1 THRESHOLD	Level 1 set potentiometer [DB7;DB0]
		Level 1 set potentiometer [DB15;DB8]
2 Byte	LEVEL 2 THRESHOLD	Level 2 set potentiometer [DB7;DB0]
		Level 2 set potentiometer [D15;DB8]
2 Byte	OVER CURRENT THRESHOLD	Over I set potentiometer [DB7;DB0]
		Over I set potentiometer [D15;DB8]
2 Byte	IONIZATION MEASURE	Ionization current reading [DB7;DB0]
		Ionization current reading [D15;DB8]
2 Byte	HV READING	High-voltage reading [DB7;DB0]
		High-voltage reading [DB15;DB8]
1 Byte	PROTECTION	VPI Controller status information
	LEVEL 1	
	LEVEL 2	
	CABLE	
	OVER I	
	TIME OUT	
	HV	
not used		

## 5.3 LabVIEW Interface

LabVIEW is a graphical programming platform offering full integration with existing software, intellectual property (IP) cores and hardware. Besides the PLC master interface, a LabVIEW interface using a serial port connection to communicate with the PROFIS was also developed. The microcontroller's USART module is used to communicate with LabVIEW, presenting the VPI levels and also the status information. To control the high-voltage power supply, two press buttons were added to this application.

### 5.3.1 State Machine

This LabVIEW application runs according to the state machine presented in Figure 5.10. It contains two main states: Idle and Refresh. In Idle state (Figure 5.11(a)), the high-voltage buttons (switch on/off) and the stop button are pooled. If no button is pressed, the state machine by default continues to Refresh state. If one of the high-voltage buttons is pressed, the state machine enters in a new state handled by a CASE structure. In case of the stop button is pressed, Exit Application is the next state, where the application is stopped. In Refresh state (Figure 5.11(b)), the data from the VPI controller are received through the serial port and decoded by a string to array block. The elements of the

generated array (digital and analogue data) are the inputs of the FOR loop. Here, the data are presented in the numeric fields and LEDs of the LabVIEW application front panel.

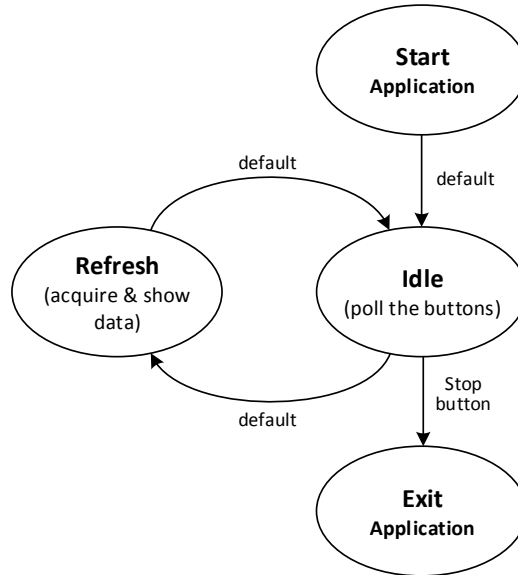


Figure 5.10: LabVIEW application state machine.

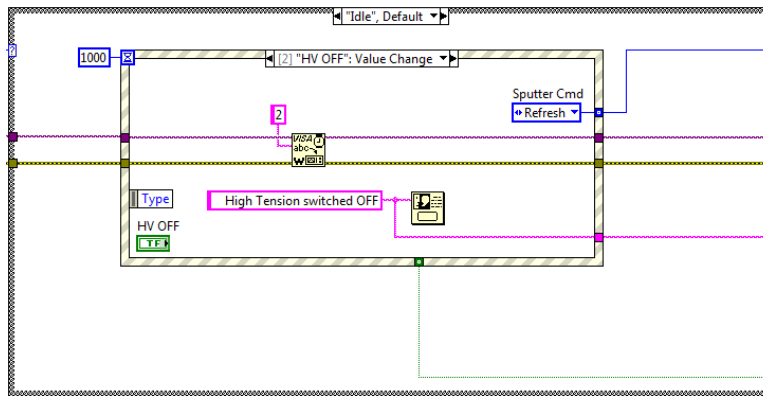
### 5.3.2 Case Structure

In order to turn-on and turn-off the high voltage, two press buttons were added. Ideally they only actuate on the high-voltage supply when one of them is pressed. Therefore, an interrupt system had to be used. A CASE structure located inside of the Idle state implements three possibilities, "HV\_ON", "HV\_OFF" and "Exit\_Application". The default case is the absence of interrupts on it, meaning that if no buttons are pressed it executes the Refresh state, continuously.

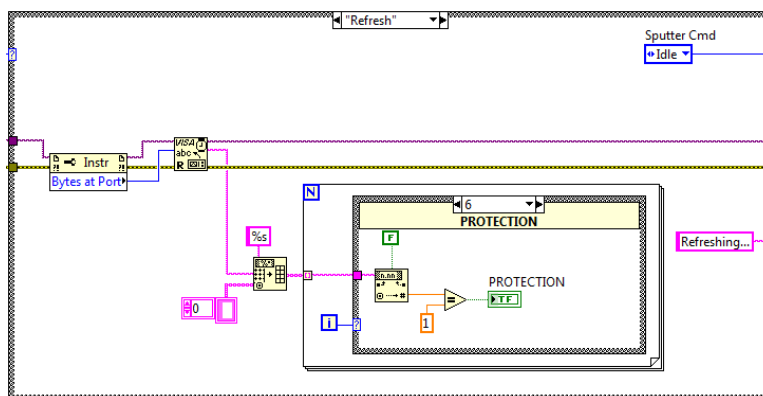
Each case is triggered whenever a button is pressed, executing what is programmed for: turning on/off the high-voltage and stop the LabVIEW application. Figure 5.12 shows the four options of the CASE structure.

### 5.3.3 FOR loop

The FOR loop runs inside the Refresh state and organizes the VPI values and the status information on the LabVIEW application panel. A "String to Array" block decodes the received stream into an array with 9 elements (FOR elements). The decoded array elements are, in each FOR cycle, presented in the different numeric fields and LEDs. Figure 5.13 presents the FOR cycles 1, 2, 5 and 6.



(a)



(b)

Figure 5.11: LabVIEW state machine (a) Idle state; (b) Refresh state.

### 5.3.4 Application Panel

The VPI status information are presented through LEDs, on the left side, indicating whether the threshold levels were reached or not, as well as other information. The programmed levels, which are set through potentiometers in the VPI's controller front panel, are shown at the center of the LabVIEW application panel. On the right side, there is a high-voltage LED indicator informing its status. Finally, the ionization current measurement (output voltage form the VPI controller) is shown on the panel next to the high-voltage turn-on and off press buttons. Figure 5.14 depicts the developed LabVIEW application panel.

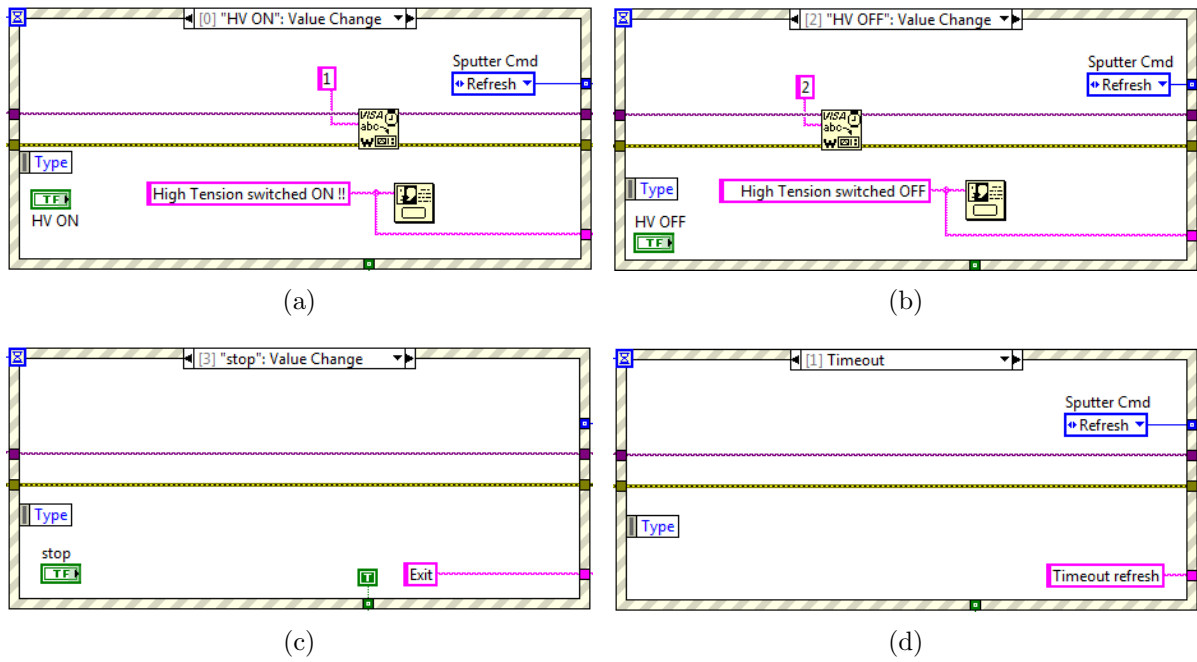


Figure 5.12: LabVIEW CASE structures (a) HV on button; (b) HV off button; (c) stop application button; (d) time out.

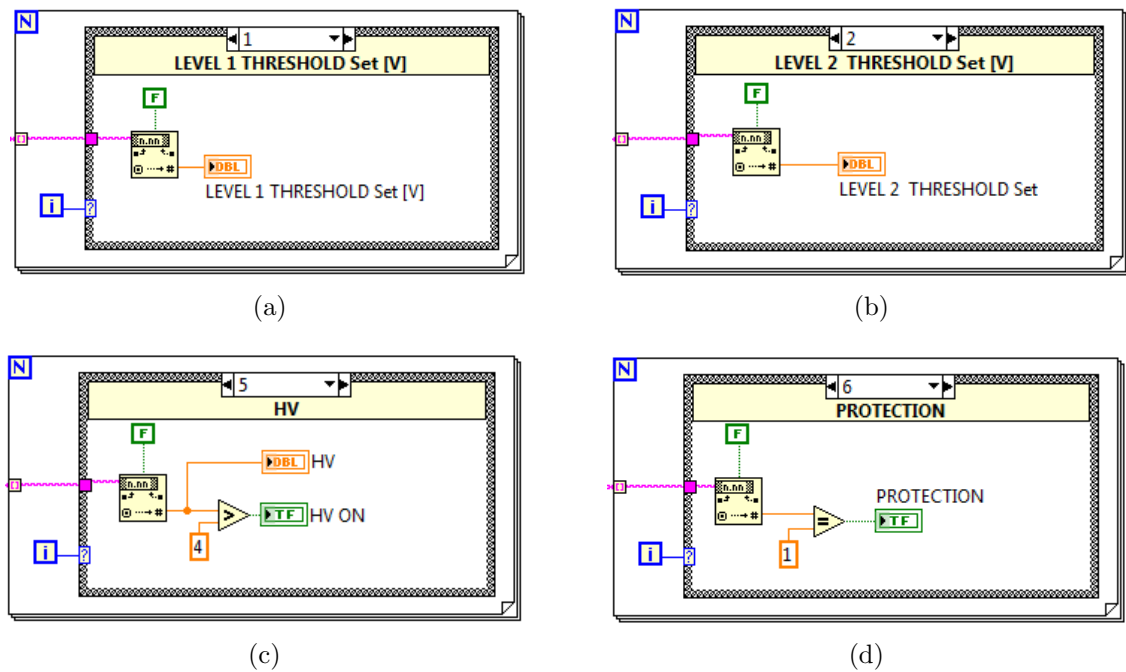
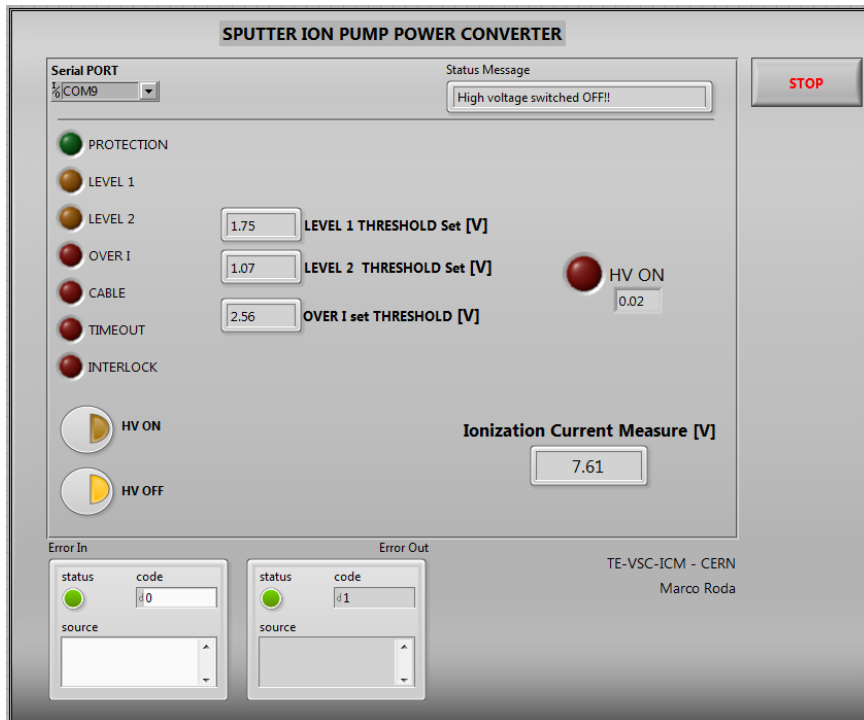
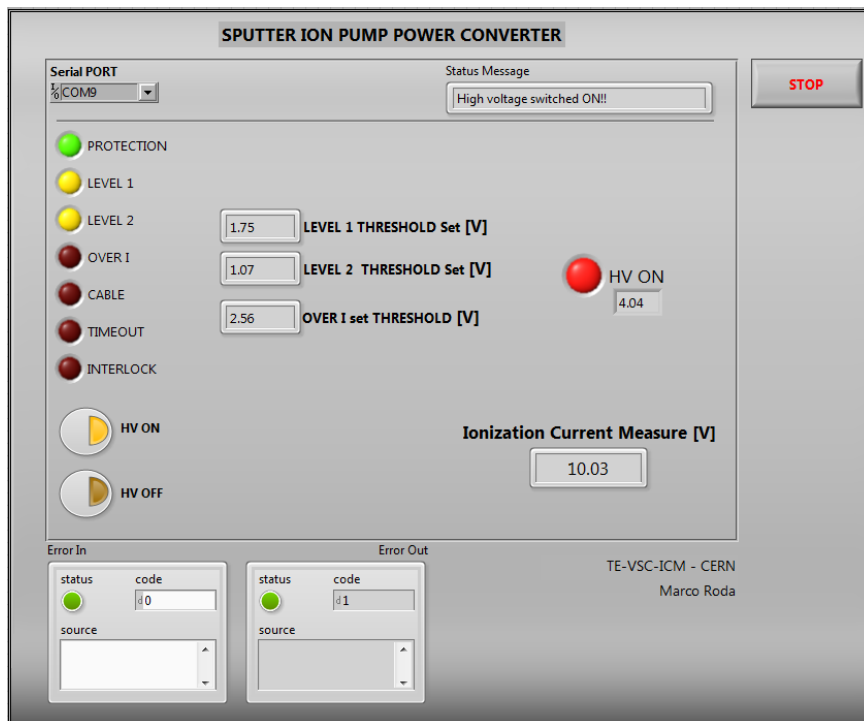


Figure 5.13: Example of FOR cycles.



(a)



(b)

Figure 5.14: LabVIEW application panel (a) high-voltage off and (b) high-voltage on.

# Chapter 6

## Conclusion and Future Work

### 6.1 Conclusions

The aim of this master internship was to work on two different projects: sector valves control units and sputter ion pump controllers. To accomplish these goals, the work was split into two parts with several stages.

The first part of the internship (October 2013 - March 2014) focused on studying the existing control architecture of the sector valves of the LHC, SPS and CPS and upgrade its control firmware. The involved interlocks and their function on the protection system were identified and the control logic was understood. A new control firmware based on VHDL was developed and digital simulations using Modelsim were performed. The control logic used in SPS was also used in CPS. The new developed firmware for CPS was upgraded and only the needed logic was kept. This problem seems to be solved since no events were reported, so far. Furthermore, three technical reports were written documenting the existing system. Since no documentation was found in any repository, the production of such documents was a good achievement for CERN.

In the second part of the internship (April 2014 - November 2014) an electronic card for sputter ion pumps was designed and developed. To accomplish this goal, several stages were defined and each of them has required a significant amount of time to obtain optimal results. In the first stage, the electronic cards of the existing VPI controller were studied. All the VPI signals, as well as their functionalities had to be well assimilated in order to design the new Profibus-DP controller. The IO circuit of the PROFIS was designed in this stage.

With the hardware of the IO module defined, a Microchip microcontroller was chosen as controller and data acquisition through its internal 10-bit ADC. It allowed to interface the ASIC with only four wires, instead of using a parallel bus. An external crystal is used

and, at the maximum, the microcontroller can run up to 40 MHz when the internal PLL is enabled.

For the Profibus-DP ASIC, a market survey was conducted. The VPC3+S from Profichip was chosen. This ASIC allows to be interfaced through a serial protocol. This ASIC contains a dual port 4k RAM that can be interfaced using SPI, I2C or parallel bus. The microcontroller's SPI module was used to communicate with the ASIC.

The first PROFIS prototype was developed using 2-Layer PCB. Corrections and optimizations were evaluated and the final prototype was produced using a 4-layer PCB, two layers for routing signals and two internal layers for power supply distribution. The final prototype performed as expected.

The performance of the PROFIS exceeds the initial expectations both in terms of the data acquisition accuracy, and communication with the PLC master and LabVIEW. A specific GSD file for the new VPI controller was created in order to identify this device on the Profibus-DP network.

Summarizing, all the initial goals of the master internship were accomplished.

## 6.2 Future Work

The work presented in this report presents a good background that might be used in future projects. The sector valves control unit is interfaced, as stated in this report, with 23 digital IOs using a specific protocol. Since the Profibus-DP protocol is implemented on the PROFIS, the microcontroller can be replaced by another one with more IOs and a new board may be designed suited for these specifications.

Regarding the PROFIS itself, instead of using a plastic enclosure, metallic supports can be designed and added to the PCB in order to fix it in the VPI controller chassis.

Although three prototypes have been developed and tested at the laboratory using a sputter ion pump controller, ensuring the correct behavior of the new PROFIS, a new test must be performed with the VPI controller on-site connected to the underground Profibus network.

# Bibliography

- [1] “European Organization for Nuclear Research, CERN.” [Online]. Available: <http://home.web.cern.ch/about/accelerators>
- [2] P. Gomes, F. Antoniotti, S. Blanchard, M. Boccioli, G. Girardot, H. Vestergard, L. Kopylov, and M. Mikheev, “THE CONTROL SYSTEM OF CERN ACCELERATORS VACUUM [ CURRENT STATUS & RECENT IMPROVEMENTS ],” 2011.
- [3] M. Felser, “PROFIBUS Manual,” Jan. 2003. [Online]. Available: <http://profibus.felser.ch/en/>
- [4] I. Acromag, “INTRODUCTION TO PROFIBUS DP,” 2002. [Online]. Available: <http://www.diit.unict.it/users/scava/dispense/II/Profibus.pdf>
- [5] “High-efficiency communication subsystem for small data packages – Vol. 3, Profibus DP,” 1998.
- [6] “Digital data communications for measurement and control – fieldbus for use in industrial control systems – part 5: Application layer service definition, communication model type 3 specification,” 2000.
- [7] V. Oerlikon Leybold, “Fundamentals of Vacuum Technology,” Jun. 2007. [Online]. Available: [https://www3.nd.edu/~nsl/Lectures/urls/LEYBOLD\\_FUNDAMENTALS.pdf](https://www3.nd.edu/~nsl/Lectures/urls/LEYBOLD_FUNDAMENTALS.pdf)
- [8] “CERN EDMS document.” [Online]. Available: <https://edms.cern.ch/edmsui/#!/master/navigator/document?D:1594851923:1594851923:subDocs>
- [9] “EPM7064slc84-10n Data Sheet.” [Online]. Available: [https://www.altera.com/en\\_US/pdfs/literature/ds/m7000.pdf](https://www.altera.com/en_US/pdfs/literature/ds/m7000.pdf)
- [10] “ADM2486 Data Sheet.” [Online]. Available: [http://www.analog.com/static/imported-files/data\\_sheets/ADM2486.pdf](http://www.analog.com/static/imported-files/data_sheets/ADM2486.pdf)
- [11] “NTE0305 Data Sheet.” [Online]. Available: [http://www.mjdic.com/Attachments/201408/20140825\\_111909\\_131.pdf](http://www.mjdic.com/Attachments/201408/20140825_111909_131.pdf)

- 
- [12] “LP3985 Data Sheet.” [Online]. Available: <http://www.ti.com/lit/ds/symlink/lp3985.pdf>
- [13] “Siemens ASICs.” [Online]. Available: <http://w3.siemens.com/mcms/distributed-io/en/profibus-components/asics/pages/default.aspx>
- [14] “Profichip ASICs.” [Online]. Available: <http://www.profichip.com/products/overviewasics/?L=5>
- [15] “VPC3+ User Manual (Revision 1.04),” 2012. [Online]. Available: [www.profichip.com](http://www.profichip.com)
- [16] “PIC18f4680 Data Sheet.” [Online]. Available: <http://ww1.microchip.com/downloads/en/DeviceDoc/39625c.pdf>
- [17] “FT232rl Data Sheet.” [Online]. Available: [http://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS\\_FT232R.pdf](http://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT232R.pdf)
- [18] “ADP7105 Data Sheet.” [Online]. Available: [http://www.analog.com/static/imported-files/data\\_sheets/ADP7105.pdf](http://www.analog.com/static/imported-files/data_sheets/ADP7105.pdf)
- [19] “LM7805 Data Sheet.” [Online]. Available: <https://www.sparkfun.com/datasheets/Components/LM7805.pdf>
- [20] “ADP1706 Data Sheet.” [Online]. Available: [http://www.analog.com/media/en/technical-documentation/data-sheets/ADP1706\\_1707\\_1708.pdf](http://www.analog.com/media/en/technical-documentation/data-sheets/ADP1706_1707_1708.pdf)
- [21] “SIMATIC net PROFIBUS networks manual,” 1999.
- [22] “SIMATIC S7-300, CPU 314c-2pn/DP COMPACT,” 2015. [Online]. Available: <https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6ES7314-6EH04-0AB0>

# Appendix A

## PROFIS - Price Estimation

### A.1 CERN

In total, three prototypes were produced at CERN (PCB+parts+assembly). The unit price per PCB was 295.79 CHF, as shown in Table A.1.

Table A.1: PROFIS price estimation (Unit size: 71.86x145.29mm)

Specification	Quantity	Unit price (PCB+Parts+Assembly)	Total amount
Board type: 4 Layer rigid PCB Board thick: 1.60mm	6	295.79 CHF	1,774.73 CHF
Copper thickness:1OZ Green solder mask	15	251.49 CHF	3,772.35 CHF

### A.2 External Company

A price estimation was ordered to Shenzhen Sienta Industry Co.,Ltd to produce 1, 6, 15, 30 or 150 PCBs. The price is presented in Table A.2.

Table A.2: PROFIS price estimation (Unit size: 71.86x145.29mm)

Specification	Quantity	Unit price (PCB+Parts+Assembly)	Tooling	Total amount
Board type: 4 Layer rigid	1	450 USD	334 USD	784 USD
PCB Board thick: 1.60mm	6	177.5 USD	200 USD	1,534 USD
Copper thickness:1OZ	15	170 USD	180 USD	3,034 USD
Green solder mask	30	168.5 USD	175 USD	5,584 USD
Surface treatment:L/F HAL	150	150 USD	154 USD	23,434 USD



# Appendix B

## PROFIS Schematics



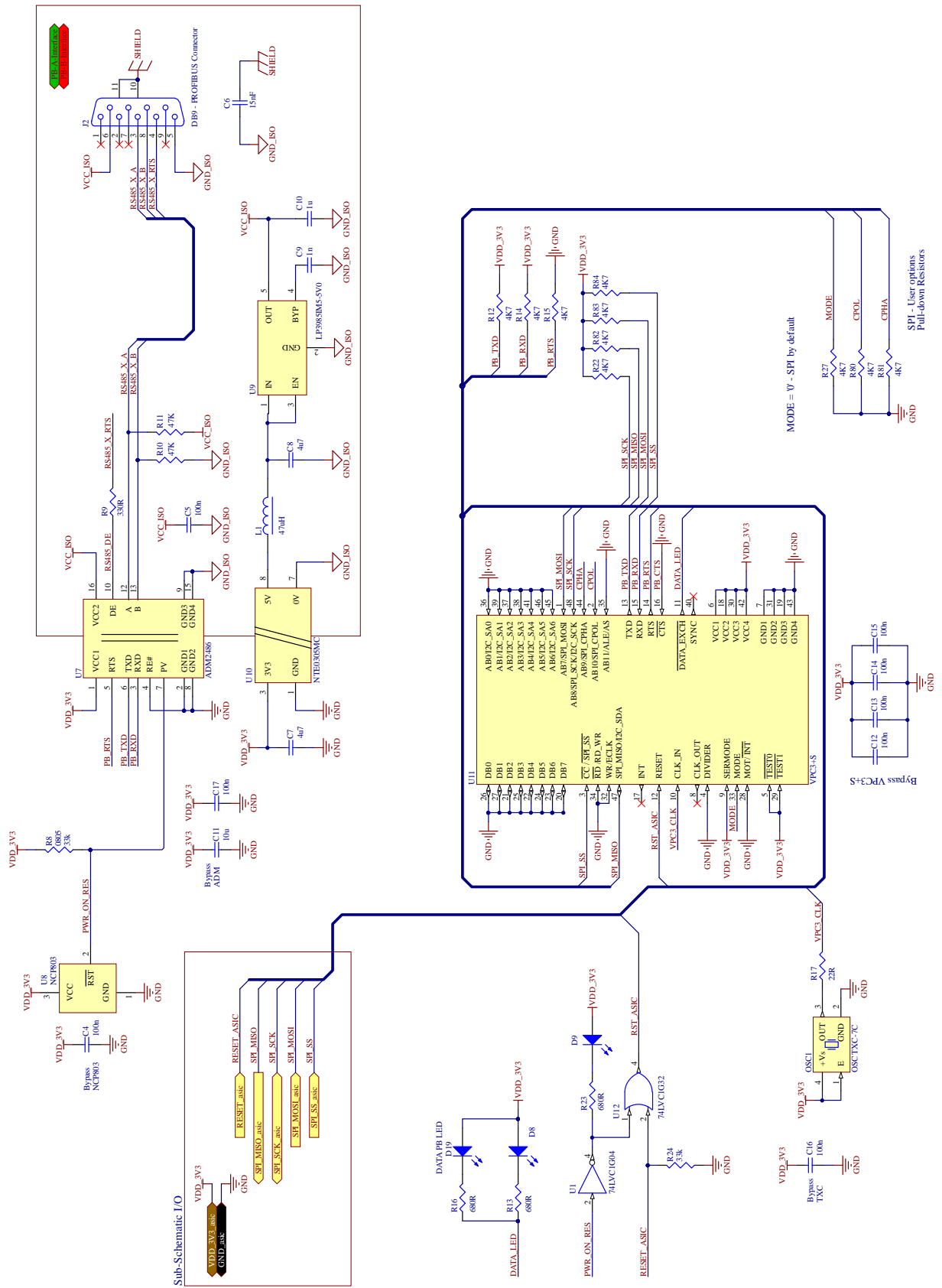


Figure B.2: Profibus-DP interface module.

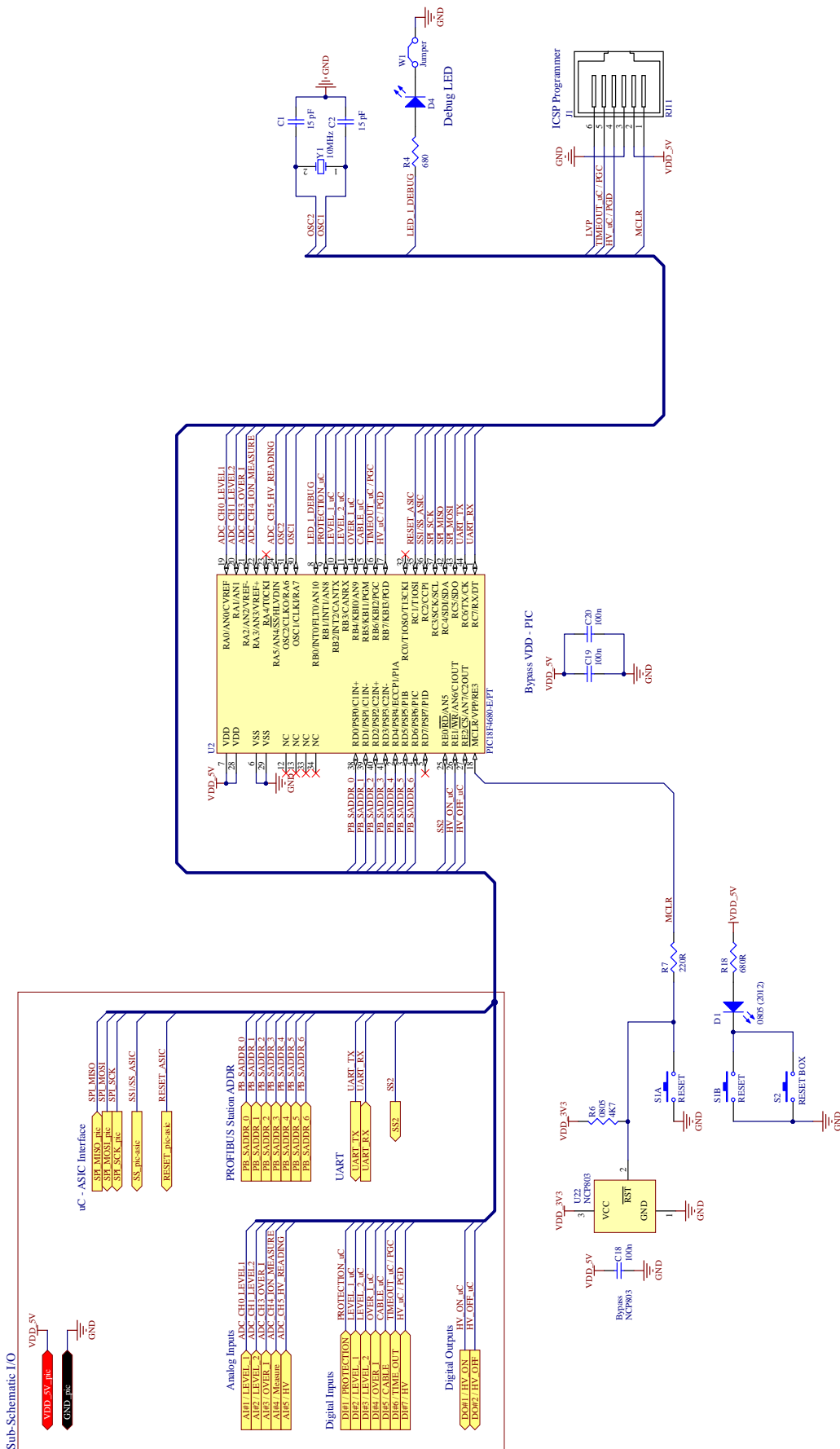


Figure B.3: Microcontroller module schematic.

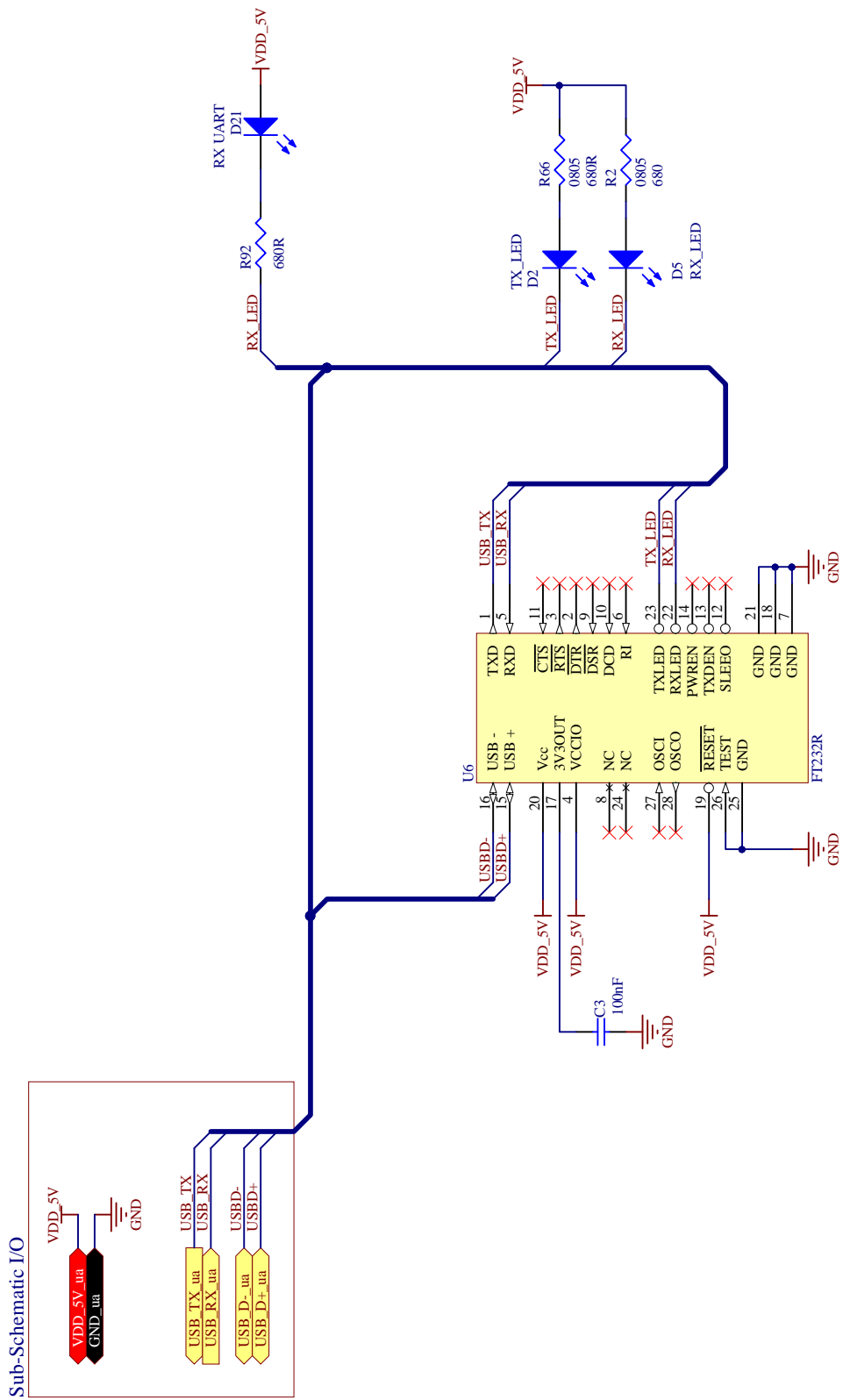


Figure B.4: UART module schematic.

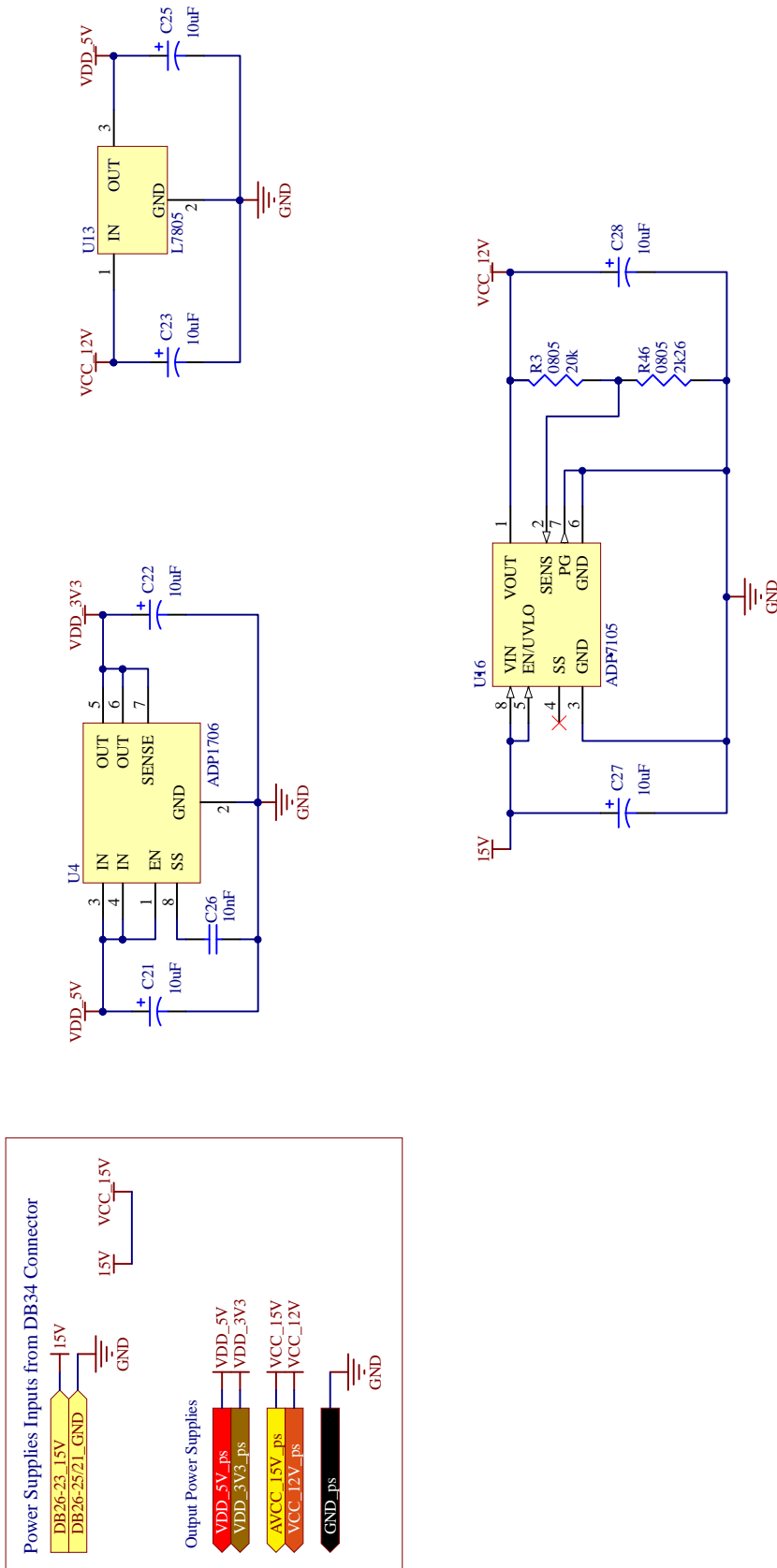


Figure B.5: Power-supply module schematic.





# Appendix C

## Designed PCB

### C.1 First Prototype

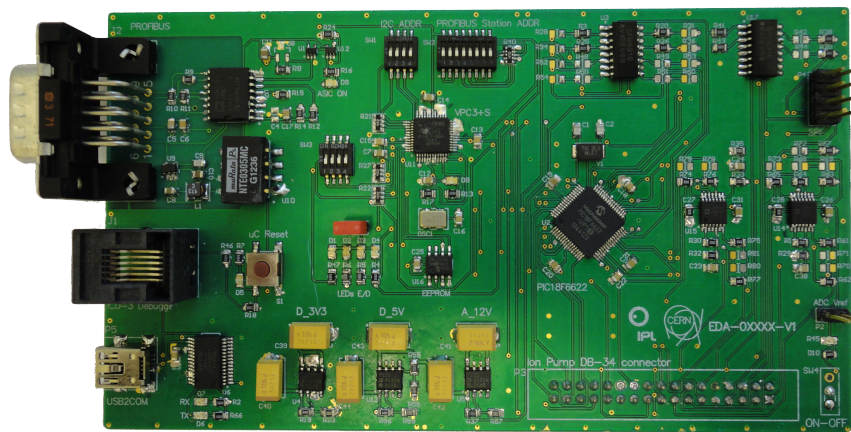


Figure C.1: First PCB prototype.

### C.2 Final Prototype

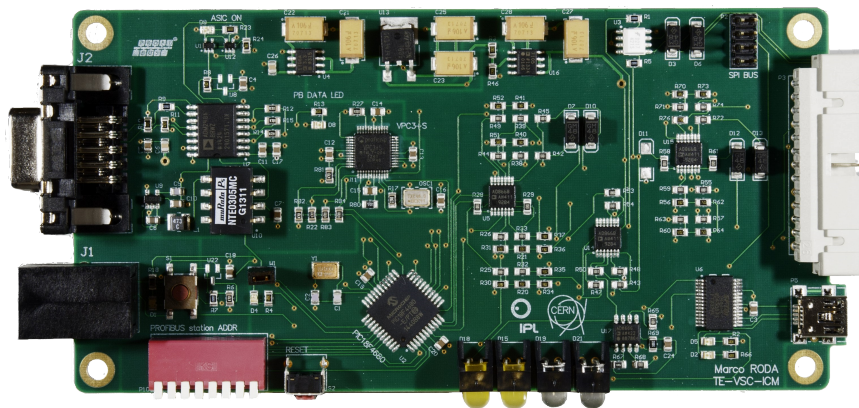


Figure C.2: Final PCB prototype.

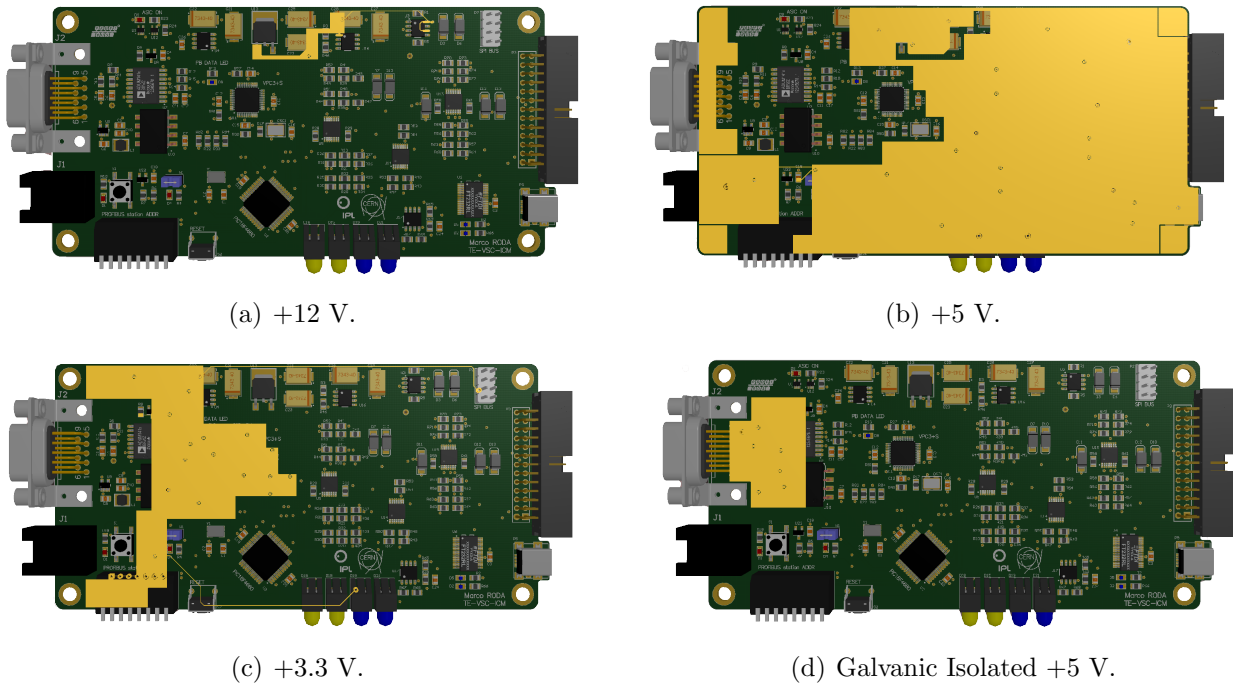


Figure C.3: Power planes.

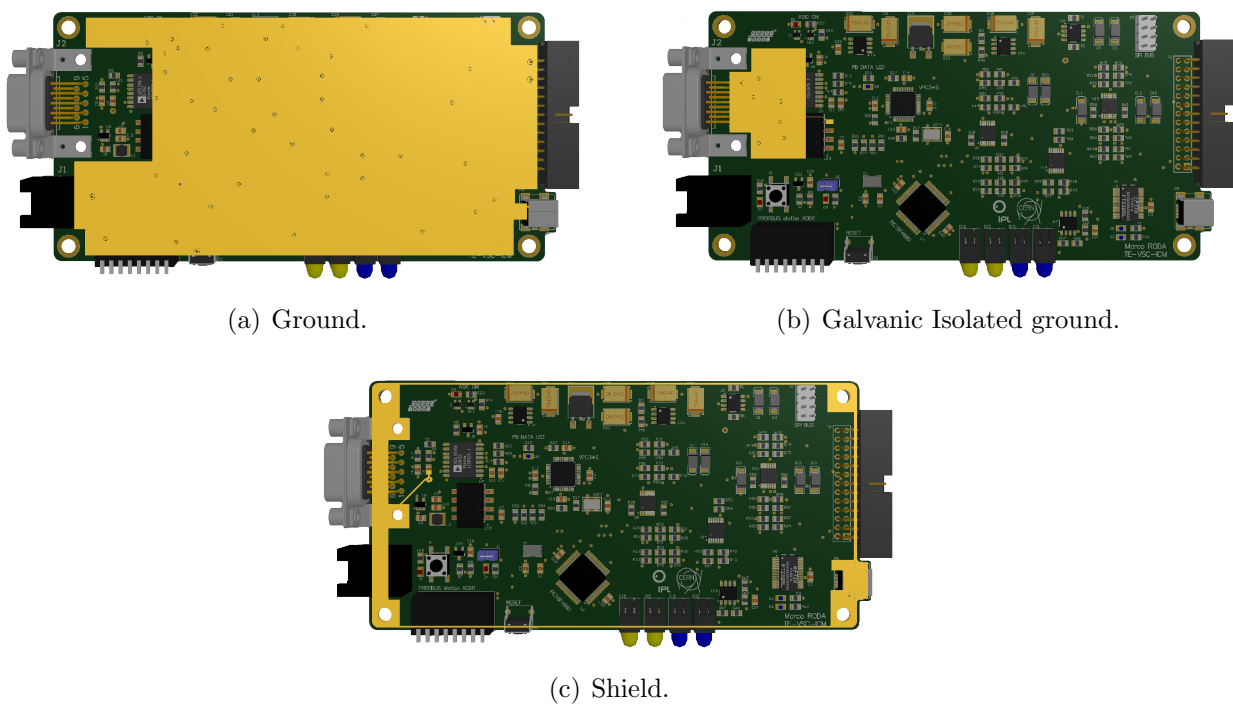


Figure C.4: Ground planes.

