UNIVERSITÀ DEGLI STUDI ROMA TRE

CORSO DI DOTTORATO DI RICERCA IN ELETTRONICA APPLICATA

XXX CICLO

Innovative Techniques for Photovoltaic Panels Modeling and Monitoring

Stefano Gaiotto

Professori guida

prof. Francesco Riganti Fulginei

prof. Alessandro Salvini

Coordinatore del Dottorato:

prof. Enrico Silva

Contents

Introduction

In the last years, the photovoltaic (PV) energy market had an extremely rapid expansion and this brought to the request of effective devices and tools for the estimation of the electrical power produced by PV plants. Nevertheless, often the estimation of energy production coming from PV plants is not a simple task. Indeed, it depends both on weather and environmental factors (e.g. temperature, irradiance, sky conditions, etc...) usually estimated through statistical data and on the goodness of the PV model utilized for the Current-Voltage $I-V$ outputs prediction under real operating conditions (temperature and irradiance). Manufacturers of solar PV modules provide their $I-V$ characteristics for allowing to estimate power conversion yield for either solar home or grid systems or for research activities. These characteristic curves are obtained in the laboratory under simulated and controlled test at Standard Reference Conditions (SRC) of light intensity (G =1000 W/m2) and temperature (T = 25°C). Unfortunately, the conditions under which these curves are found are usually different from those obtained under varying natural atmospheric conditions in which the PV modules work. Several sophisticated devices are commercially available on the market but are very expensive. In this PhD thesis work an accurate $I-V$ tracer based on micro-

computer assisted DC-DC converter used as electronic load is described. A preliminary version of this instruments has been presented in [Gaiotto et al., 2015].

Actually, the described instrument, is much more than just a simple tracer. Many works, like those in [Leite et al., 2012, Ibirriaga et al., 2010, Erkaya et al., 2014, Rivai and Rahim, 2014, Papageorgas et al., 2015, Spertino et al., 2015, Kuai and Yuvarajan, 2006, Leite and Chenlo, 2010], provide more or less sophisticated equipments for tracing the $I - V$ curves of a photovoltaic panel. The equipment proposed here is much more complex and accurate than these, being able to trace PV panels with a maximum power up to 300W with a very high accuracy, moreover, since it is controlled by means a powerful control unit, complex algorithms can be executed.

In the first part of this work the hardware of the instrument is described, after that, it is shown how the identification algorithms described in [Laudani et al., 2014a] can be easily be embedded into the system. These algorithms provide the five parameters of the "one-diode" model of a PV panel, but other characterizations are possible, allowing researchers and engineers to evaluate new PV panels and new models to describe them. In this work it is also shown how other devices can be characterized like power supplier or battery.

Furthermore, since the proposed system is able to collect voltage and current values both in input and in output of the DC-DC converter, it can be used for the development of DC-DC converter control algorithms. In [Gaiotto et al., 2016] it is described how the proposed system can be used for the fast realization of a solar battery charger. It is shown how the system can

be used to characterize PV panels, batteries, both during the charge and during the discharge phase, and, moreover, how it is possible to evaluate the performance of a fast Neural Network (NN) Maximum Power Point Tracking (MPPT) algorithm of a PV panel.

For all these capabilities, rather than referring to the instrument like a simple tracer, we prefer to refer to it like a system for the development of new techniques related to the PV panels.

A lot of technical problems have been solved during the system realization: among these, those related to the presence of a powerful DC-DC boost converter near to an accurate measurement unit have been the most tricky ones. The research activity about this item led to the development of a new switched capacitor DC-DC converter that introduce a new concept of EMI noise active filtering. This work has been published by Wiley in [Gaiotto et al.,]

Both PV characterization and DC-DC converter control algorithm verification, are topics about maximization of the power a PV panel can deliver, but, in the real world, this is not the only issue to take into the account. Ageing of PV panel can limit the overall return of investment (ROI) for a PV panel much more than its power delivery maximization. In the second part of this PhD thesis work it is described an innovative technique that allow to monitor the PV panel ageing in real time, that is while the panel is working without the need to disconnect it from the load, simply collecting its working conditions. This innovative technique has been accepted for publication by Elsevier in the prestigious journal "Solar Energy" and will be published soon

([Faba et al., 2017]).

Furthermore, the technique introduced in this work to monitor PV panel ageing, is indeed useful also in other applications where an updated PV panel model is required. Two of them are described in this work: model based MPPT algorithms, and irradiance measurement.

Following this last part, conclusions and final remarks on future developments will close this work.

Part I

Photovoltaic panels modeling

Chapter 1

Photovoltaic panels identification

Usually photovoltaic panels are described by means of their characteristic $I-V$ curves. To achieve this task, a variable electronic load is used to reproduce all the working conditions required to plot them. Besides the electronic load, it is necessary a measurement system able to get voltage and current values at each working condition set by the electronic load. Furthermore, a way to collect the measurement data is required as well.

Among these items, however, the electronic load is the most important section. Many methods have been proposed to accomplish this task. The most commonly used technique are variable resistors, capacitive loads, electronic load with MOSFET or BJT transistors, four-quadrant power supply and DC-DC converters, each of them with its pros and cons. In [Duran et al., 2008] these methods are reviewed.

Many works have been presented on PV panels characterization, each one

exploiting some specific characteristic of the electronic load used. For example, in [Benghanem, 2009, Papageorgas et al., 2015, Kuai and Yuvarajan, 2006, Leite and Chenlo, 2010, Mahrane et al., 2010] it is proposed the use of electronic loads implemented with a power MOSFET, in [Ibirriaga et al., 2010, Erkaya et al., 2014, Spertino et al., 2015] the capacitor load is used, whereas in [Lima et al., 2013, Ramaprabha et al., 2014] a DC-DC converter is employed. Less often, like in [Rivai and Rahim, 2014], the resistive load is adopted.

In the system developed during the research project described in this PhD thesis a DC-DC boost converter is used. A DC-DC converter is able to work at a desired fixed current and voltage condition for an indefinite period of time. This characteristic allows to study the behavior of a PV panel when light, temperature or other environmental conditions change. These kind of investigations is not possible with capacitive load or electronic load used with periodic generator like in [Kuai and Yuvarajan, 2006, Leite and Chenlo, 2010, Mahrane et al., 2010]. Sometimes just a qualitative description of the $I-V$ curves is given by the proposed system, like in [Kuai and Yuvarajan, 2006, Leite and Chenlo, 2010, Mahrane et al., 2010]. In these works the $I-V$ curves are collected by means of an oscilloscope and the electronic load is just a MOSFET driven with a periodic signal that makes the panel work continuously sweeping between the open circuit condition and that of short circuit. Some works propose systems able to characterize a PV in field conditions, both to track the maximum power point working condition, and for diagnostic purpose, whereas other systems, like those proposed in [Rivai and Rahim, 2014, Ibirriaga et al., 2010], are intended to be used in laboratory for

research purpose. Some systems, like those in [Ibirriaga et al., 2010, Lima et al., 2013, Rivai and Rahim, 2014, Benghanem, 2009, Papageorgas et al., 2015], have a microcontroller board used to manage the whole system and to communicate with other equipments. In [Durán et al., 2012] a prototype of system where a SEPIC DC-DC converter is used as electronic load. The system is controlled by a low cost controller (PIC 16F877) and data are collected by means of data acquisition card and virtual instrument, based on LabVIEW interface, running on a personal computer.

Chapter 2

A comprehensive development system

In this chapter it is described an instruments that has been developed during the research project described in this PhD thesis work. At beginning it was intended as just a tracer for the $I-V$ curves of PV panels, that is as just an instrument able to collect some points of the curve $I-V$ of a PV panel. During its development, however, it was improved in order to give engineers and scientists the possibility to verify the behavior of DC-DC control algorithms as well. Examples of this algorithms are the Maximum Power Point Tracking (MPPT) ones used to maximize the power delivered by the PV panels to the load, but also any other power scheduling policy, like those involved with battery management, can be verified.

Furthermore, due to the high processing capability of the control unit in the instrument, complex algorithms for the one diode model parameters extraction have been embedded into it, making a simple curves tracer, an instru-

ment able to provide a full characterization of a PV panel. Furthermore, because of its DC-DC control algorithms capability, it could also be referred to as a comprehensive development system for solar energy exploitations by means of PV panels.

2.1 An overview of the instrument

In Fig. 2.1 it is shown the block diagram of the instrument. It is possible to identify three subsystems:

- The variable electronic load
- The signal conditioning and measurement unit
- The control and communication unit

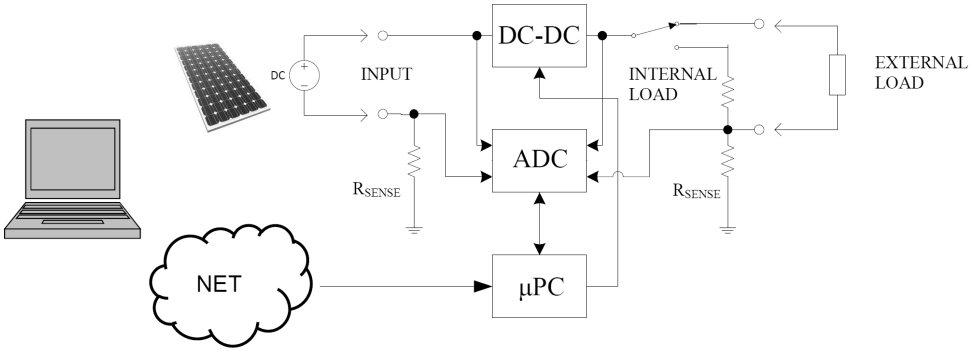


Figure 2.1: The schematic diagram of the development system.

In the block diagram there is a switch (actually an electronic switch) that allows the user to select two different kind of load:

- Internal 300W resistors load
- External load

When the load switch in Fig. 2.1 is set in the 'internal load' position, the output of the DC-DC converter is directed to an internal air-cooled resistors bank where all the power in input is converted into heat. With this configuration, the proposed equipment can be used to trace the PV panels characteristic curves. Since the maximum power that can be dissipated by the instrument is 300W, it is possible to analyze almost all the commercial PV panels available.

After moving the load switch to the 'internal load' position, a suitable SW running in the control unit set the duty cycle D of the DC-DC converter all over the range $[0, ..., 1]$ and take for each point the values I and V for that point. In this way the variable electronic load simulates a PV panel working from the open circuit condition to that one of short circuit.

When it is required to provide a PV panel characterization using a non-resistive load, a suitable external load can be used in place of the internal one. This possibility is usually not available with other instruments.

The internal resistive load should be able to convert into heat all the power generated by the PV panel also for long time period, to let the user to perform other type of characterization, for example to study how the Maximum Power Point (MPP) changes in relation to the temperature or irradiance.

PV panels could be very different, with a wide range of generated power,

from few watts to more than 200W, with current spanning from near zero to more than 10A and voltage up to more than 50V. With these requirements, the electronic load is required to be very versatile and robust.
A DC-DC boost converter controlled by using a PWM signal provides good control capability and is very easy to be implemented. With this approach it is also possible to manage high power modules since the produced power can be dissipated more conveniently than with other approaches, for example those using a single high power MOSFET.

The second subsystem is the signal conditioning and measurement unit. This unit is in charge to provide the voltage and current measurement data. The heart of this unit is the Analog Device 24 bits ADC chip, AD7714Y. This chip has an SPI interface, used for communication with the microcontroller, programmable gain, autocalibration and programmable signal filtering functions.
The signal conditioning circuits amplify or attenuates voltage and currents signals in order to provide to the ADC chip the signals that let to have the most accurate measures. The resulting system is able to manage voltage up to 60V and current up to 10A in input.

The third subsystem is the control and communication unit. This subsystem has at least the following two tasks:

- Control of the electronic load and of the conditioning and measurements unit.
- Communication with external systems requesting the measurement data.

Due to the high processing capability of the board, however, complex data processing is possible as well.

This subsystem has been effectively realized making use of a microcomputer board. The board used in the prototype is the Seeed Studio Beaglebone Green, a cheaper version of the Texas Instruments Beaglebone Black one. This board has a lot of general purpose I/O pins (GPIOs), that allow to control the whole equipment, high resolution PWM controllers, used to drive the electronic load, and SPI ports used to communicate with the ADC chip. In the board there is an AM3358 1GHz ARM® Cortex-A8 compatible processor running at 1 GHz and a complete Linux distribution. The presence of the Linux operating system allows to develop quite complex software in a easy way. At moment the control board is able to provide the five parameters of the one diode model of a PV panel using the procedure described in [Laudani et al., 2014a].

The presence of a lot of communication stacks, allows to handle remote connections using different communication protocols on different media type, like USB, Ethernet, Bluetooth and WiFi. In the prototype it is used a SSH connection over an Ethernet link, but it is easy to develop clients running on PCs or tablets or even smartphones connected with many types of physical media.

The client is expected to provide the graphic output and to perform further analysis on the collected data.

2.2 The DC-DC boost converter

2.2.1 The principle of the electronic load

The heart of the proposed equipment is the variable electronic load. A DC-DC boost converter is exploited, as showed in Fig. 2.3, for this task. In the left side of this diagram a PV panel is represented by means of its one diode model. In place of the common representation as in Fig. 2.2, a modified circuit, where the anti-parallel diode is replaced by an external control current source as proposed in [Tian et al., 2012], is used.
On the right side of the picture, a basic DC-DC boost converter is represented.
In an ideal DC-DC boost converter, working in Continuous Conduction Mode (CCM), the voltage provided in output depends only on duty cycle D of the PWM signal, that drives the MOSFET used as electronic switch, and on the voltage in the input as described by the following [Billings and Morey, 2011]:

$$V_{out} = V_{in}/(1 - D). \tag{2.1}$$

When the output of the converter is closed on a resistor R_{load}, it is easy to show that the current drawn from the source depends only on duty cycle D and on the value of the resistor R_{load}, where the power in output from the switch converter is dissipated:

$$I_{in} = V_{in}/\left[R_{load}(1 - D)^2\right]. \tag{2.2}$$

The one diode model of a PV panel gives the current provided by the

panel as a function of five parameters, I_{irr}, I_0, R_S, R_{SH}, and n, that are typical of each panel:

$$I = I_{irr} - I_0 \left[\exp\left(\frac{q\,(V + IR_S)}{N_s nkT} \right) - 1 \right] - \frac{V + IR_S}{R_{SH}}. \tag{2.3}$$

where N_s is the number of cells in series in a PV panel. When the output of a PV panel is used as source for a DC-DC boost converter, the current provided by panel must satisfy both eq. (2.2) and eq. (2.3). Since the duty cycle is fixed, the only unknown is the voltage V at the output of the PV panel. Unfortunately, this equation needs to be numerically solved.
In Fig. 2.4 there is a code fragment of the C language routine used to solve the equations. Just few iterations are enough to provide a 10^{-6} accuracy.

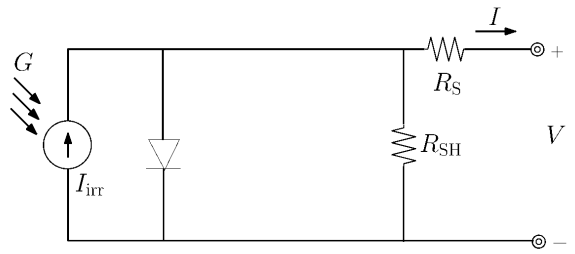


Figure 2.2: One-diode equivalent circuit for a single PV solar cell.

2.2.2 An accurate model of the electronic load

Even if the model provided in the previous chapter is quite good because the efficiency of a switching DC-DC converter is usually high, a better model for the boost converter has been developed. With this model it is possible to estimate exactly the current drawn by the PV panel, the ripple on this

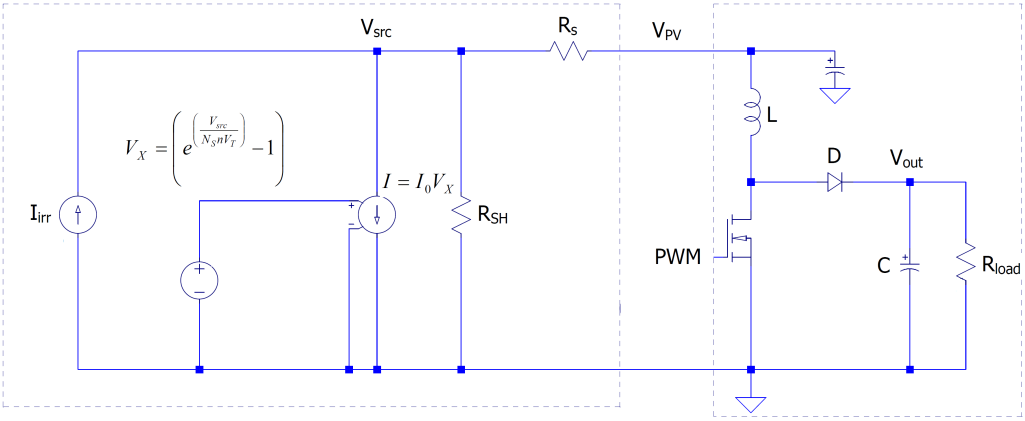


Figure 2.3: DC-DC boost converter used as electronic load for a PV panel.

current, and the power loss on each component of the DC-DC converter

To develop a more accurate model for the electronic load, the first step is to evaluate the current $i_{on}(t)$ when the MOSFET is conducting, that is during the time interval $T_{ON} = [t_0 + kT, t_0 + kT + DT]$, with $k = 0, 1, 2, ...$, where T is the PWM signal period and D is the duty cycle. It is easy to get that $i_{on}(t)$ is the following:

$$i_{on}(t) = i_0 e^{-t/\tau_{on}} + \frac{V_{src}}{R_1}\left(1 - e^{-t/\tau_{on}}\right) \qquad t \in T_{ON} \tag{2.4}$$

In the same way, the current through the inductor when the the MOSFET is off, that is when $t \in T_{OFF} = [t_0+kT+DT, t_0+(k+1)T]$, with $k = 0, 1, 2, ...$, is given by the following:

$$i_{off}(t) = i_0 e^{-t/\tau_{off}} + \frac{V_{src}}{R_2}\left(1 - e^{-t/\tau_{off}}\right) \qquad t \in T_{OFF} \tag{2.5}$$

where $R_1 = R_s + R_L + R_{DS}$, $R_2 = R_s + R_L$, $\tau_{on} = L/R_1$, $\tau_{off} = L/R_2$, and $V = V_{src} - V_\gamma - V_{out}$, R_L is the inductor resistance, V_γ is the threshold

```
if (D<1) {
    i=0;
    niter=0;
    v=v0;

    // Newton's method
    while (niter<100 && error>0.000001) {
        // PV panel current
        ipv = Iirr-I0*(exp(v/(Ns*n*Vt))-1)-v/Rsh;

        // load is an ideal boost converter: R=1/[Rload*(1-D)^2]
        fv = ipv - v/(Rload*pow((1-D),2)+Rs);                    // f(v)
        dfv = (-I0/(Ns*n*Vt))*e-1/Rsh-1/(Rload*pow((1-D),2)+Rs);  // f'(v)

        // solution refinement
        v1 = v - fv/dfv;
        error = fabs(v-v1);
        v=v1;

        niter++;
    }
    if (niter == 100) printf("%f: error = %f\n", __func__, error);
}
```

Figure 2.4: C code for numerical solution with Newton's method.

voltage of the diode, and R_{DS} is static Drain-to-Source on-resistance of the MOSFET.

If I is the mean value of the current $i(t)$, because $i_{on}(t_k + DT) = I + \Delta I$, from eq. (2.4) it is possible to get:

$$\Delta I = \left(\frac{V_{src}}{R_1} - I \right) \frac{\left(1 - e^{-T_{on}/\tau_{on}}\right)}{\left(1 + e^{-T_{on}/\tau_{on}}\right)} \tag{2.6}$$

In the same way, but starting from eq. (2.5), it is possible to write:

$$\Delta I = -\left(\frac{V_{src} - V_\gamma - V_{out}}{R_2} - I \right) \frac{\left(1 - e^{-T_{off}/\tau_{off}}\right)}{\left(1 + e^{-T_{off}/\tau_{off}}\right)} \tag{2.7}$$

Since ΔI given by eq. (2.6) must be the same of that given by eq. (2.7), comparing the two expressions of ΔI it is possible to find:

$$V_{out} = -R_2 \left(\frac{\alpha}{\beta} + 1 \right) I + V_{src} \left(1 + \frac{\alpha R_2}{\beta R_1} \right) - V_\gamma \tag{2.8}$$

where:

$$\alpha = \frac{\left(1 - e^{-T_{on}/\tau_{on}}\right)}{\left(1 + e^{-T_{on}/\tau_{on}}\right)}, \quad and \quad \beta = \frac{\left(1 - e^{-T_{off}/\tau_{off}}\right)}{\left(1 + e^{-T_{off}/\tau_{off}}\right)}. \tag{2.9}$$

Using the definitions:

$$k_1 = -R_2 \left(\frac{\alpha}{\beta} + 1 \right) \tag{2.10}$$

$$k_2 = V_{src} \left(1 + \frac{\alpha R_2}{\beta R_1} \right) - V_\gamma \tag{2.11}$$

it is possible to simply write:

$$V_{out} = k_1 \cdot I + k_2 \tag{2.12}$$

Another expression where V_{out} and I are involved is needed. The power balance expression can be used, that is the expression that provides the power loss in the DC-DC converter:

$$V_{src} I - \frac{V_{out}^2}{R_{load}} = \left(R_s + R_L + D R_{DS}\right) I^2 + \left(1 - D\right) V_\gamma \frac{V_{out}}{R_{load}} \tag{2.13}$$

Substituting in eq. (2.13) the expression of V_{out} given by eq. (2.12), it results a quadratic equation:

$$aI^2 + bI + c = 0 \tag{2.14}$$

where

$$a = (R_s + R_L + DR_{DS})\, R_{load} + k_1^2 \tag{2.15}$$

$$b = k_1\,(1 - D)\, V_\gamma + 2k_1k_2 - V_{src}R_{load} \tag{2.16}$$

$$c = R_S I_{SC} - R_S I_{MPP} - V_{MPP} \tag{2.17}$$

The solution of this equation provides the mean current I absorbed from the PV panel.

2.2.3 Expected behavior

In order to determine the values of the electronic components of the DC-DC converter, a software application, written in C language, implementing the model derived above, has been developed in order to simulate the behavior of the boost converter connected to a PV panel.

To accomplish this task, two PV panels have been considered: an high power module, i.e. the SUNTECH STP235-20/Wd [STP235-20/Wd, 2011], whose datasheet values have been listed in Table 2.1, and a little panel, namely the BP Solar SX310 [SX310, 2007], whose datasheet values are in Table 2.2.

The five parameters for these two panels, required to perform the simulations, have been derived by using the procedure described in [Laudani et al., 2014c]. These value are listed in Table 2.3.

Table 2.1: Datasheet values for the PV panel SUNTECH STP235-20/Wd [STP235-20/Wd, 2011]

Parameter	Value @ STC
Short-circuit current I_{SC}	8.35A
Open-circuit voltage V_{OC}	37.0V
Maximum Power current I_{mpp}	7.79A
Maximum Power voltage V_{mpp}	30.2V
Temperature Coefficient of I_{SC} ($\alpha_{I_{SC}}$)	0.055%/°C
Temperature Coefficient of V_{OC} ($\alpha_{V_{OC}}$)	−0.33%/°C
Number of cells	60

Table 2.2: Datasheet values for the PV panel BP Solar SX310

Parameter	Value @ STC
Short-circuit current I_{SC}	0.69A
Open-circuit voltage V_{OC}	21.0V
Maximum Power current I_{mpp}	0.59A
Maximum Power voltage V_{mpp}	16.8V
Temperature Coefficient of I_{SC} ($\alpha_{I_{SC}}$)	0.065%/°C
Temperature Coefficient of V_{OC} ($\alpha_{V_{OC}}$)	−0.80mV/°C
Number of cells	60

Table 2.3: Parameters obtained with the method proposed in [Laudani et al., 2014c]

Parameter	BP solar SX310	Suntech STP235-20/Wd
Ns	36	62
$I_{irr,ref}$ [A]	0.693	8.35
$I_{0,ref}$ [A]	$1.61 \cdot 10^{-8}$	$2.32 \cdot 10^{-8}$
$R_{P,ref}$ [Ω]	279.1	574.8
$R_{S,ref}$ [Ω]	1.555	0.199
n_{ref}	1.3	1.18

Table 2.4: Components values chosen for the DC-DC converter.

Component	Value
Inductor	$L = 750\mu H$, $R_L = 0.035\Omega$, $I_{max} = 12A$
MOSFET	IPP320N20N3, $I_D = 34A$, $R_{DS(ON)} = 32m\Omega$ (typ.)
Diode	MUR2020R, $I_{max} = 20A$, 200V, $t_{rr} = 95ns$, $V_\gamma = 0.7V$ (typ.)
R_{load}	SW selected 28.2Ω or 282Ω, 300W
PWM period	$10\mu s$

In the end, the components in Table 2.4 have been selected.

The internal resistive load actually is a double resistive load, of different values (28.2Ω or 282Ω), that can be SW selected. Each resistor bank is made with five 50W resistors, namely five 4.7Ω resistors for the 28.2Ω bank and five 47Ω resistors for the other one. Table 2.5 shows the expected electronic load behavior with the 235W, photovoltaic module SUNTECH STP235-20/Wd. The data in the table have been collected with $R_{load} = 28.2\Omega$.

Table 2.5: Expected behavior of the proposed electronic load for the PV module SUNTECH STP235-20/Wd.

D	I (A)	V (V)	w (W)	ΔI(mA)	η	z (Ω)	V_{load}
0.00	1.27	36.45	46.55	0.0	0.99	28.55	36.01
0.10	1.55	36.32	56.63	24.1	0.98	23.29	39.64
0.20	1.96	36.12	70.90	48.0	0.99	18.41	44.41
0.30	2.54	35.83	91.08	71.3	0.99	14.09	50.38
0.40	3.41	35.35	120.62	93.7	0.99	10.36	58.00
0.50	4.78	34.45	164.85	113.8	0.99	7.20	67.79
0.60	6.96	32.19	224.21	127.0	0.99	4.62	78.95
0.70	8.30	21.84	181.37	99.5	0.98	2.63	70.69
0.80	8.32	10.09	84.07	51.1	0.95	1.21	47.49
0.90	8.34	2.96	24.71	14.7	0.83	0.36	24.09
1.00	8.34	0.51	4.25	14.7	0.00	0.06	0.00

In the first column of the table there is the duty cycle D that set the working conditions. Only ten values are in the table.
In the table are listed, for each working condition set by D, the current drawn by the PV panel, the voltage at the output of the panel, the output power, the ripple in the input current and the output voltage. There is also listed the efficiency of the DC-DC converter, given by the the ratio between the power supplied by the PV panel and that dissipated in the load resistor:

$$\eta = \frac{V_{load} I_{load}}{V I} \tag{2.18}$$

In the second to last column there is the electronic load impedance. Its maximum value is strictly related to the value of the load resistor R_{load}. This resistor is SW selected.

In order to verify the results in Table 2.5, each of them has been verified with the Linear Technologies circuit simulator LTspice IV where a much more accurate model for each component is involved. Using the schematic diagram in Fig. 2.3 and the values in Table 2.4, the plots in Fig. 2.5 and Fig. 2.6 have been obtained. The steady state values are compared with those provided with our simulator in Table 2.6. The relative error is very low in each case.

In Fig. 2.7 and in Fig. 2.8 there are the expected characteristic curves for the two panels described above. For each one both the R_{load} have been used. It is clear from the plots that for the little panel it is better to have $R_{LOAD} = 282\Omega$, whereas for the bigger one, it is more effective to select $R_{LOAD} = 28.2\Omega$. In each case it is important to satisfy the condition $V_{out} < V_{DS,max}$, where

Table 2.6: Comparison of the values obtained with the C simulator and with LTSPice, D=0.5

Parameter	Simulator	LTSpice	error (%)
V_{pv}	34.46V	34.40V	0.17
I_{pv}	4.77A	4.86A	1.85
V_{out}	67.67A	67.76A	0.13
Δ_I	113.9mA	115mA	0.95

$V_{DS,max}$ is the maximum drain-to-source voltage of the MOSFET. The open circuit voltage V_{oc} has been obtained simulating a stop of the PWM controller ($D = 0$) and a disconnection of the resistors banks. In Fig. 2.9 there are the expected values for the electronic load, using both the resistor banks, and both the panels. In Fig. 2.10 there are the expected power losses in the DC-DC converter. In Fig. 2.11 there are the expected values for voltage and current in the load resistor bank.

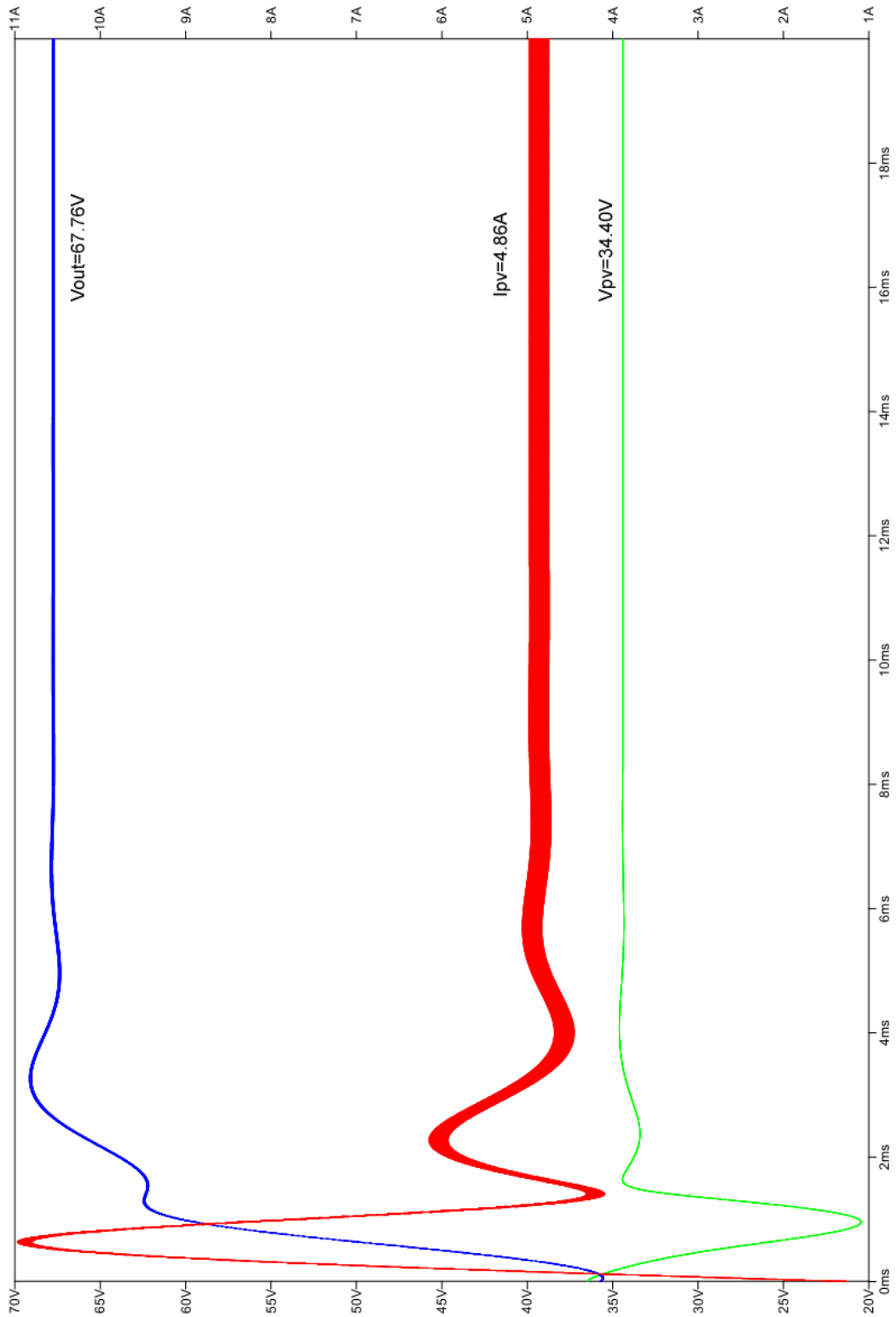


Figure 2.5: Input and output voltage and current provided by the circuit simulator LTSpice IV

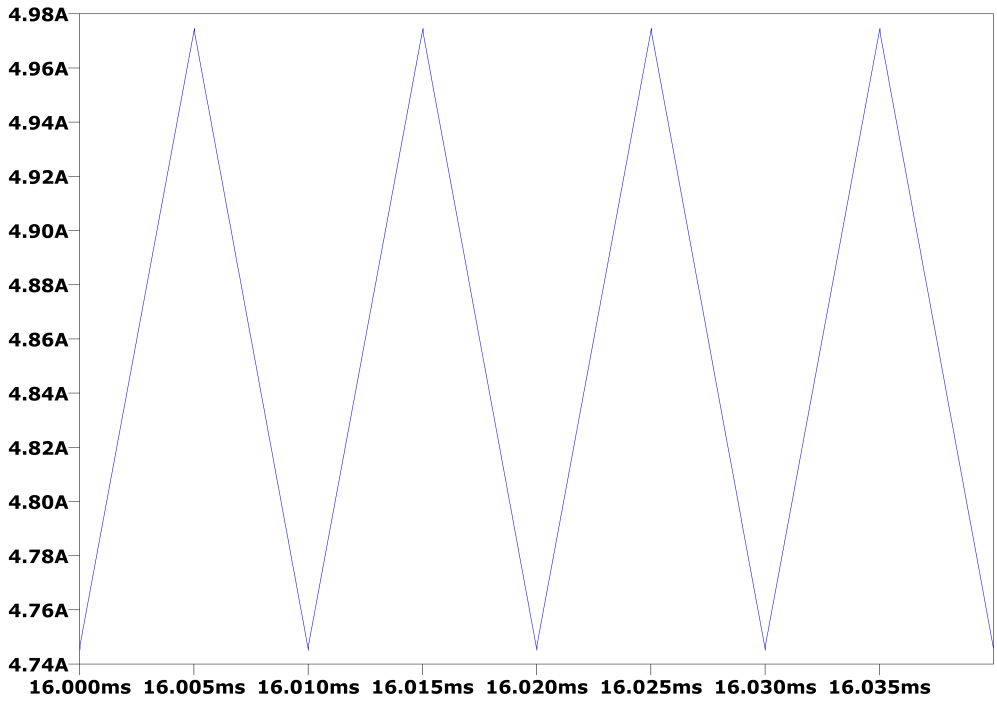


Figure 2.6: Ripple in the input current provided by the circuit simulator LTSpice IV

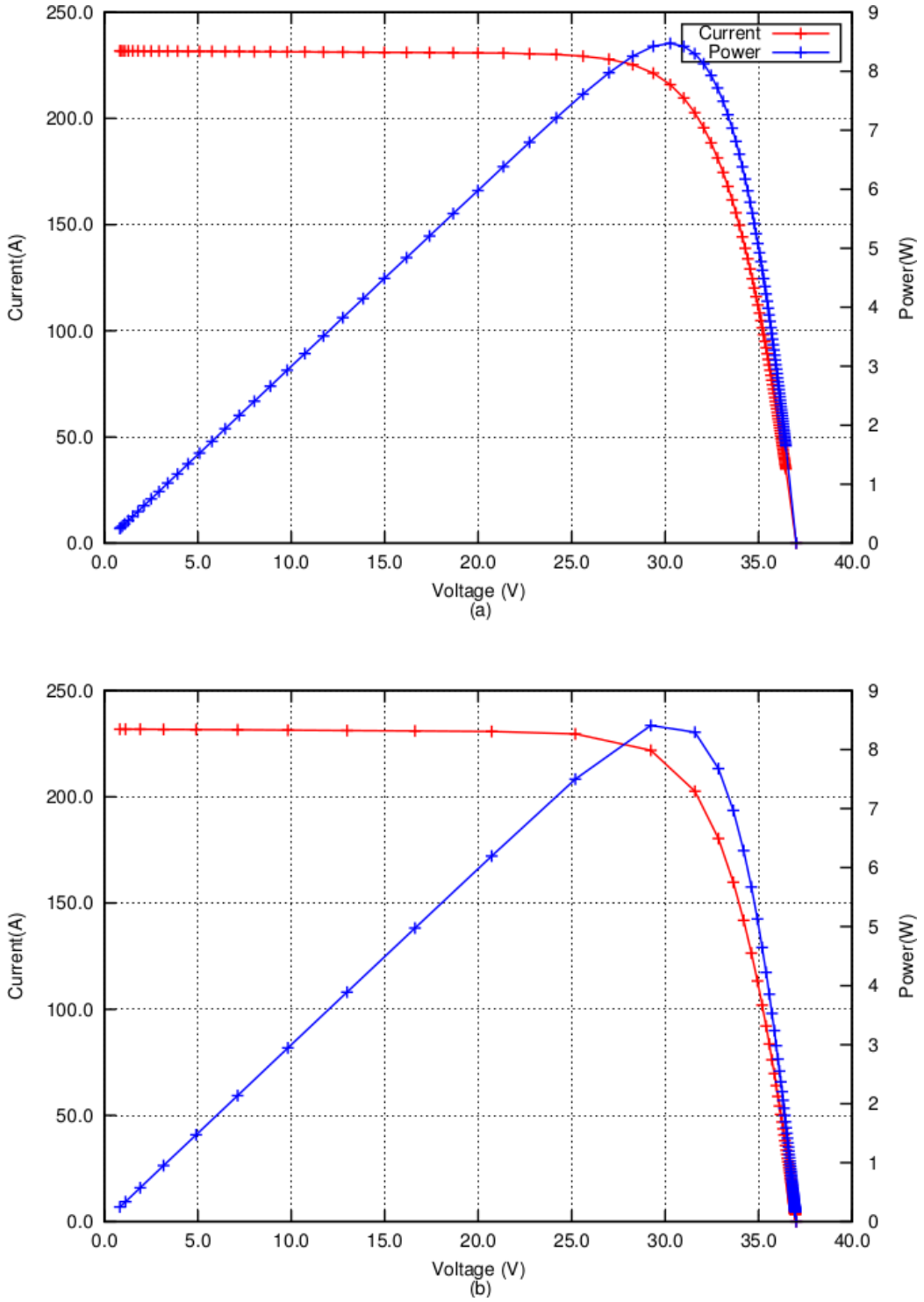
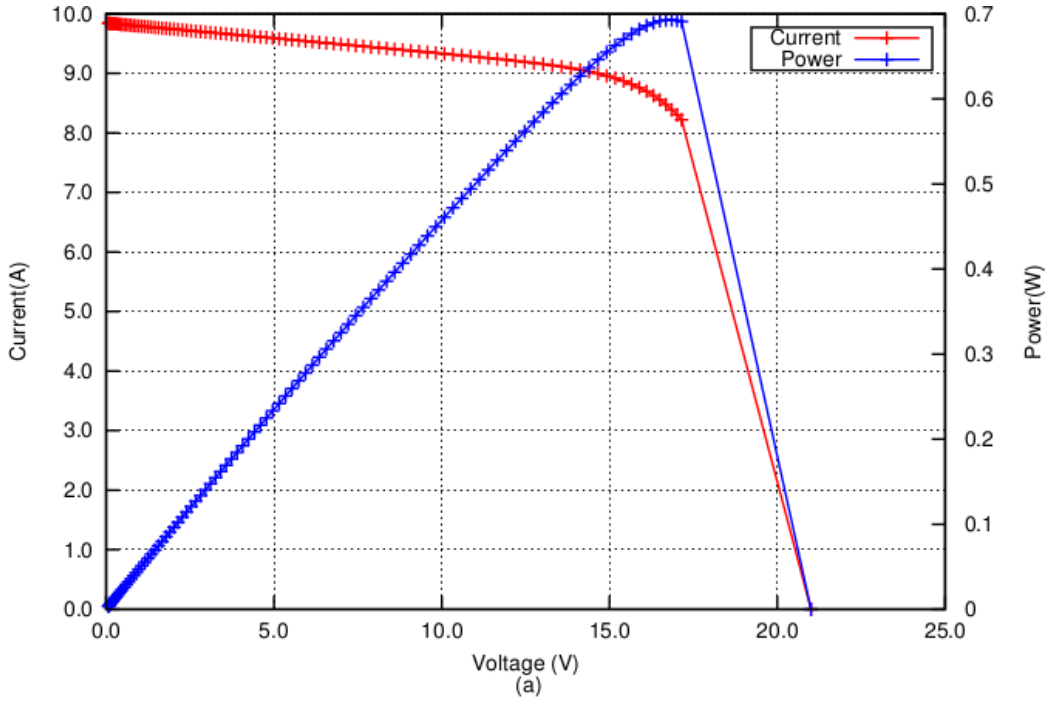


Figure 2.7: Simulation of tracing with the PV module Suntech STP235-20/Wd. $R_{LOAD} = 28.2\Omega$ for the (a) case, and $R_{LOAD} = 282\Omega$ for the (b) case.



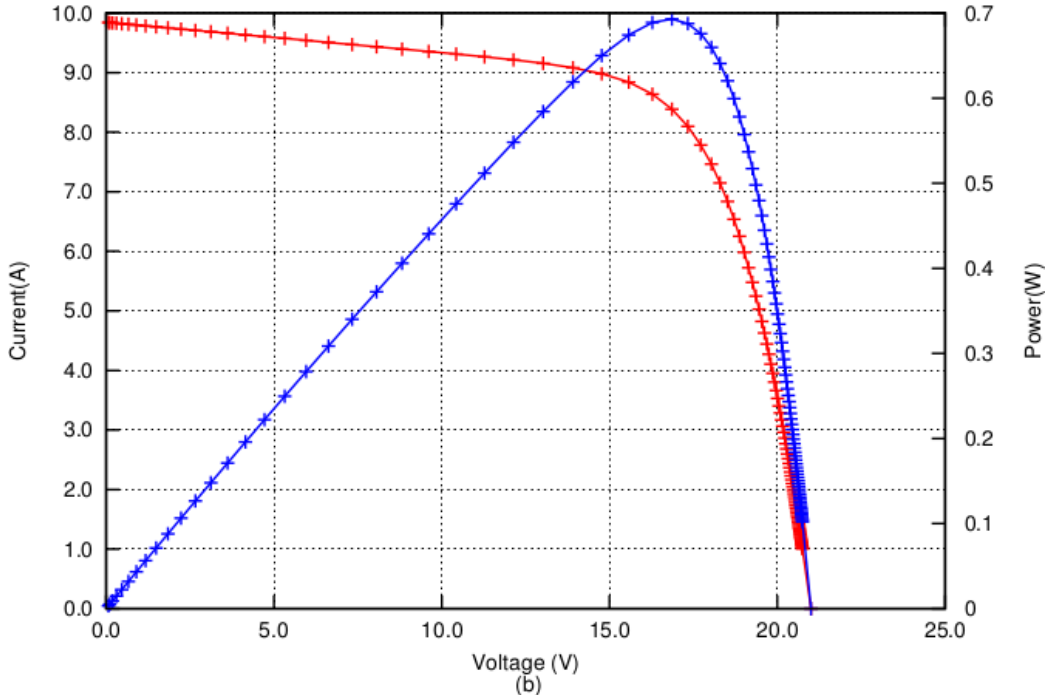


Figure 2.8: Simulation of tracing with the PV module BP solar SX310. $R_{LOAD} = 28.2\Omega$ for the (a) case, and $R_{LOAD} = 282\Omega$ for the (b) case.

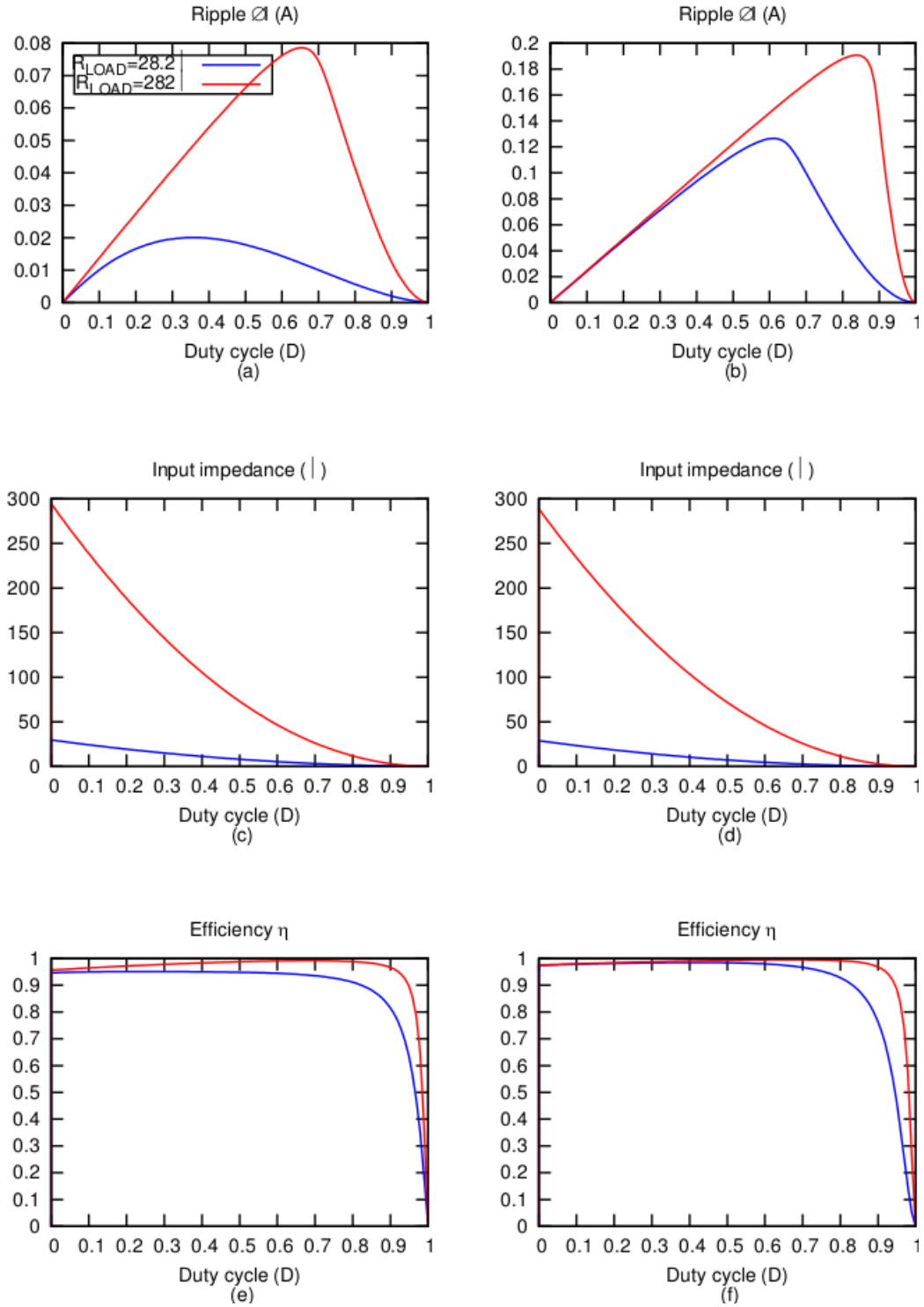


Figure 2.9: Current ripple, input impedance, and efficiency of the buck converter. The column on the left refers to the PV module BP solar SX310 (10W), whereas the plots of the right column are related to the PV module Suntech STP235-20/Wd (235W).

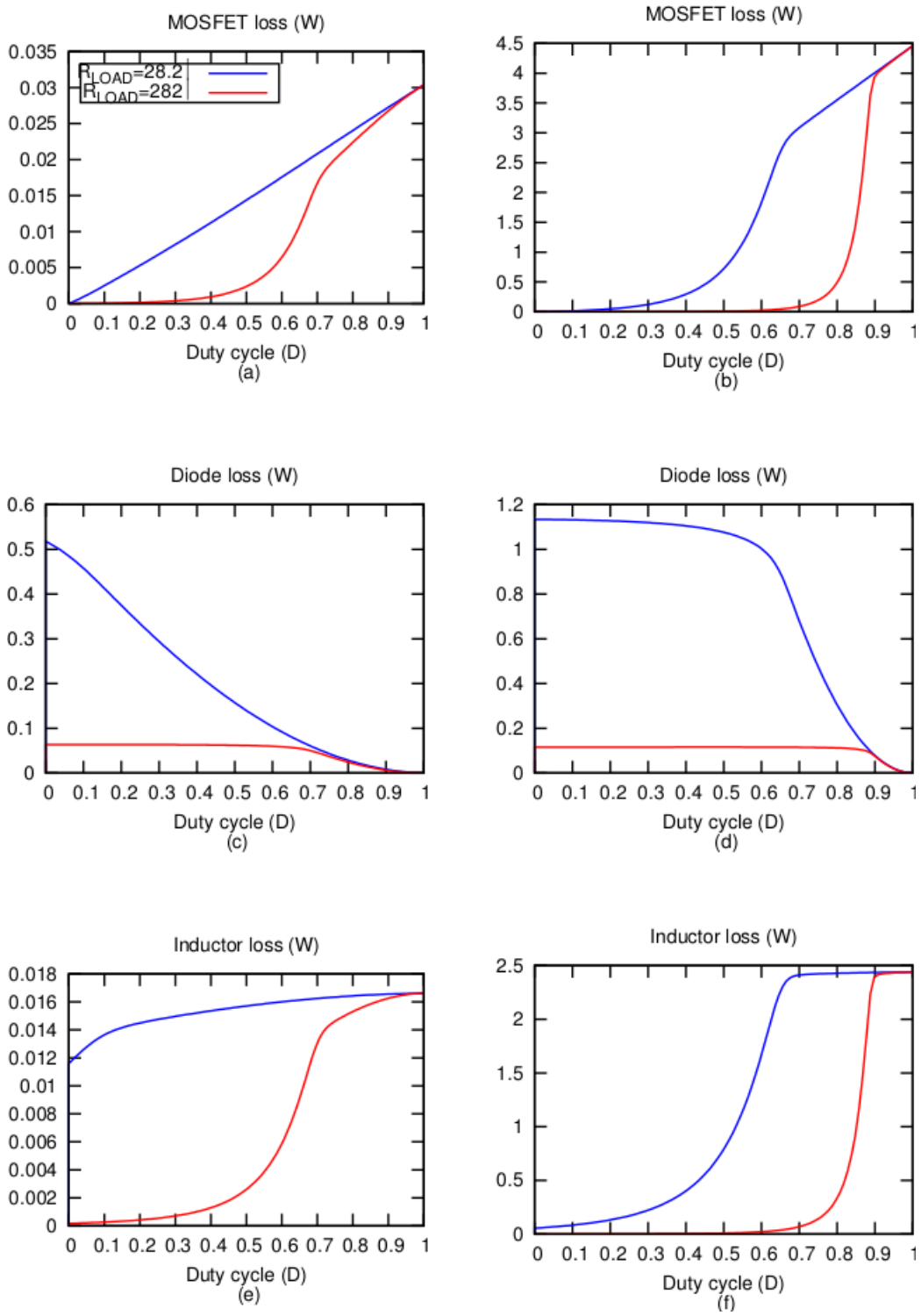


Figure 2.10: Power loss in the components of the DC-DC converter. The column on the left refers to the PV module BP solar SX310 (10W), whereas the plots of the right column are related to the PV module Suntech STP235-20/Wd (235W).

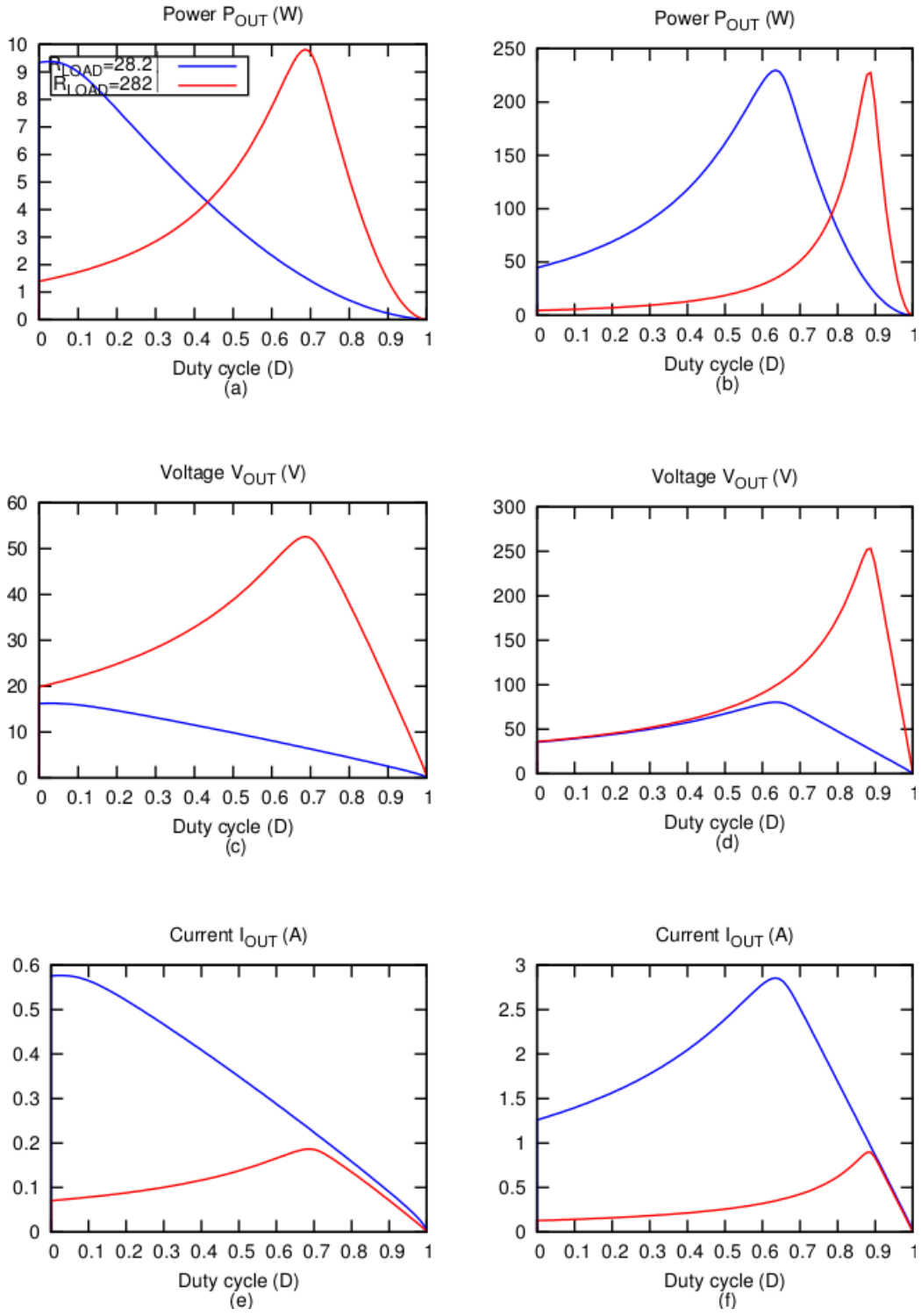


Figure 2.11: Voltage, current, and power expected in the resistive internal load. The column on the left refers to the PV module BP solar SX310 (10W), whereas the plots of the right column are related to the PV module Suntech STP235-20/Wd (235W).

2.2.4 The DC-DC boost converter realization

After having defined the values of the components of the boost converter, listed in Table 2.6, it is possible to build a real DC-DC converter. The schematic diagram in Fig. 2.3 is just an ideal DC-DC converter. In a real device, some other sub-circuits are needed.

The first one is the driver for the MOSFET. It must provide a suitable V_{GS} value to guarantee a good conduction. The value of 10V has been chosen to drive the MOSFET and the Microchip TC1411 as driver.

Furthermore, two other MOSFETs are required to select the load resistor banks. In the end, two PI low-pass LC filters have been added. The first one is used to reduce conducted electromagnetic interference (EMI) toward the PV panel. The second one the conducted EMI toward the load resistors.

It is worth nothing that radiated EMI generated by the ripple in the internal load should be reduced in order to reduce the generation of induced EMI in the signal conditioning and measurement unit. The electrolytic capacitor use in the prototype are suitable for switching applications, that means they exhibit low equivalent series resistance (ESR).

The printed circuit board (PCB) that hosts all the components, has been made by hand. In Fig. 2.13 is shown the final complete schematic diagram of the boost converter.

Inductor realization

All the components in the schematics can be easily purchased, but there is no commercial choice for the required inductor. The only solution was to find a suitable magnetic core and to wrap in it the right number of turns

Table 2.7: ETD59\31\22 Effective core parameters

Symbol	Parameter	Value	Unit
$\Sigma(I/A)$	core factor (C1)	0.378	mm^{-1}
V_e	effective volume	51500	mm^3
l_e	effective length	139	mm
A_e	effective area	368	mm^2
A_{min}	minimum area	360	mm^2
m	mass of the core half	130	g

using a thick copper wire. A toroidal core was a preferable solution because the magnetic field is completely confined in it. Even if using the biggest one in the market, it was not possible to satisfy the requirements of current (10A) and inductance (750µH) at the same time. The solution was found using the biggest ETD type magnetic core from Ferroxcube, namely the model ETD59\31\22 [ETD59-31-22, 2008], and using a "gap" in the core in order to reduce the magnetic flux through it. The effective core parameters of this chip are summarized in the Table 2.7, whereas in Fig. 2.12 are plotted its dimensions. The core is made with material 3C90 [3C90, 2008], that is suitable for power transformer operating at frequencies up to 0.2MHz.

If a gap is inserted in the core, the effective magnetic permeability μ_e of the core is given by the following:

$$\mu_i = \frac{\mu_i}{1 + \frac{G\mu_i}{l_e}} \tag{2.19}$$

where μ_i it the magnetic permeability of the material used and G is the gap width. Eq. (2.19) is good only for little gap. Assuming an operating temperature of 60°C the datasheet of the material 3C90 provides the value

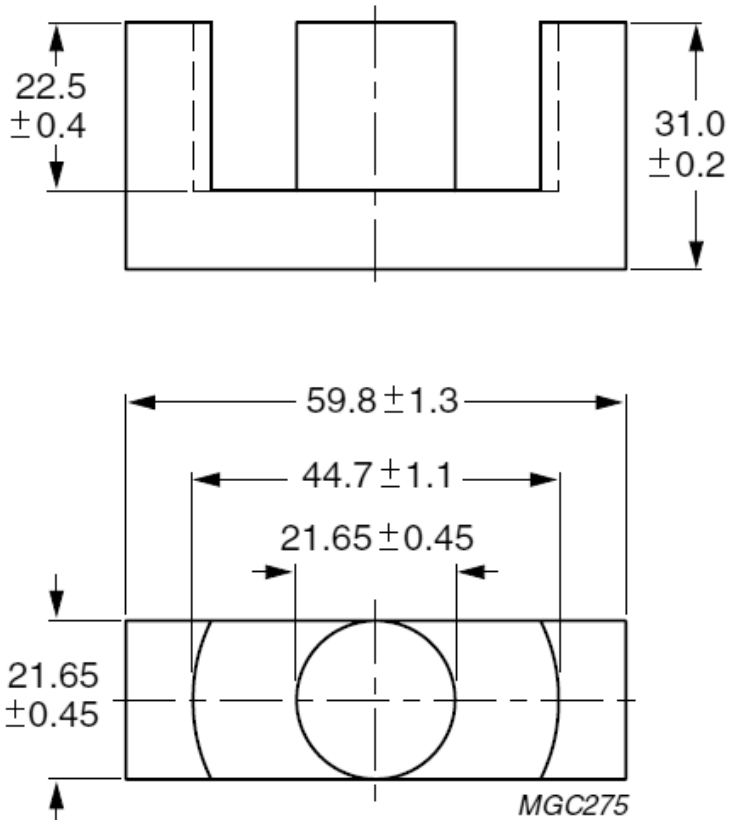


Figure 2.12: ETD59\31\22 core half. Dimensions in mm

of 2700. Considering G=3mm equation eq. (2.19) gives the value $\mu_e = 45.5$. Since the inductance of a solenoid is:

$$L = \mu \frac{N^2 A_e}{l_e} \tag{2.20}$$

it is easy to obtain that the number of turns N to have the wanted value $L = 750\mu H$ is N=70.

The inductor is intended to be used with currents up to 10A, therefore, even when working with this current, the magnetic flux density B must be below

the saturation value B_{max} that is about 400mT for the material used 3C90. The flux density B is given by the following:

$$B = \mu \frac{N \cdot I}{l_e} \tag{2.21}$$

that with $I = 10$A gives the value of 287mT, that is below the saturation value B_{max}. In Fig. 2.14 there is a picture of the DC-DC converter prototype. It is easy to see the big inductor and the gapes in its magnetic core.

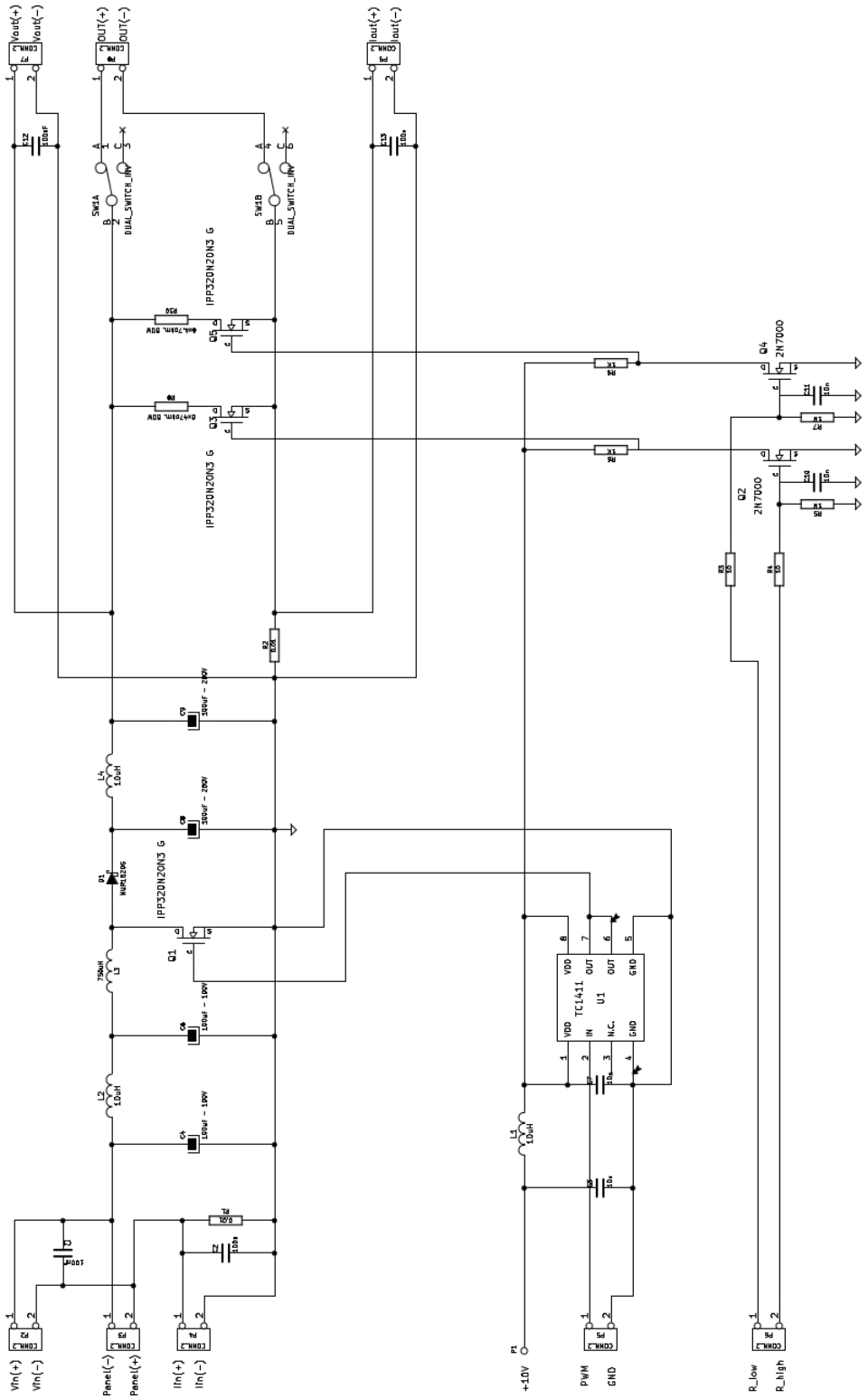


Figure 2.13: The final boost converter schematic diagram.

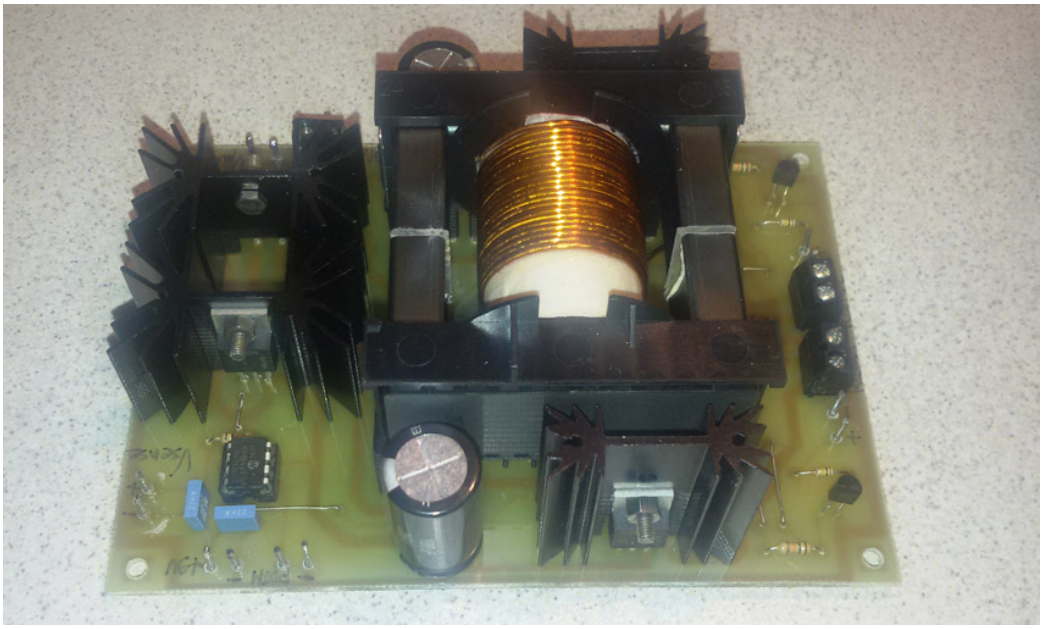


Figure 2.14: The first prototype of the DC-DC boost converter.

2.3 The signal conditioning and measurement unit

The signal conditioning and measurement unit is in charge to collect the values of current and voltage both at the input of the instrument and on the load. When operating as PV panel characteristic curve tracer, only the values in the input are meaningful, but for other application it could be interesting to know what happen at the output. For example, if the load is a battery (external load), it is required to take into account both voltage and current at the output.

The heart of this unit is the Analog Device 24 bits ADC chip AD7714Y [AD7714, 1998]. The main features of this chip are summarized in the Table 2.8, whereas in Fig. 2.15 there is its functional block diagram.

All the chip configurations are performed by means of internal registers that are accessible through an SPI interface.

This device can be configured to have three fully differential inputs or five pseudo-differential inputs, that means the five inputs share the negative input, in addition, some of the input channel combinations share the calibration registers too. Since we need four channels, pseudo-differential configuration is used. In Table 2.9, there is the channel configuration used in our system. Due to the registers calibration sharing in channel AIN3 and AIN4 of the chip AD7714Y, a new calibration should be restarted each time a channel change between AIN3 and AIN4 occurs, otherwise the measurement error could be unacceptable. Another option is to save the values of the calibration registers

Table 2.8: Analog Device 24 bits ADC chip AD7714Y

MAIN FEATURES
Charge Balancing ADC
24 Bits No Missing Codes
0.0015% Nonlinearity
Five-Channel Programmable Gain Front End
Gains from 1 to 128
Can Be Configured as Three Fully Differential
Inputs or Five Pseudo-Differential Inputs
Three-Wire Serial Interface
SPI™, QSPI™, MICROWIRE™ and DSP Compatible
Low Noise (¡150nVrms)
Low Current (350µA typ) with Power-Down (5mA typ)
2.7V V to 3.3V or +4.75V to +5.25V Operation
0.0010% Linearity Error
Schmitt Trigger on SCLK and DIN
Low Current (226µA typ) with Power-Down (4mA typ)
Lower Power Dissipation than Standard AD7714
Available in 24-Lead TSSOP Package
Low-Pass Filter with Programmable Filter Cutoffs
Ability to Read/Write Calibration Coefficients

Table 2.9: Input channels configuration

Purpose	AIN(+)	AIN(-)	Calibration register pair	Input range	G
V_{in}	AIN1	AIN6	Register Pair 0	[0...63V]	1/51
I_{in}	AIN2	AIN6	Register Pair 1	[0...10A]	11
V_{out}	AIN3	AIN6	Register Pair 2	[0...250V]	1/201
I_{out}	AIN4	AIN6	Register Pair 2	[0...10A]	11

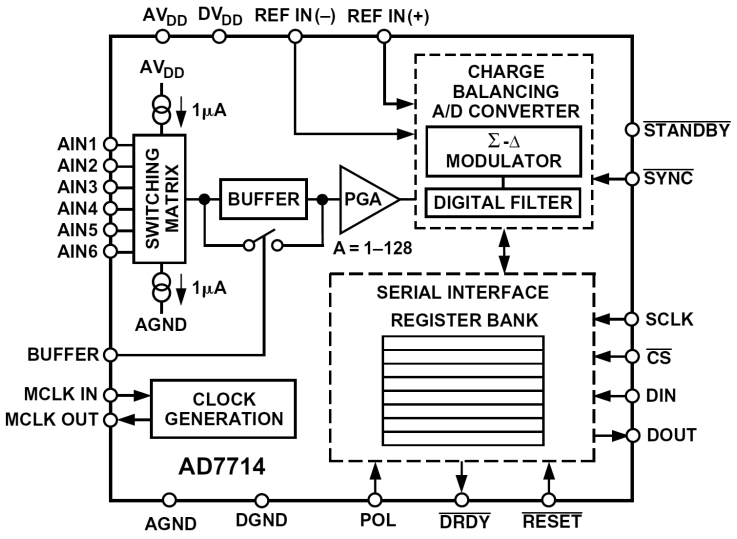


Figure 2.15: AD7714 functional block diagram.

and to load the proper ones before a measurement.

Since this procedure is faster than repeating the calibration procedure, this option has been adopted in the SW in case both V_{out} and I_{out} are required. The ADC chip has an on-chip low-pass digital filter which processes the input signal. This filter is a low-pass filter with a $(sin(x)/x)^3$ response. The transfer function for this filter is described in the z-domain by:

$$H(z) = \left[\frac{1}{N}\frac{1-z^{-N}}{1-z^{-1}}\right] \tag{2.22}$$

and in the frequency domain by:

$$H(f) = \left[\frac{1}{N}\frac{sin\,(N\pi f/f_s)}{sin\,(\pi f/f_s)}\right] \tag{2.23}$$

where N is a 12-bit value loaded in the filter register. The first filter notch frequency is $f_{CLK_{IN}}/128/N$. With the nominal $f_{CLK_{IN}}$ of 2.4576MHz, this

results in a first notch frequency range from 4.8Hz to 1.01kHz.
Fig. 2.16 shows the filter frequency response when the first filter notch frequency is set to 10Hz. The plot is shown from dc to 65Hz. This response is repeated at either side of the input sampling frequency and at either side of multiples of the input sampling frequency. The filter cut-off frequency, in association with the gain selection, also determines the output noise (and hence the effective resolution) of the device. In Table 2.10, are both the output noise and the effective resolution for some filter first notch frequencies and gain $G = 1$. In the system the gain is programmable, but PV panel characterizations are usually performed with $G = 1$ and with a first notch frequency of 50Hz in order to reduce the EMI noise produced by the power grid that could affect the measurement operations.
When a single channel is used, the sample rate is the same of the frequency of the first notch of the filter. When a new channel is selected, however, the settling time of the filter to a full-scale step input change is worst case 4 x 1/(output data rate). For example, with the first filter notch at 50Hz, the settling time of the filter to a full-scale step input change is 80ms max.
In Table 2.10 the effective sample date rate at various first notch frequencies, with 1, 2 or 4 channels, are reported.

Chip configuration is performed through the following steps:

- Chip Reset
- Low-pass filter setting (for each channel)
- Gain setting (for each channel)

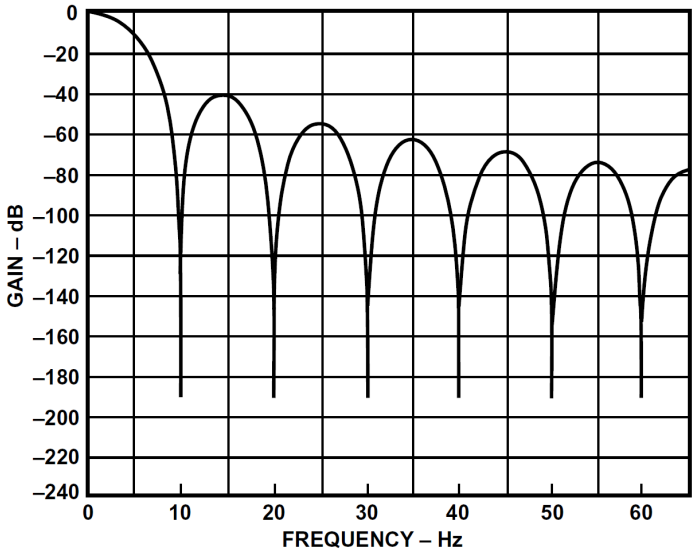


Figure 2.16: Frequency Response of AD7714 Filter

- Self calibration (for each channel). This operation is a single step operation that includes a zero-scale calibration, that is performed at the selected gain on internally shorted (zeroed) inputs, and the full-scale calibration, that is performed at the selected gain on an internally generated $V_{REF}/SelectedGain$

In order to provide to the ADC chip the most suitable signals level, a proper signal conditioning circuit must be provided. The input range for the ADC chip is $[0, ..., 1.25\text{V}]$, therefore, assuming a shunt resistor with value $0.01\Omega]$ for current measurements, the gains in Table 2.9, shall be provided.

With the input gain listed in Table 2.9, four conditioning circuits have been designed. They are build around the instrumentation amplifier chip, by Analog Device, AD623 [AD623, 2016]. The main features of this chip are

Table 2.10: AD7714Y Output Noise/Resolution

Filter first notch (Hz)	-3dB filter freq (Hz)	Output RMS Noise (µV)	Effective res, (bits)	Data rate 1 ch (sps)	Data rate 2 ch (sps)	Data rate 4 ch (sps)
5	1.31	1.07	21	5	0.8	0.4
10	2.62	1.69	20.5	10	1.7	0.9
25	6.55	3.03	19.5	25	4.1	2.0
30	7.86	3.55	19.5	30	5	2.5
50	13.1	4.72	19	50	8.3	4.2
60	15.72	5.12	19	60	10	5
100	26.2	9.68	18	100	16.6	8.4
250	65.5	44	16	250	41.7	20.8
500	131	304	13	500	83.3	41.7
1000	262	1410	11	1000	166.7	83.4

summarized in the Table 2.11, whereas in Fig. 2.17 there is its functional block diagram.

With this chip it is very easy to built conditioning circuits with excellent characteristics. For the current measurements a single 10kΩ is required to set a gain $G = 11$, whereas, for voltage measurements, the gain is set to $G = 1$ removing this resistor. An external network provides the required attenuation.

In Fig. 2.18 there are the conditioning circuits used for voltage and current measurements. The resistor R1 is set to 50kΩ for the input voltage measurement ($G = 1/51$), and to 200kΩ in order to set ($G = 1/201$) for the output voltage measurements.

In addition to providing the proper attenuation or gain for each channel, the conditioning circuit should also remove all the high frequency noise in input. This noise, once rectified, appears as dc offset error in output. Resistor

Table 2.11: Instrumentation Amplifier AD623

MAIN FEATURES
Easy to use
Rail-to-rail output swing
Input voltage range extends 150mV below ground(single supply)
Low power, 550µA maximum supply current
Gain set with one external resistor
- Gain range: 1 to 1000
High accuracy dc performance
0.10% gain accuracy (G = 1)
0.35% gain accuracy (G >1)
Noise: 35 nV/vHz RTI noise at 1kHz
Excellent dynamic specifications
- 800kHz bandwidth (G = 1)
- 20µs settling time to 0.01% (G = 10)

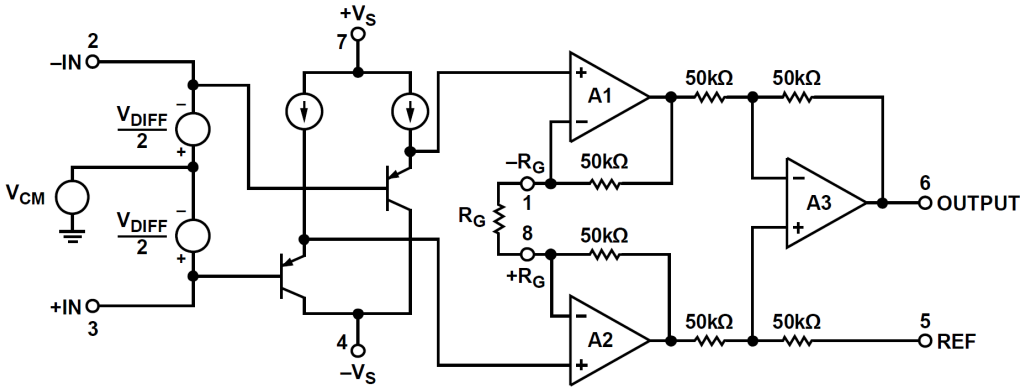


Figure 2.17: AD623 functional block diagram.

R1 and capacitor C1 (and likewise, R2 and C2) in Fig. 2.18 form a low-pass RC filter that has a -3 dB bandwidth equal to $f = 1/(2\pi R1C1)$. With the values shown, the -3 dB bandwidth of approximately 40kHz, that is below the first harmonic of the PWM signal. To preserve common mode rejection in the pass band of the amplifier, used C1 and C2 are 5% polyphenylene

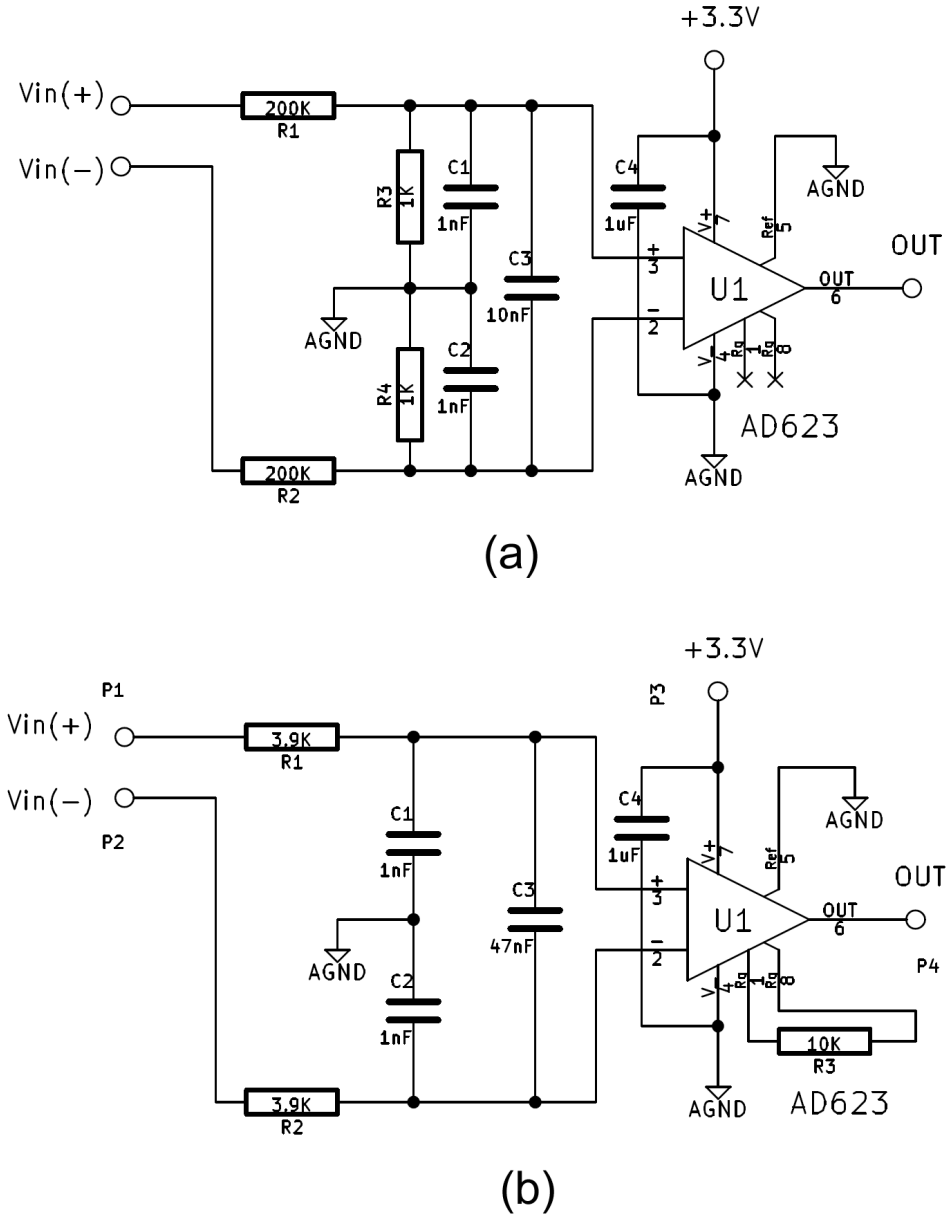


Figure 2.18: Conditioning circuits for voltage (a) and current (b) measurements

sulfide (PPS) film capacitors.

C3 ensures that any RF signals are common mode and are not applied differentially. This second low-pass network, R1 + R2 and C3, has a -3 dB frequency equal to $1/(2\pi(R1 + R2)(C3))$. Using a C3 value of 0.047µF, the -3 dB signal bandwidth of this circuit is approximately 400Hz.

Circuit realization

Building a measurement electronic circuit, working in a harsh environment due to the powerful DC-DC switching converter working in the same box, is a challenging task.

Switching operations on currents that can be up to 10A produce strong radiated and conducted EMI. For these reason big attention must be paid in order to mitigate measurement errors and digital signal integrity between the ADC chip and the control and communication unit.

Decoupling filters and digital isolators have been used in order to break the loops and dump the propagation of EMI noise through the power lines, as depicted in Fig. 2.20. All the decoupling capacitors used are ceramic type with very low Equivalent Series Resistance.

Digital isolators ISO724x [SLLS868S, 2016] are high speed devices that have logic-input and logic-output buffers separated by Texas Instrument's silicon-dioxide (SiO2) isolation barrier. In our case, they are used to break the loops in digital lines where induced currents could mine signal integrity of the SPI bus. This bus is very sensitive to EMI noise since every clock transition make the data be sampled.

In Fig. 2.19 there is the conceptual Block Diagram of a Digital Capacitive

Isolator. The capacitors are used to provide the galvanic isolation between the two circuits. This solution is much faster than opto-isolators.

During the sub-circuits analysis of connections phase, a lot of attention was payed to the problem of EMI propagation that a new kind of decoupling filter has been conceived and proposed in [Gaiotto et al.,]. This work is also reported in Appendix A.

The final schematic of the circuit for voltage and current measurements acquisition is reported in Fig. 2.21. It is possible to see some other components related to power supply and other minor tasks like power-on/power-off of the instrument, fan control etc. In Fig. 2.22 there is the PCB designed to host this circuit. It is a double faces PCB (2 layers) that has been designed with the SW KiCad ver. 4.0.6 and made by Würth Elektronik.

In Fig. 2.23 there is a picture of the assembled board. Each sub-circuit of it was accurately verified. Only few little adjustments have been required.

The accuracy of the the measurements has been verified against the digital multimeter Fluke 115. The measurement error is below 1% both for current and voltage V_{in}, I_{in}, V_{out} and I_{out} all over the admittable ranges.

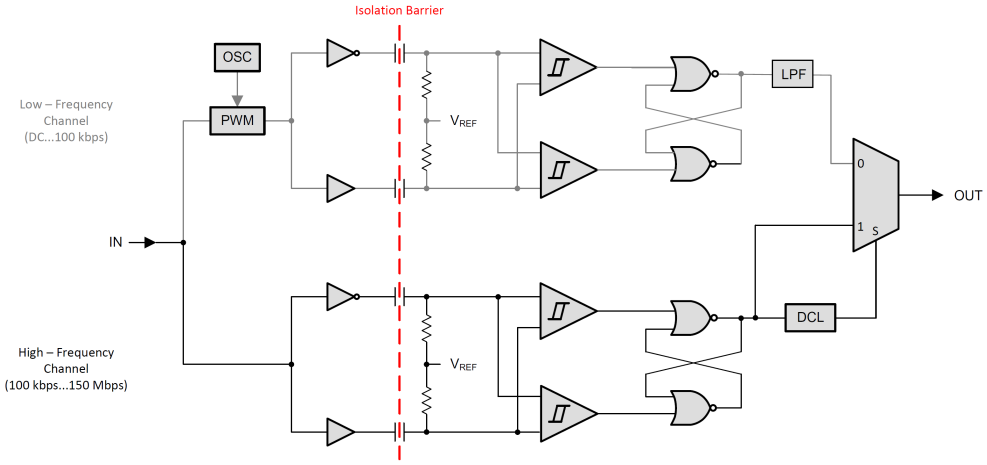


Figure 2.19: Conceptual Block Diagram of a Digital Capacitive Isolator

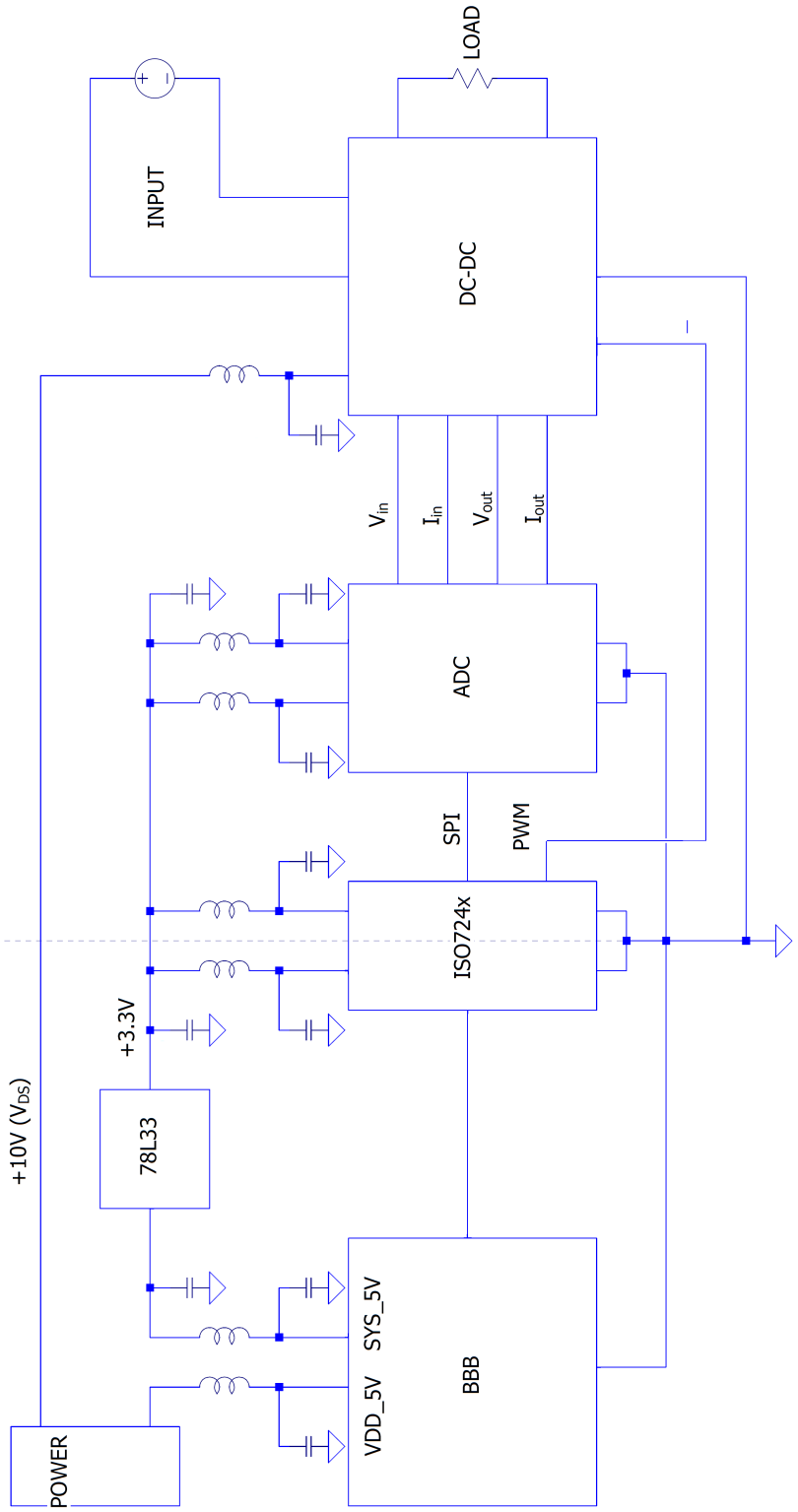


Figure 2.20: Subsystems connections

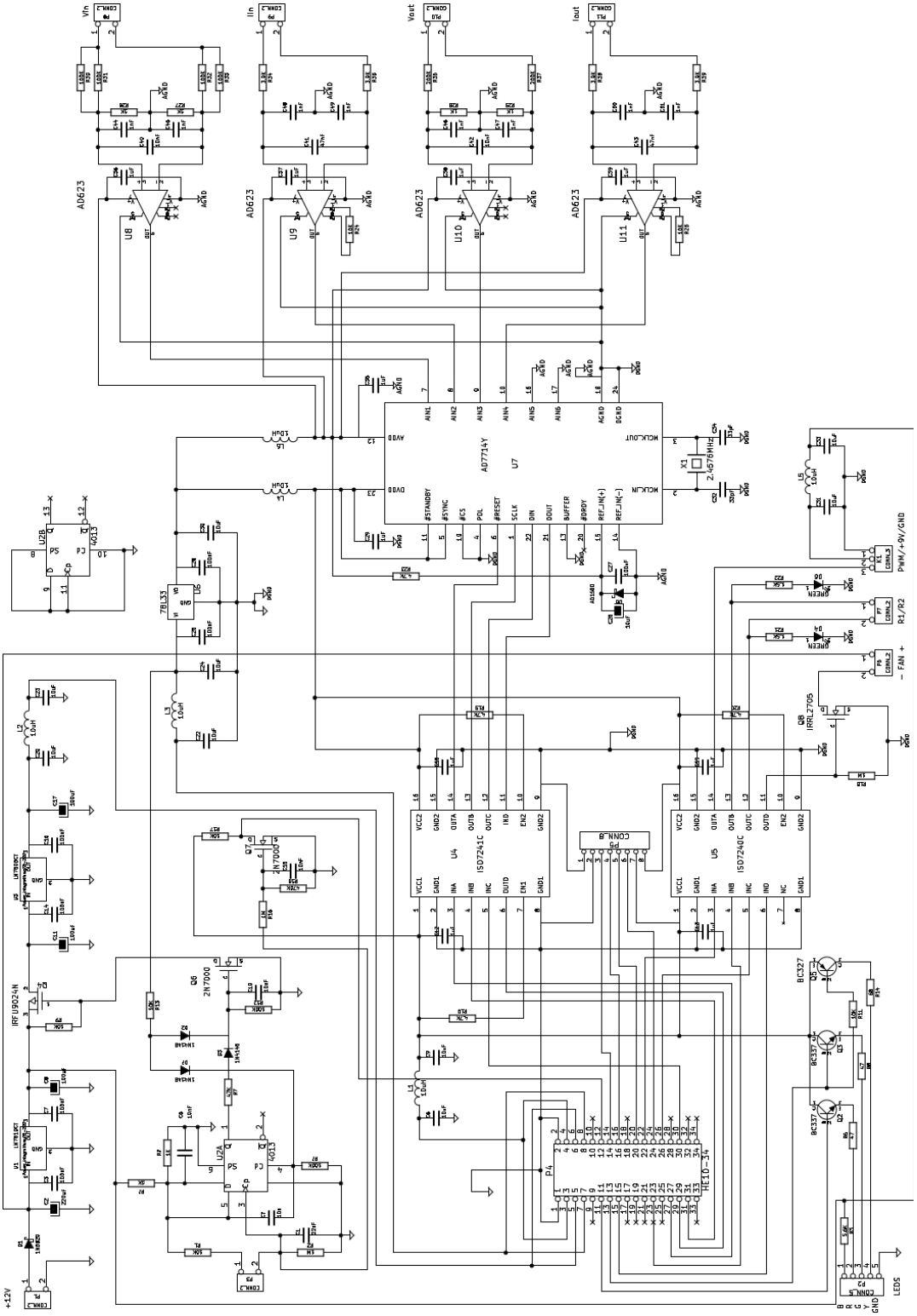


Figure 2.21: Schematic of the signal conditioning and measurement unit

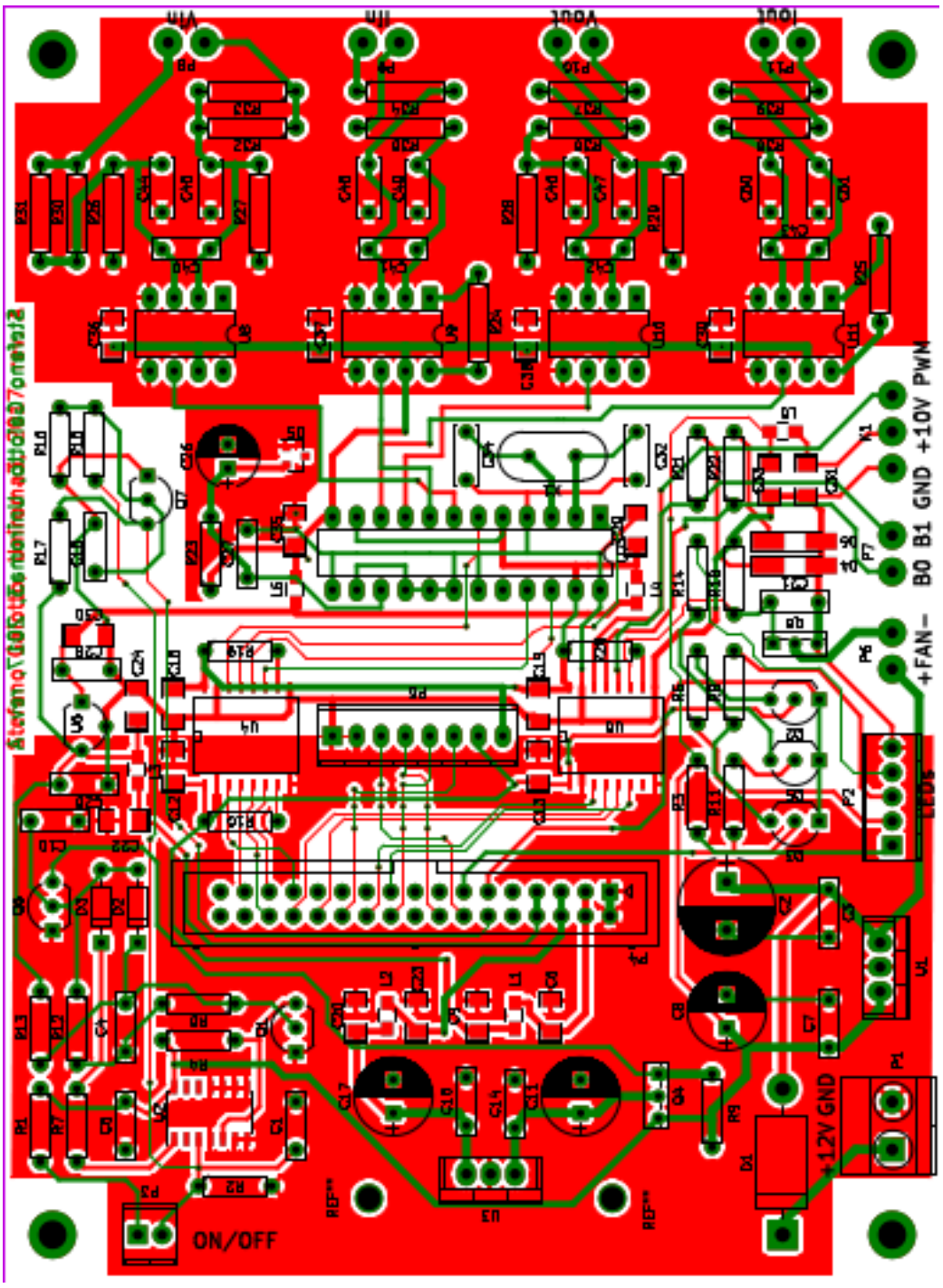


Figure 2.22: PCB layers of the signal conditioning and measurement unit

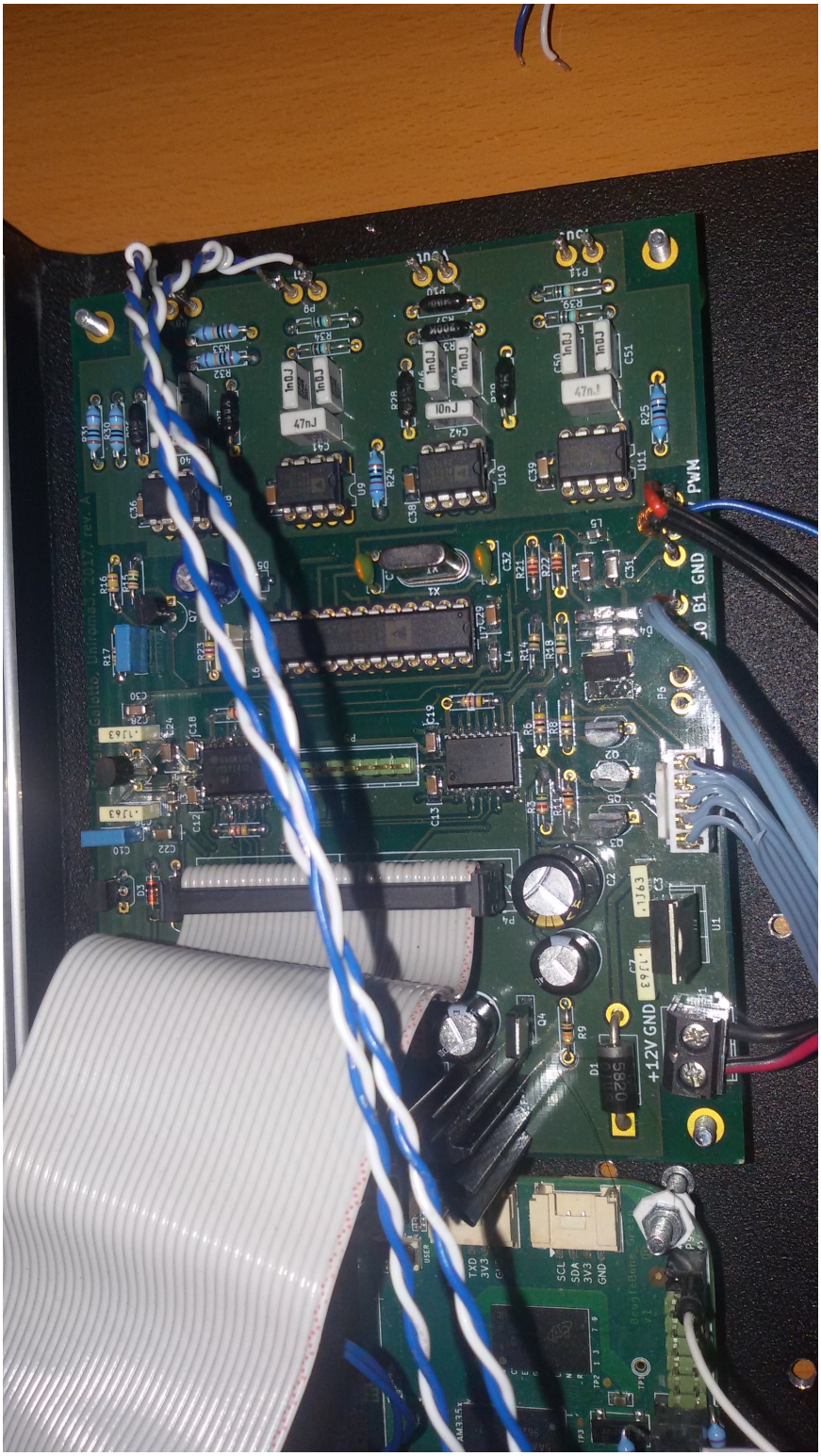


Figure 2.23: The signal conditioning and measurement unit board

2.4 The control and communication unit

As previously mentioned, the control and communication unit is in charge to provide the control of the identification operations (or algorithms verification) and the connectivity toward an external client. A single board computer (SBC) has been identified as an interesting solution due to its low cost and the high processing capabilities available today in these products.
A SBC is a complete computer built on a single circuit board, with microprocessor(s), memory, input/output (I/O) and other features required by a functional computer and are usually much more powerful than microcontroller units. Usually they have an almost complete operating system and many times it is possible to choose between many of them.
There are lot of SBCs available today, (in https://www.board-db.org/ there is an almost comprehensive search engine to find the most suitable one), among the most popular ones, BeagleBone Black rev. C and Raspberry Pi 2, model B, seemed to be the most interesting ones because of their great popularity among the "makers".
In Table 2.12 there is a comparison between them. In Fig. 2.24 there is picture of the BeagleBone Black board, whereas in Fig. 2.25 that of the Raspberry Pi 2B one. In the end, the Texas Instruments BeagleBone Black was chosen because of its big amount external connections. It has two 46 pin headers, i.e. a total of 92 possible connection points. Some of these connections are reserved, but almost all of them can be reconfigured to be used if needed.
Among all the available possibilities, the high resolution PWM controllers was considered a crucial feature. With f = 100kHz, the resolution of the

PMW controller is 15.8 bits (0.002%) that allows the tracer to have a great resolution when tracing the $I-V$ curves or, more generally, in any situation where a fine adjustment of the DC-DC converter is required.
Another interesting characteristic of the BBB is the possibility to host a programmable real time unit (PRU) if it is required to investigate about an application where hard real time constraints must be satisfied.
Actually in the instrument it is used a Seeed Studio BeagleBone Green (BBG) board. This board, due to the lack of the HDMI connector, is a cheaper version of the BBB.

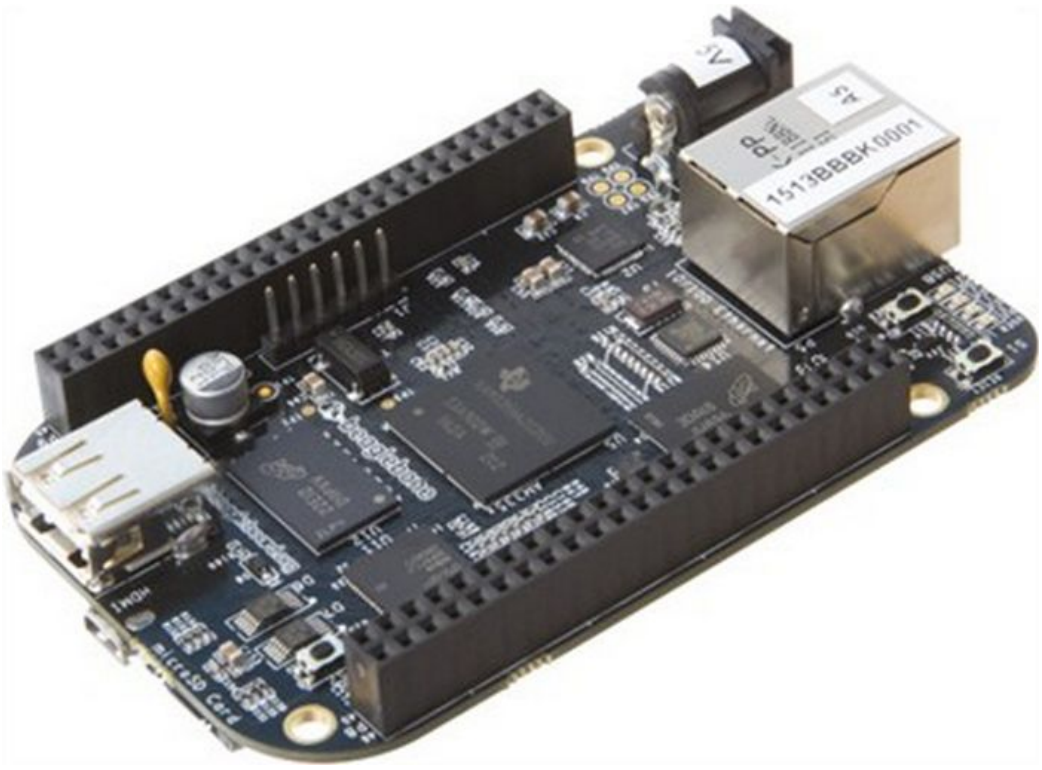


Figure 2.24: The BeagleBone Black, rev. C, board

Table 2.12: Comparing Raspberry Pi 2, Model B and BeagleBone Black rev. C

	BeagleBone Black rev. C	Raspberry Pi 2, Model B
Base Price	55 EUR	32 EUR
Chip	TI Sitara AM3358BZCZ100	Broadcom BCM2837 SoC
Core architecture	ARM Cortex A8	Quad-core ARM Cortex-A7
CPU frequency	1 GHz	900 MHz
RAM	512 MB DDR3L @ 800MHz	1GB LPDDR2 @ 400MHz
Storage	2GB, 8bit Embedded MMC	Micro SD
Video Out	16b HDMI, 1280x1024 (MAX)	HDMI (rev 1.3 & 1.4)
Supported Resolutions	1024x768,1280x720,1440x900, 1920x1080@24Hz w/EDID Support	Extensive from 640x350 up to 1920x1200, this includes 1080p
Audio Output	over HDMI	Over HDMI, 3.5 mm jack
Operating Systems	Debian(Default), Angstrom, Android, ArchLinux, Gentoo, Minix, RISC OS, Ubuntu...	Raspbian (Recommended), Ubuntu, Android, ArchLinux, FreeBSD, Fedora, others...
Power	210-430mA @ 5V under varying conditions	150-350mA Micro USB socket 5V, 2A
Dimensions	86.36 x 53.34mm	85 x 56 x 17mm
Expansion Connectors	Power 5V, 3.3V , VDD_ADC(1.8V) McASP0, SPI1, I2C, GPIO(69 max), LCD, GPMC, MMC1, MMC2, 7AIN(1.8V MAX), 4 Timers, 4 Serial Ports, CAN0, EHRPWM(0,2),XDMA Interrupt, Power button, Expansion Board ID	40-pin 2.54mm expansion header Providing 27 GPIO pins as well as +3.3 V, +5 V and GND supply lines
Peripherals	1 USB Host, 1 Mini-USB Client, 1 10/100 Mbps Ethernet	2 USB Hosts, 1 Micro-USB Power, 1 10/100 Mbps Ethernet, RPi camera connector

Table 2.13: BeagleBone GPIOs configuration

P9 connector pin	direction	usage	gpio
P9_01	GND		-
P9_02	GND		-
P9_03	3.3V		-
P9_04	3.3V		-
P9_05	VDD_5V		-
P9_06	VDD_5V		-
P9_07	SYS_5V		-
P9_08	SYS_5V		-
P9_11	OUT	LED RED	30
P9_12	IN	OFF	60
P9_13	OUT	LED GREEN/YELLOW	31
P9_14	-	RESERVED EXP.	-
P9_15	OUT	FAN	48
P9_16	-	RESERVED EXP.	-
P9_20	-	RESERVED EXP.	-
P9_22	OUT	PWM	-
P9_23	OUT	BANK_1 (282Ω)	14
P9_24	-	RESERVED EXP.	-
P9_26	OUT	BANK_0 (28.2Ω)	49
P9_27	OUT	AD7714 - RESET	115
P9_29	IN	AD7714 - MISO	-
P9_30	OUT	AD7714 - MOSI	-
P9_31	OUT	AD7714 - CLK	-

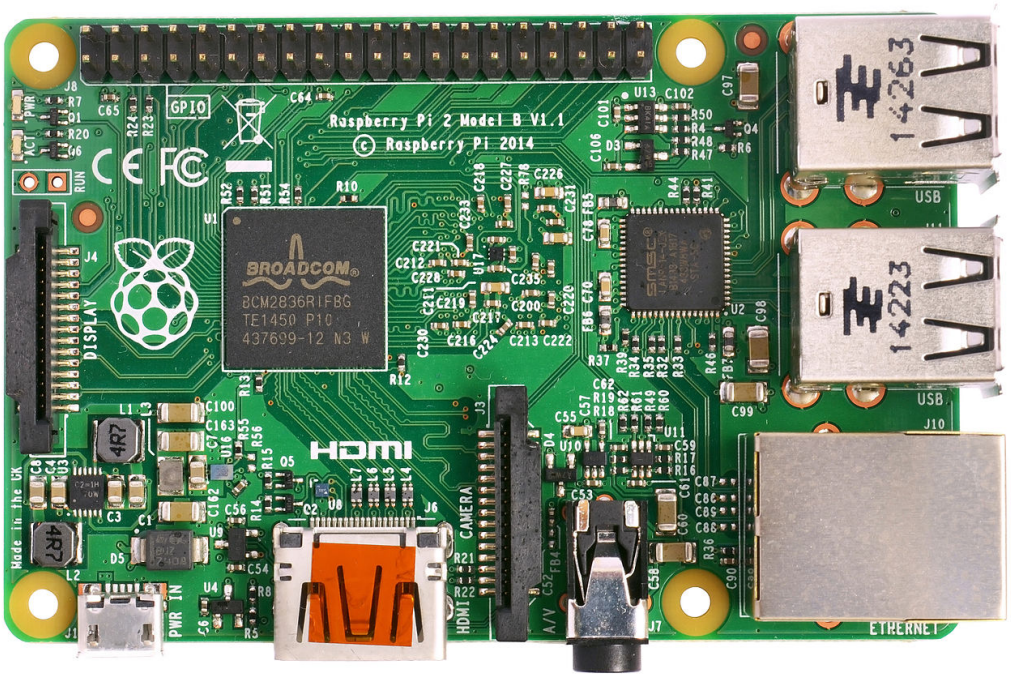


Figure 2.25: The Raspberry Pi 2, Model B, board

In Table 2.13 there is the GPIOs configuration used in the system. The configuration of the GPIOs is performed by means of the Linux device tree overlay file. This file is a textual description of the HW configuration we want to have. Once processed with the "device tree compiler" it results in the "device tree block" file that, loaded with the Linux kernel image, make the kernel perform the desired configuration of the GPIOs.

Since GPIOs are multiplexed among many devices that are in the main processor, this file provides the Linux kernel all the information required to properly configure the GPIOs and to have the configuration in Table 2.13. After the Linux boot process, the board is ready to be used. We developed all our SW applications in C language using the GNU gcc C compiler (actually

a cross-compiler) running on an Intel x86 personal computer.

2.5 Final assembly of the instrument

In the end, the instrument we developed, has three PCBs (the BBB board, the DC-DC boost converter, and the signal conditioning and measurement unit) and the resistors banks used as internal load. The resistors of these banks are expected to convert into heat power up to 300W, for this reason they need to be mounted on big heat sink. We used two cumbersome units with a global thermal resistance of 0.5°C/W.
This value means, however, at maximum power a temperature increase of $300\text{W} \cdot 0.5°\text{C/W} = 150°\text{C}$.
To provide the proper air circulation and to lower the high temperature that the heat sink can reach, a 12V-0.9A fan was added. The size of this fan is 80x80x25 mm and its status can be SW controlled. For just a PV panel $I-V$ curve tracing the fan can be turned off, but for other characterization it may be required to be on.
The arrangement of these items in a 110x350x250 mm metallic box is depicted in the picture in Figs. 2.26, 2.27, and 2.28. It is possible to see in the front panel the input and output connectors, the power on/power off button and three LEDs. These LEDs are:

- BLU LED: it is on when the system is connected to an external 12V power source.
- YELLOW/GREEN LED: this bicolor LED is used to indicate process-

ing/waiting status of the instrument.

- RED LED: it is usually used to signal an error condition.

Figure 2.26: The first prototype of the system

2.6 Performance evaluation

In this section the expected error for voltage and current measurements is evaluated. To evaluate the accuracy of these measurements, the accuracy of the ADC chip (AD7714Y), that of the reference voltage provided to the ADC chip, and, in the end, the accuracy of the conditioning circuits must be taken into account.

The accuracy of the ADC chip can be obtained by Table 2.10 considering that the first filter notch frequency is usually set to 50Hz. The effective resolution is in this case 19 bits, that means a relative error 0.0002% that is

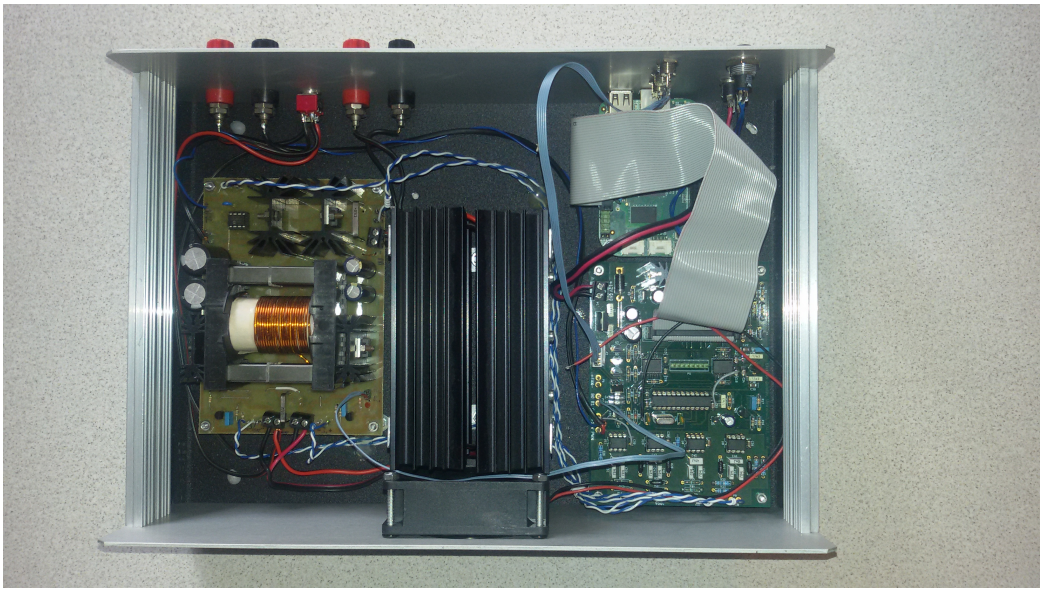


Figure 2.27: Top view of the system

Figure 2.28: Back view of the system

a very low error. This value, however, is meaningful only if the reference voltage provided to the ADC chip is error-free.

In the real world this voltage is provided by a dedicated chip. In our system, the Analog Device AD1580 chip is used, that exhibits an initial accuracy of $\pm 0.1\%$. Because this accuracy is much lower than that of the ADC chip, the accuracy of the voltage measurements of the AD7714Y can be estimated to be about $\pm 0.1\%$.

The error introduced by the conditioning circuits depends on the accuracy of the intrumentation amplifier chip AD623 and on that of the external resistive network. The first one is provided by the manufacturer of the chip and is $\pm 0.10\%$ when the gain $G = 1$, and $\pm 0.35\%$ when $G > 1$. In each case it is greater than that introduced by the tolerance of the resistors (0.1%) employed in the external network.

For the current measurements, moreover, the accuracy of the shunt resistor must be taken into account. It is $\pm 1.0\%$.

Taking into the account all these elements, since the gain G is 1 for voltage measurements, and $G > 0$ for the current measurements, we can assume that the accuracy for voltage measurements is about $\pm(0.1\% + 0.1\%) = \pm 0.2\%$, and $\pm(0.1\% + 0.35\% + 1.0\%) = \pm 1.45\%$ for the current ones.

Because the maximum input voltage is 63V for V_{in}, and 250V for V_{out}, the maximum absolute error is 0.126V for V_{in} and 0.50V for V_{out}.

The maximum input current is 10A for both I_{in} and I_{out}, therefore their maximum absolute error is 0.145A.

These values are collected in Table 2.14.

Table 2.14: Measurements Accuracy

Channel	accuracy	Error max.
V_{in}	0.2%	0.126V
I_{in}	1.45%	0.145A
V_{out}	0.2%	0.50V
I_{out}	1.45%	0.145A

Below there are some measurements obtained with in input a 19V DC laptop power supplier and the PWM set to have about 3A sunk from it. In this condition, the digital multimeter Fluke 115 provided the following values: $V = 19.36$V and $I = 3.07$A. Taking into account that this instrument exhibits an accuracy of 0.5% for the DC voltage measurements and 1% for the DC current ones, it is possible to say that the measurements obtained with the two instruments are consistent.

From these values it is possible to have an estimation of the precision of our instruments: it is about 1mV for the voltage measurements and 1mA the those of current.

```
Vin = 19.2995, Iin = 3.0988
Vin = 19.2954, Iin = 3.0987
Vin = 19.2962, Iin = 3.0990
Vin = 19.2981, Iin = 3.0993
Vin = 19.2950, Iin = 3.0988
Vin = 19.2975, Iin = 3.0993
Vin = 19.2962, Iin = 3.1004
Vin = 19.2968, Iin = 3.0997
Vin = 19.3017, Iin = 3.0998
Vin = 19.2981, Iin = 3.0995
Vin = 19.2973, Iin = 3.0996
Vin = 19.2968, Iin = 3.0998
Vin = 19.3016, Iin = 3.1000
```

```
Vin = 19.2975, Iin = 3.1001
Vin = 19.2940, Iin = 3.1006
Vin = 19.2977, Iin = 3.1003
Vin = 19.2959, Iin = 3.1004
Vin = 19.2976, Iin = 3.1005
Vin = 19.2994, Iin = 3.1008
Vin = 19.2977, Iin = 3.1017
Vin = 19.2970, Iin = 3.1012
Vin = 19.2968, Iin = 3.1007
Vin = 19.3028, Iin = 3.1007
Vin = 19.3005, Iin = 3.1012
Vin = 19.3001, Iin = 3.1012
Vin = 19.2986, Iin = 3.1008
```

2.7 Tracing a PV panel characteristic curves

In this section, some characteristic curves, taken with the system described above, are given. The source code of the little application (tracer.c) used to collect the samples is reported in Appendix A. The plots have been obtained processing off-line the collected data by means of the command line free application "gnuplot".

The first characterization, depicted in Fig. 2.29, is related to a PV panel with $V_{oc} = 22$V, $I_{sc} = 5$A. In place of a real PV panel, the PV panel simulator Elgar TerraSAS series, model ETS60X14-C-3.1.4KP was used. This instrument is able to simulate high power PV solar array. Unfortunately it does not implement the one-diode model to describe the PV panel, but just a simpler model where the opens circuit voltage, the short circuit current and

the MPP are required.

The plots have been obtained getting 100 samples with $R_{LOAD} = 28.2\Omega$. Besides the PV panel related plots, in Fig. 2.30 there is a characterization of a 12V-4Ah lead-acid battery. In this case, on the X-axis there is the duty cycle D, on the Y-axis both the voltage and the current given by the battery as a function of D. When $D = 0$ the load is disconnected, therefore it is actually an open circuit measurement, but when $D > 0$ the load is connected. This is the reason of the step when moving from $D = 0$ to $D > 0$.

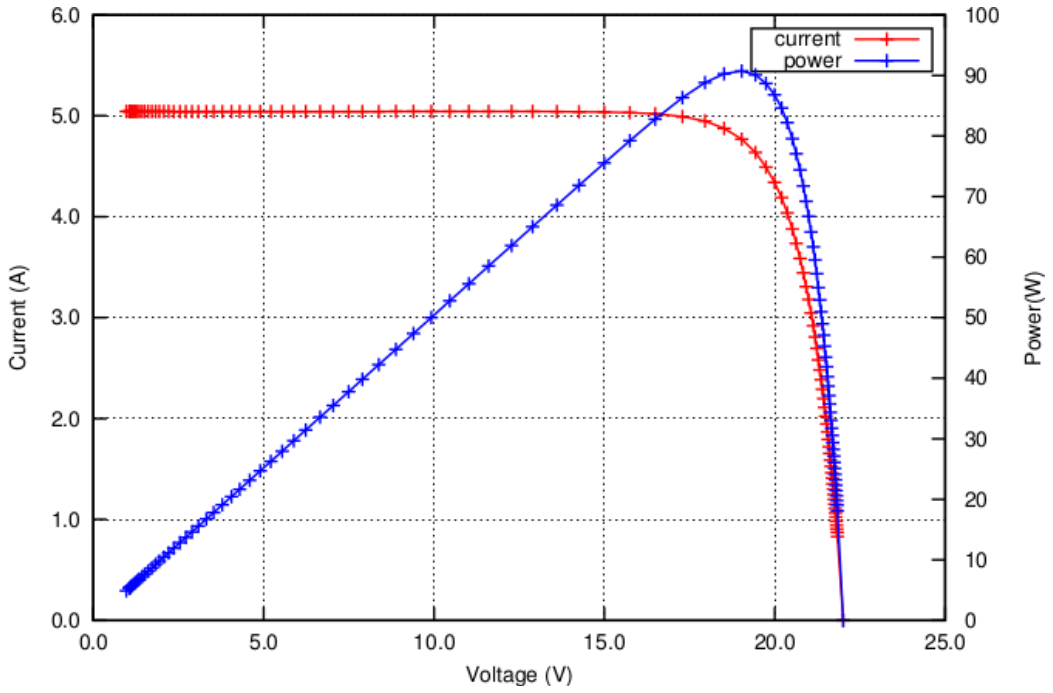


Figure 2.29: Characterization of a PV panel with $V_{oc} = 22$V and $I_{sc} = 5$A

In Fig. 2.31 Fig. 2.32 there are the characterizations of a laptop power supplier rated 19V, 4A, the first one has been obtained with $R_{LOAD} = 28.2\Omega$,

whereas, the other one, with $R_{LOAD} = 282\Omega$.

In Fig. 2.33 there is that one of two of these power suppliers connected in series. In each case, for safety reason, the tracing was stopped when the current reached the value of 3.5A.

In all these cases, it is possible to see how clean the plots are, even when the current is very high (8A).

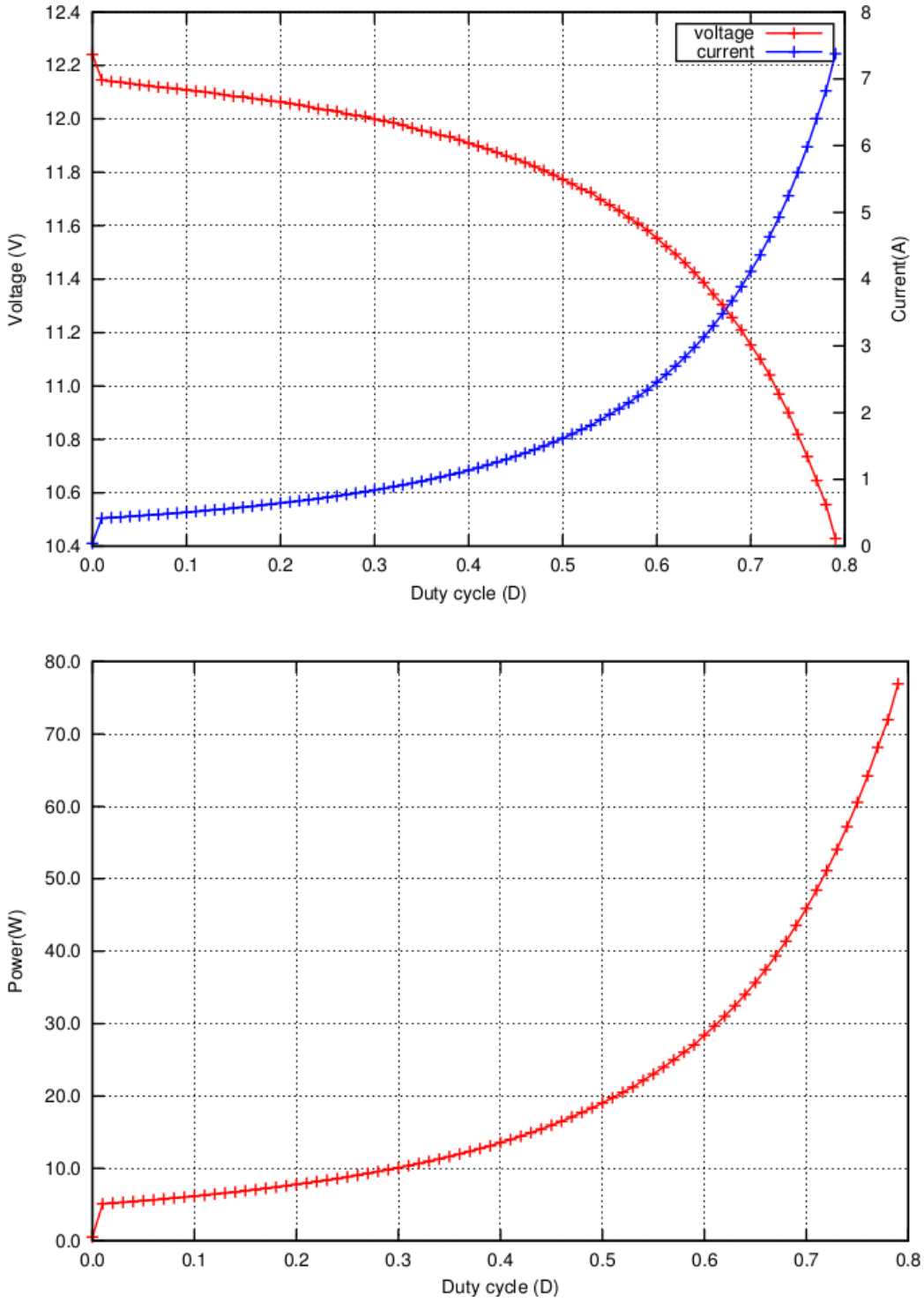


Figure 2.30: 12V-4Ah lead-acid battery characterization

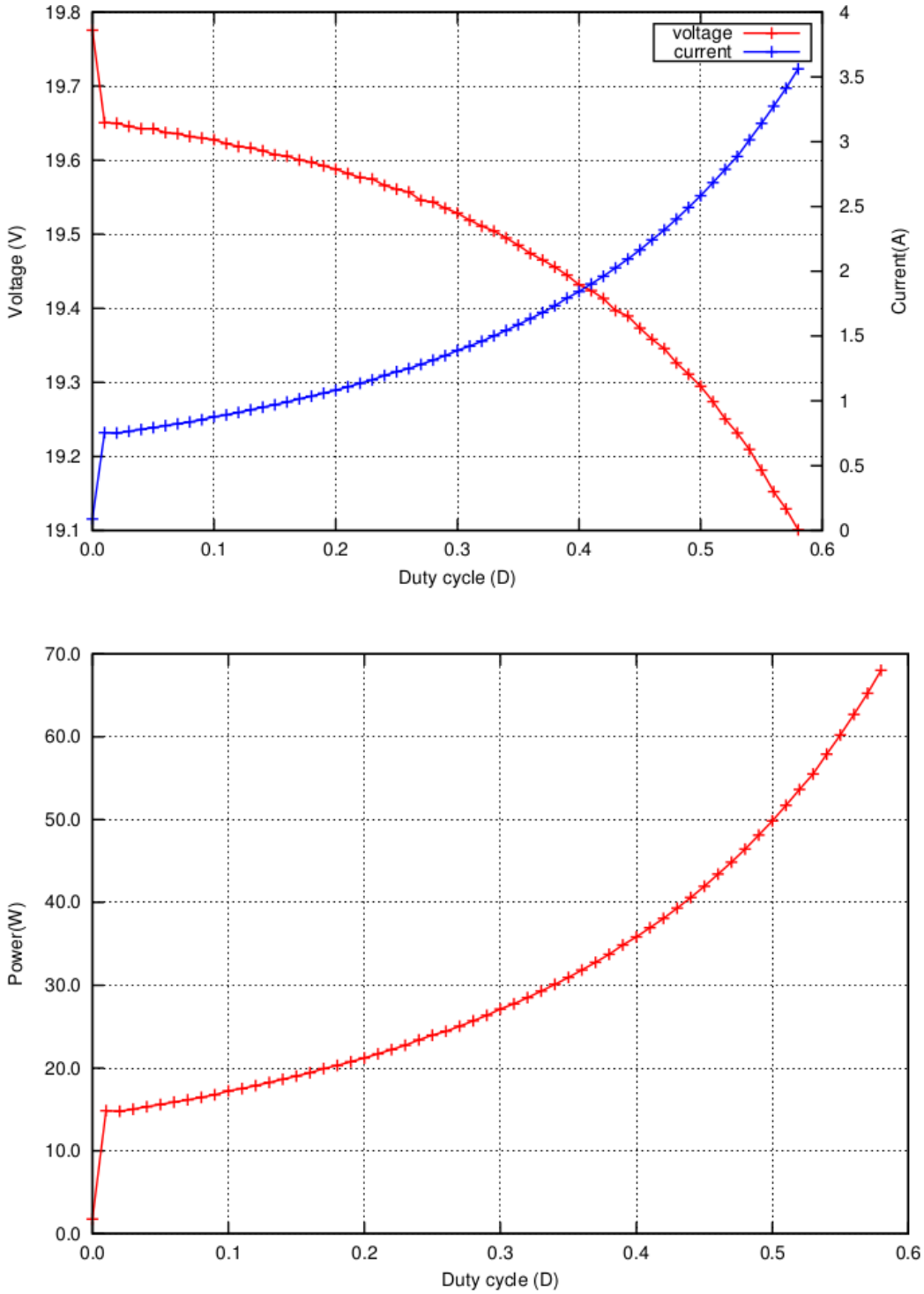


Figure 2.31: Characterization of a 19V-4A laptop power supplier. $R_{LOAD} = 28.2\Omega$

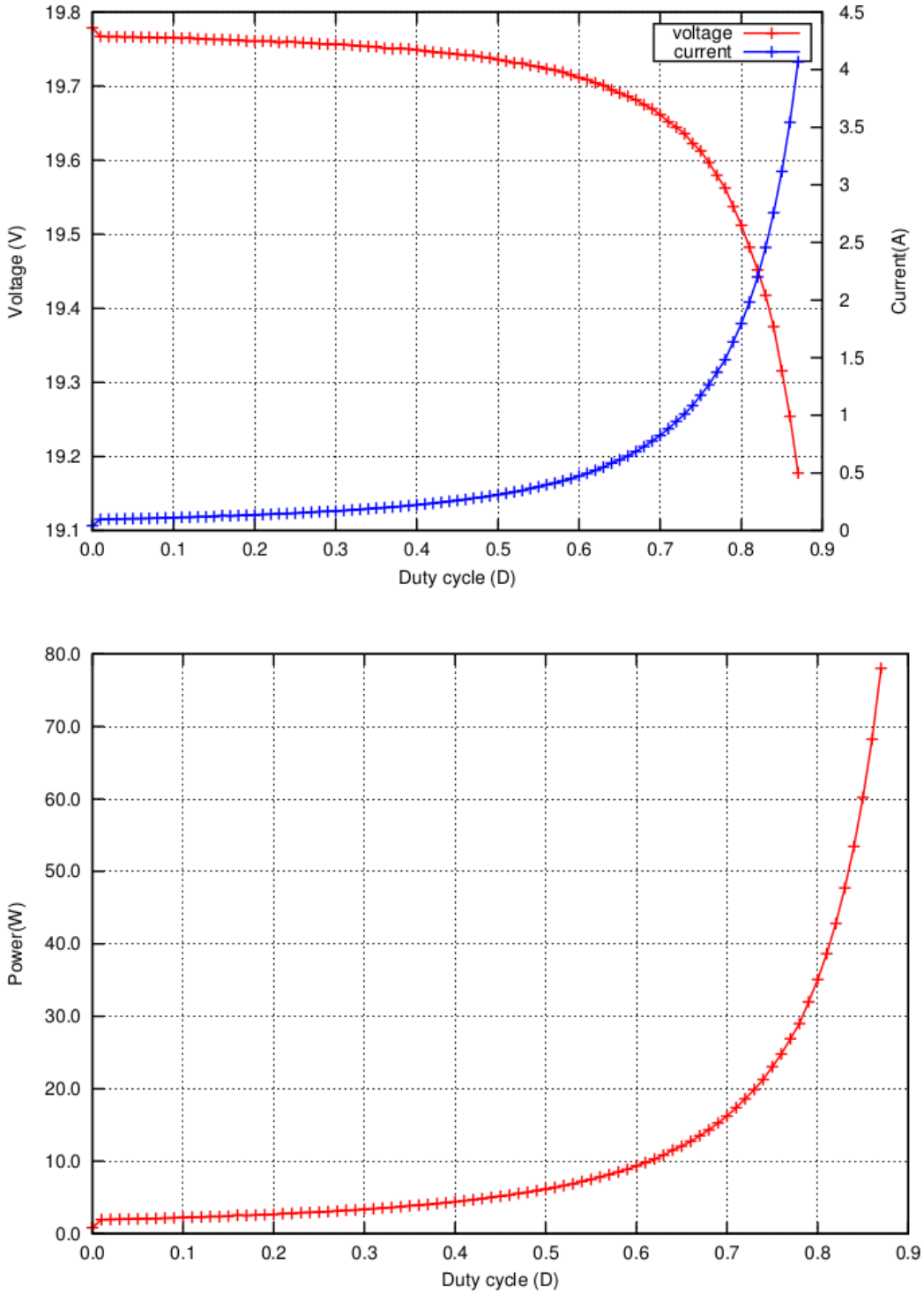


Figure 2.32: Characterization of a 19V-4A laptop power supplier. $R_{LOAD} = 282\Omega$

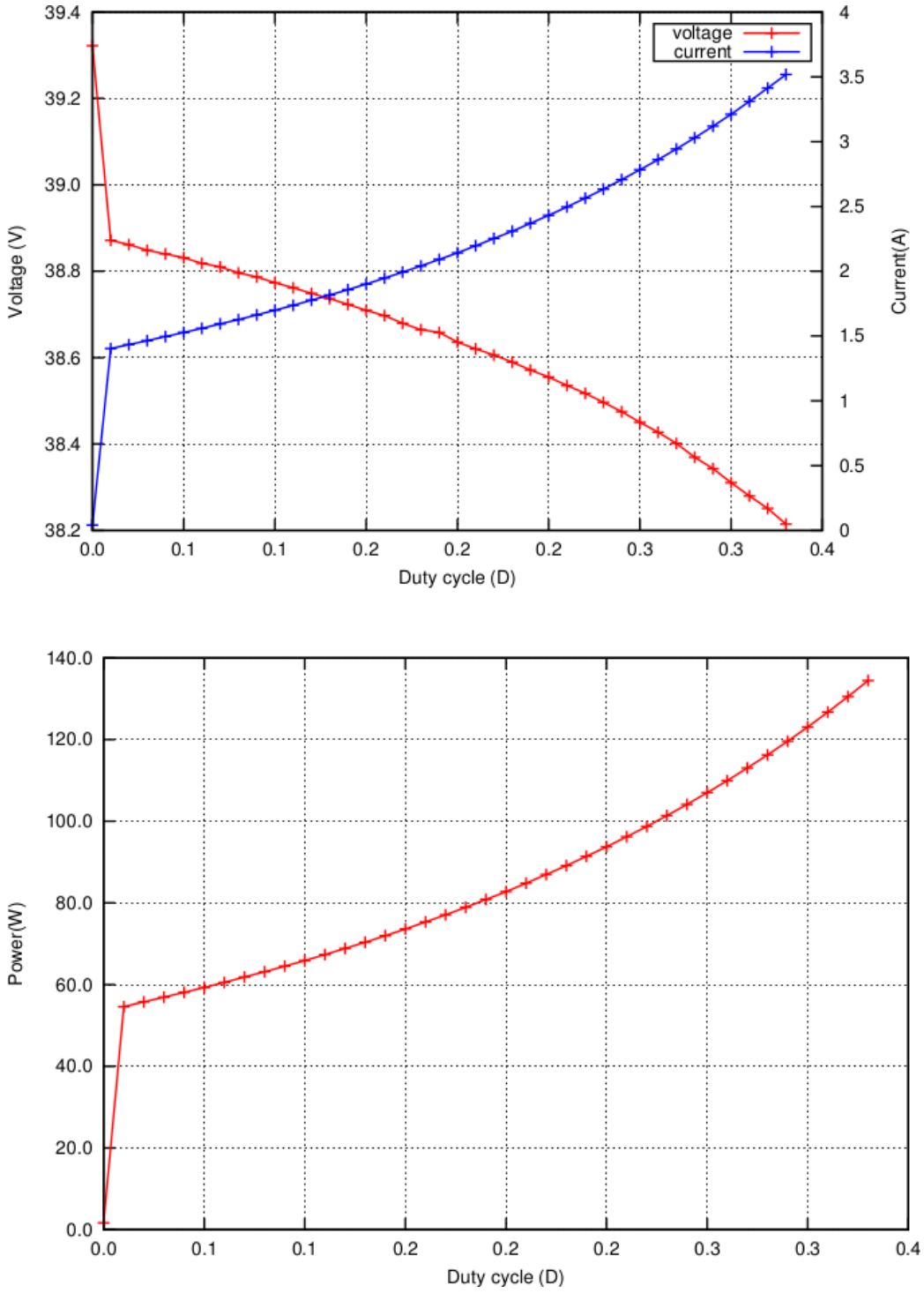


Figure 2.33: Characterization of two series connected 19V-4A laptop power suppliers

Chapter 3

One diode model parameters estimations

In the last years, due to the fast and strong growth of the photovoltaic energy market, the need of effective tools for the estimation of the electrical power produced by PV plants has become more and more important. These tools make use of PV panel models in order to get the power obtained knowing temperature and irradiance. These can be predicted by means of statistical models.

The most widely used PV model is the five parameters "one-diode model" due to its low complexity and good accuracy ([Tian et al., 2012, De Soto et al., 2006]). The estimation of its five parameters, however, is not a trivial task. The extraction of the five-parameters model has been widely discussed in literature by following a twofold approach: i) several authors proposed different equations/approaches to be used for the calculation of the five parameters from information provided by manufacturers in datasheets [De Soto et al.,

2006, Tian et al., 2012, Rajasekar et al., 2013, Laudani et al., 2013b, Dobos, 2012, Brano et al., 2010, Orioli and Di Gangi, 2013, Chen et al., 2011, Chouder et al., 2012]; *ii*) many kinds of optimization techniques have been proposed to solve the inverse problem related to the extraction of the five parameters from experimental $I - V$ curves. These latter approaches in general perform fitting procedures on the experimental data without taking into account any theoretical conditions such as the open circuit, short circuit, maximum power point conditions and so on. In this direction, softcomputing techniques have been also recently tested in the resolution of this kind of problems: [Jiang et al., 2013] presented an improved differential evolution algorithm for the extraction of the five parameters from both synthetic and experimental I–V data; [Ishaque et al., 2012] used penalty differential evolution method; in [Askarzadeh and Rezazadeh, 2013] and in [Askarzadeh and Rezazadeh, 2012] Askarzadeh et al. applied the Artificial Bee Colony (ABC) optimization algorithm and Harmonic Search-based algorithm to the identification of one-diode and two-diode models from experimental data; in [El-Naggar et al., 2012] the Simulated Annealing algorithm is proposed; similarly [AlHajri et al., 2012] used the pattern search algorithm and [Rajasekar et al., 2013] Bacterial Foraging one. Finally, other works make use of approximate formulae obtaining interesting results [Cubas et al., 2014, Peng et al., 2013]. In any case, the matter related to the identification of the five-parameter model of PV panels still represents an open problem involving the fields of both the mathematical modeling and the optimization algorithms. In [Laudani et al., 2014a] a fast and accurate method for obtaining the five parameters of the one diode model by starting from the experimental $I-V$ curve

of the PV panel is proposed. In particular the authors exploit the adoption of the *reduced forms* of the original five-parameter model. These reduced forms take advantage of important mathematical considerations through which the five parameters of the model are splitted in independent and dependent unknowns, in order to reduce the dimensions of the search space. Thus, the fitting of experimental data can be performed by using a restricted number of unknowns (namely two) and this provides strong benefits in terms of convergence, computational costs and execution times. In this way, the approach allows to characterize a PV panel from its measured $I-V$ curve with an accuracy never obtained before in literature, employing few steps and execution times of few milliseconds on a simple notebook computer.

3.1 The reduced forms

3.1.1 The identification of the one-diode model from $I-V$ experimental data

The well know one-diode equivalent circuit for a single PV solar cell is shown in Fig. 2.2. That circuit can be also generalized for a PV module with N_S solar cells in series by writing the $I-V$ relation as follows:

$$I = I_{irr} - I_0 \left[e^{\left(\frac{V+IR_S}{N_S n V_T}\right)} - 1 \right] - \frac{V+IR_S}{R_{SH}} \tag{3.1}$$

where $V_T = \frac{kT}{q}$, with q is absolute value of the electron charge ($q = 1.602 \times 10^{-19}$ C), k is the Boltzmann constant ($k = 1.3806503 \times 10^{-23}$ J/K and T is

the cell temperature; n is the cell ideality factor [1]; I_{irr} is the irradiance current; I_0 is the cell reverse saturation current; R_S and R_{SH} represent the cell series and shunt resistance, respectively. The knowledge of the five unknowns/parameters I_{irr}, I_0, n, R_S and R_{SH}, which give the name "five-parameter" to the model, allows tracing the $I-V$ curve for fixed temperature and irradiance values. In order to do this it is useful to employ the closed-form expression equivalent to the equation eq. (3.1), which provides the current I in function of the voltage V and of the five parameters $\theta = (I_{irr}, I_0, n, R_S, R_{SH})$, by using the Lambert W function [Corless et al., 1996]

$$I = f_I(V,\theta) = \frac{R_{SH}\left(I_{irr}+I_0\right)-V}{R_S+R_{SH}} - \frac{N_S n V_T}{R_S} W\left(\frac{I_0}{N_S n V_T}\frac{R_S R_{SH}}{R_S+R_{SH}} e^{\frac{R_{SH}}{R_S+R_{SH}}\frac{V+R_S(I_{irr}+I_0)}{N_S n V_T}}\right)$$

In an analogous way it is also possible to reformulate the equation eq. (3.1) by posing V in function of I and θ.

$$V = f_V(I,\theta) = R_{SH}\left(I_{irr}+I_0\right) - \left(R_S+R_{SH}\right) I - N_S n V_T W\left(\frac{I_0 R_{SH}}{N_S n V_T} e^{\frac{R_{SH}(I_{irr}+I_0-I)}{N_S n V_T}}\right)$$

The matter of the five-parameter model identification from $I-V$ experimental curves basically consists on a nonlinear least squares problem. Indeed, given a set of N couples (I_n, V_n), with $n = 1...N$, of measured current and

[1] Some authors use to write the equation 3.1 by using the coefficient $a = N_S n$ instead of the product.

voltage values respectively, we want to find the vector θ, which minimize the squared error (SE) between the values of current $I(V_n) = f_I(V_n, \theta)$, calculated by means of eq. (**??**) in correspondence to the voltages V_n and the values of measured current I_n:

$$SE = \sum_{n=1}^{N} [I_n - f_I(V_n, \theta)]^2 \tag{3.2}$$

From a more general point of view, the functional eq. (3.2) can assume any other similar expression aimed to find the best values of the parameters I_{irr}, I_0, n, R_S and R_{SH}, making the five-parameter model able to fit with the experimental $I - V$ curve as much as possible. Clearly, the choice of the functional expressions may also lead to different results. In addition, one can also decide on making the fitting procedure to operate on power–voltage curve, instead of current-voltage one, and this in order to reduce the error on power (above all around the point of maximum power P_{max}). In literature, the following functions have been adopted:

- the root mean square error (RMSE)

$$RMSE = \sqrt{\sum_{n=1}^{N} (I_n - f_I(V_n, \theta))^2} \tag{3.3}$$

- the absolute error (AE) and Mean Absolute Error (MAE):

$$AE = \sum_{n=1}^{N} |I_n - f_I(V_n, \theta)| \tag{3.4}$$

$$MAE = \frac{1}{N}\sum_{n=1}^{N}|I_n - f_I(V_n,\theta)| \tag{3.5}$$

- the weighted RMSE proposed in [Cubas et al., 2014]:

$$\xi = \frac{1}{I_{SC}}\sqrt{\frac{1}{N}\sum_{n=1}^{N}[f_I(V_n,\theta) - I_n]^2} \tag{3.6}$$

The identification of the one-diode model from $I-V$ experimental data by using a five-equation system is a *nonconvex* optimization problem and the extensive use of different numerical techniques in literature is justified by the difficulty to face with this kind of problem due to both numerical and theoretical issues. In particular, the extraction of the five parameters from experimental $I-V$ curves is a very hard multimodal problem to such an extent that it could be also considered as a real benchmark for any kind of optimization technique. Thus, the presence of several local minima (multimodal problem) prevents utilizing directly deterministic algorithms since they are strongly sensitive to the initial guesses (i.e. the starting values of the five parameters). Indeed, by changing them, the algorithm could remain trapped in a different local minimum and then, return a different solution. With the aim to detect the global optimum (i.e. the best solution), avoiding to wind up into local minima, several heuristic/stochastic algorithms coming from the Artificial Intelligence and Evolutionary Computation have been recently developed [Fulginei et al., 2012, Laudani et al., 2013a, Fulginei and Salvini, 2010, Laudani et al., 2013c]. They are effective for finding the best solution within the multimodal optimization but their convergence times and compu-

tational costs are considerably onerous together with, in general, a not very high degree of accuracy with respect deterministic algorithms. As previously stated in the Introduction section, many kinds of optimization techniques have been proposed to solve the inverse problem related to the extraction of the five parameters from experimental $I - V$ curves. Nevertheless, they are affected by the above restrictions on finding solutions within a *nonconvex* optimization problem. Thus it is clear that, with the aim to obtain better solutions than the ones proposed in literature, the use of deterministic algorithms is needed, since they are very fast and accurate.

3.1.2 The equivalent reduced forms of the five-parameter model

The first reduced form of five-parameter model was introduced by [Laudani et al., 2014c] in order to efficiently solve the problem of identification of five-parameter model from information provided by manufacturers on data sheets. Indeed, usually the PV panel manufacturers give information about some representative points at standard reference conditions (SRC): short circuit (SC) current ($I_{SC,ref}$), open circuit (OC) voltage ($V_{OC,ref}$) conditions and maximum power point (MPP) current and voltage ($I_{MPP,ref}$ and $V_{MPP,ref}$). From these three points it is possible to find the first four equations commonly used to solve a non linear system with five unknowns, as done for example by [De Soto et al., 2006, Tian et al., 2012, Rajasekar et al., 2013, Laudani et al., 2014c, Dobos, 2012]. Similarly, the same points and equations can be utilized under general conditions of temperature T and irradiance G. With the aim to simplify the notation and the expressions, the shunt conductance

$G_{\mathrm{SH}} = R_{SH}^{-1}$ will be used instead of R_{SH}, and the following quantities are defined:

$$EXP_{OC} = e^{\frac{V_{OC}}{N_S n V_T}};\ EXP_{MPP} = e^{\frac{V_{MPP}+R_S I_{MPP}}{N_S n V_T}};\ EXP_{SC} = e^{\frac{I_{SC}}{N_S n V_T}}; \quad (3.7)$$

The first three equations of the model are obtained by evaluating eq. (3.1) at the open–circuit ($V = V_{OC}$, $I = 0$), short–circuit ($V = 0$, $I = I_{SC}$) and maximum power ($V = V_{MPP}$, $I = I_{MPP}$) points, respectively, whereas the fourth equation is found by imposing the derivative of power, with respect to the voltage, equal to zero at the maximum power point:

$$0 = I_{irr} - I_0 \left(EXP_{OC} - 1\right) - G_{SH} V_{OC} \quad (3.8)$$

$$I_{SC} = I_{irr} - I_0 \left(EXP_{SC} - 1\right) - G_{SH} R_S I_{SC} \quad (3.9)$$

$$I_{MPP} = I_{irr} - I_0 \left(EXP_{MPP} - 1\right) - G_{SH} \left(V_{MPP} + R_S I_{MPP}\right) \quad (3.10)$$

$$\frac{I_{MPP}}{V_{MPP}} = \frac{\frac{I_0}{N_S n V_T} EXP_{MPP} + G_{SH}}{1 + \frac{I_0 R_S}{N_S n V_T} EXP_{MPP} + G_{SH} R_S} \quad (3.11)$$

From the above relations, two different reduced forms can be found in which n and R_S are the only unknowns (*independent unknowns*) that allow to directly evaluate the other three parameters I_0, G_{SH} and I_{irr} (*dependent unknowns*). These two reduced forms can be successfully used to solve the problem of the PV panel characterization from experimental $I - V$ curves. The first of the two reduced forms has been already used to solve efficiently the problem of extraction of the five parameters from data sheet information

by [Laudani et al., 2014c], whereas the second is presented in [Laudani et al., 2014a].

3.1.3 First reduced form (*RF #1*)

The RF #1 can be obtained by means of some algebraic manipulations on the previous equations eq. (3.8), eq. (3.10) and eq. (3.11). Further detail about this derivation can be found in [Laudani et al., 2014a]. After these manipulations it is possible to obtain the following equations:

$$G_{SH} = \frac{I_{MPP}\left[\left(V_{MPP} + N_S n V_T - I_{MPP} R_S\right) EXP_{MPP} - N_S n V_T EXP_{OC}\right]}{\left(R_S I_{MPP} - V_{MPP}\right)\left[EXP_{MPP}\left(I_{MPP} R_S + V_{MPP} - V_{OC} - N_S n V_T\right) + EXP_{OC} N_S n V_T\right]} \tag{3.12}$$

$$I_0 = \frac{I_{MPP}\left(V_{OC} - 2V_{MPP}\right) N_S n V_T}{\left(R_S I_{MPP} - V_{MPP}\right)\left[EXP_{MPP}\left(I_{MPP} R_S + V_{MPP} - V_{OC} - N_S n V_T\right) + EXP_{OC} N_S n V_T\right]} \tag{3.13}$$

$$I_{irr} = \frac{I_{MPP}\left[V_{OC}\left(V_{MPP} - I_{MPP} R_S + N_S N V_T\right) EXP_{MPP} + N_S n V_T\left(2V_{MPP}\left(1 - EXP_{OC}\right) - V_{OC}\right)\right]}{\left(R_S I_{MPP} - V_{MPP}\right)\left[EXP_{MPP}\left(I_{MPP} R_S + V_{MPP} - V_{OC} - N_S n V_T\right) + EXP_{OC} N_S n V_T\right]} \tag{3.14}$$

In these expressions G_{SH}, I_0 and I_{irr} *explicitly* are function of n and R_S.

3.1.4 Second reduced form (*RF #2*)

In a very similar way, but using this time the equations eq. (3.8), eq. (3.9) and eq. (3.10), it is possible to find the following expressions of G_{SH}, I_0 and I_{irr} *explicitly* in function of n and R_S:

$$G_{SH} = \frac{EXP_{OC}\left(I_{MPP} - I_{SC}\right) + EXP_{MPP}I_{SC} - EXP_{SC}I_{MPP}}{A_1 EXP_{SC} + A_2 EXP_{MPP} + A_3 EXP_{OC}} \tag{3.15}$$

$$I_0 = \frac{V_{OC}\left(I_{SC} - I_{MPP}\right) - V_{MPP}I_{SC}}{A_1 EXP_{SC} + A_2 EXP_{MPP} + A_3 EXP_{OC}} \tag{3.16}$$

$$I_{irr} = \frac{I_{SC}V_{OC}\left(EXP_{MP} - 1\right) + I_{SC}V_{MPP}\left(1 - EXP_{OC}\right) + I_{MPP}V_{OC}\left(1 - EXP_{SC}\right)}{A_1 EXP_{SC} + A_2 EXP_{MPP} + A_3 EXP_{OC}} \tag{3.17}$$

where the expressions for A_1, A_2 and A_3 are:

$$A_1 = V_{MPP} + R_S I_{MPP} - V_{OC} \tag{3.18}$$

$$A_2 = V_{OC} - R_S I_{SC} \tag{3.19}$$

$$A_3 = R_S I_{SC} - R_S I_{MPP} - V_{MPP} \tag{3.20}$$

3.1.5 Domain of the search space and initial guesses

The previously presented two reduced forms also allow to gain insight about the domain of the search space of the remaining two unknown parameters (n and R_S). Indeed, it is clear that the values of the dependent parameters must be physically significant, that is their expressions must return positive values. As a result, not all values of n and R_S can be accepted as solutions and then, by exploiting the reduced form *RF #1*, it is possible to write a relation which determines the maximum values of R_S, called R_S^{max}, in function of n [Laudani et al., 2013a]:

$$R_S^{\max}(n) = \frac{N_S n V_T}{I_{MPP}}\left[1 + W\left(-e^{\frac{V_{OC} - 2V_{MPP} - N_S n V_T}{N_S n V_T}}\right)\right] + \frac{V_{MPP}}{I_{MPP}} \tag{3.21}$$

and conseguently the feasible range for R_S is:

$$0 < R_S < R_S^{\max}(n) \tag{3.22}$$

Regarding the parameter n, the values within the range specified in eq. (3.23) are well accepted in literature:

$$0.5 \leqslant n \leqslant 2.5 \tag{3.23}$$

Therefore, the previous expressions eq. (3.22)-eq. (3.23) allow to identify a well defined domain in which you can choose very general and optimal initial values (initial guesses) of the two independent parameters that are the following:

$$\begin{cases} n = 1.0 \\ R_S = 0.9 R_S^{\max}(1.0) \end{cases} \tag{3.24}$$

where the coefficient 0.9 comes from the observation that, in all the final solutions, the values of R_S prove to be very close to the $R_S^{\max}$ one.

3.2 Fitting experimental $I-V$ curve by using reduced forms

As previously stated, the first reduced form has been successfully used to solve the matter of extraction of the five parameters from data sheet information by [Laudani et al., 2013a]: its effectiveness is due to the reduction of

the number of unknowns with the consequent possibility of choosing optimal initial values for the two unknown parameters within a specific feasible domain. In a similar way, the two reduced forms can be used for decreasing the number of unknowns of the minimization problem by exploiting the three expressions of I_0, I_L and R_{SH} that can be written as functions of the remaining two (n, R_S). In this way, by using the reduced forms, the least square problem is solved for the vector of the only two independent parameters $\vartheta = (n, R_S)$. In order to obtain the original five-parameters $\theta = (\vartheta, I_0(\vartheta), I_{irr}(\vartheta), R_{SH}(\vartheta))$, the values of the remaining parameters I_0, I_{irr}, R_{SH} are computed by using eq. (3.12), eq. (3.13) and eq. (3.14) if one uses the reduced form *RF #1*, or eq. (3.15), eq. (3.16) and eq. (3.17) if one uses the reduced form *RF #2*. As a consequence, for example, the squared error eq. (3.2) previously seen becomes:

$$SE = \sum_{n=1}^{N} \left[I_n - f_I(V_n, \vartheta, I_0(\vartheta), I_{irr}(\vartheta), R_{SH}(\vartheta)) \right]^2 \tag{3.25}$$

Using the initial guesses for n and R_S described above, and making use of deterministic optimization algorithm like the well-known Levenberg–Marquadt one, it is possible to easily get a solution. Since the search space is limited to two unknowns, the optimization algorithm provides a solution very quickly. The Authors of [Laudani et al., 2014a] underline that the solution is unique because of the constraints imposed by the reduced forms and the initial guesses. It does not mean, however, the solution is unique.

3.3 Using the reduced forms in the real world

In the previous sections it has been shown that the method proposed in [Laudani et al., 2014a] is the most effective one due to its speed and accuracy. Because of these characteristics, its adoption in an embedded system, like the proposed one, can be taken into account. An instrument that is able to execute the procedure described in [Laudani et al., 2014a] would be able to provide the five parameters of the one-diode model directly, without needing any off-line post-processing. This behavior, at the best of our knowledge, is not available in any commercial or research prototype device.

The adoption of this method in an embedded system, however, is a tricky task. First of all, even if the method is computationally very efficient, the system should be powerful enough. The control board in our system, with a CPU running with a clock frequency of 1GHz, should be powerful enough to provide the five parameters in a very short time.

Another issue in an embedded system is that it is not easy to write complex SW applications. Since the BBB (or the BBG) has a complete Linux operating system, the SW routines can be easily written in high level programming language, like C, as if it were written for a common personal computer.

When dealing with mathematical commonly used routines, like the optimization algorithms required by the method in [Laudani et al., 2014a], it is desirable to make use of already written mathematical libraries. These libraries provide accurate, robust and efficient implementations that can be conveniently used by a SW programmer. Our need is to find a mathematical library that can be cross-compiled for the ARM architecture. Since this architecture is usually widely supported, also this need seems it is possible

to be satisfied.

3.3.1 SW preparation

Since the adoption of the procedure described in [Laudani et al., 2014a] in our system seems to be feasible, a SW simulator, running on a Linux personal computer, was written in order to evaluate and optimize the performance of our system when performing a PV panel characterization with this procedure. The first issue that requires to be carefully studied is the propagation of the $I-V$ data measurement errors in the five parameters of the one diode model. Since two reduced forms are proposed in [Laudani et al., 2014a], they will be both evaluated, with respect to the measurement errors, in order to select the most suitable one.
The simulator was written with the aim to be the same SW that will be executed in the control board of the system, with the only obvious difference that input data are generated in the SW simulator, whereas, in the real system, are given by the measurement data of voltage and current taken as described in Section 2.7.

Input data

In the SW simulator input data are generated starting from the five parameters of the one-diode model of a PV panel. The SW simulator is a command line application and the five parameters are given through the command line itself. These five parameters are generated starting from the data sheet in-

formation as described in [Laudani et al., 2014c].
In order to have the points belonging to the curve $I-V$ as equally spaced as possible, the following technique has been developed. First of all, the desired number $nsam$ of samples is taken simply changing linearly the duty cycle D from $D=0$ to $D=1$.
After that, the following approximation of the length of the $I-V$ curve is evaluated:

$$L_{IV} = \sum_{n=1}^{nsam-1} \left(\sqrt{(v_j - v_{j-1})^2 + (i_j - i_{j-1})^2} \right) \tag{3.26}$$

After having evaluated L_{IV}, the $nsam$ samples are taken again, but now with D determined in order to have between two consecutive samples the distance $\sqrt{(v_j - v_{j-1})^2 + (i_j - i_{j-1})^2}$ equal to $L_{IV}/nsam$.
This is obtained by searching in the vector of the first $nsam$ acquisition the nearest sample to the desired one and performing a linear interpolation to estimate the required duty cycle.

Starting from the five parameters, the SW simulator can easily generate the wanted number $nsam$ of data points. It also adds additive white Gaussian noise, with a desired power σ^2, to simulate the measurement error.
Adding white Gaussian noise with power σ^2 means that the measurement error has a probability of about 99.99% to be in the range $[-3\sigma, +3\sigma]$.
In this way it is possible to test the extraction procedure with many different PV panels, number of data points, and values of the measurement error.
The identification routine relies on the reduced forms described in 3.1.3 and 3.1.4. Beside some data points belonging to the $I-V$ curve of the panel, both of them need the knowledge of the of point of maximum power, that is

of V_{MP} and I_{MP}, but RF #1 is expected to be more sensitive with respect to the accuracy of this point since it imposes two conditions on the maximum power point: the belonging of (V_{MP} and I_{MP}) point to the I–V curve and the slope of the curve at MPP.

The MPP is identified by a SW routine that, starting from the data point of the $I - V$ curve with the maximum value of power, adjusts, through successive approximations, the value of the duty cycle used to collect this value to improve the accuracy of the MPP.

Besides these values, the RF #1 needs the knowledge of the open circuit voltage V_{oc}, whereas the second one also requires that of the short circuit current I_{sc}.

In our system, it is very easy to get V_{oc} because it is possible to disconnect the load, but the I_{sc} cannot be obtained directly. Since the $I - V$ curve is pretty linear near the short circuit region, an extrapolation through a simple linear regression technique of the values of the last four points is used to have an estimation of I_{sc}.

Implementation of the identification routine

The identification routine proposed in [Laudani et al., 2014a] is based on the reduced forms RF #1 and RF #2 and an optimization routine based on the well-known Levenberg–Marquardt (LM) deterministic algorithm ("lsqnonmin" function in Matlab). It is very easy to translate in C language the reduced forms, but to implement a good optimization routine is a tricky

task.

For this reason a the library levmar-2.6, available in [Lourakis, 2004], has been adopted. The following information have been taken from this web site, where additional information are available.

This library includes double and single precision LM C/C++ implementations, both with analytic and finite difference approximated Jacobians. It is provided free of charge, under the terms of the GNU General Public License.

The mathematical theory behind unconstrained levmar is described in detail in the lecture notes in [Madsen et al., 2004].

To deal with linear equation constraints, levmar employs variable elimination based on QR factorization, as described in ch. 15 of the book [Wright and Nocedal, 1999].

For the box-constrained case, levmar implements the algorithm proposed in [Kanzow et al., 2002] (pp. 375-397).

levmar offers several user-callable functions obeying the following naming convention: The first letter (d or s) specifies double or single precision and the suffix (_der or _dif) denotes analytic or approximate Jacobian. If present, the lec, bc and blec components imply linear equation, box and simultaneous box and linear equation constraints, respectively. More specifically, levmar includes the functions below:

Unconstrained optimization:

```
dlevmar_der():  double precision, analytic Jacobian
dlevmar_dif():  double precision, finite difference approximated Jacobian
slevmar_der():  single precision, analytic Jacobian
slevmar_dif():  single precision, finite difference approximated Jacobian
```

Constrained optimization:

```
dlevmar_lec_der():  double precision, linear equation constraints, analytic Jacobian
dlevmar_lec_dif():  double precision, linear equation constraints, finite difference approximated Jacobian
slevmar_lec_der():  single precision, linear equation constraints, analytic Jacobian
slevmar_lec_dif():  single precision, linear equation constraints, finite difference approximated Jacobian

dlevmar_bc_der():  double precision, box constraints, analytic Jacobian
dlevmar_bc_dif():  double precision, box constraints, finite difference approximated Jacobian
slevmar_bc_der():  single precision, box constraints, analytic Jacobian
slevmar_bc_dif():  single precision, box constraints, finite difference approximated Jacobian

dlevmar_blec_der():  double precision, box & linear equation constraints, analytic Jacobian
dlevmar_blec_dif():  double precision, box & linear equation constraints, finite difference approximated Jacobian
slevmar_blec_der():  single precision, box & linear equation constraints, analytic Jacobian
slevmar_blec_dif():  single precision, box & linear equation constraints, finite difference approximated Jacobian

dlevmar_bleic_der():  double precision, box, linear equation & inequality constraints, analytic Jacobian
dlevmar_bleic_dif():  double precision, box, linear equation & inequality constraints, finite difference approximated Jacobian
slevmar_bleic_der():  single precision, box, linear equation & inequality constraints, analytic Jacobian
slevmar_bleic_dif():  single precision, box, linear equation & inequality constraints, finite difference approximated Jacobian
```

Briefly, the arguments of the functions are as follows: pointers to routines evaluating the vector function f and its Jacobian (if applicable), pointers to the initial estimate of the parameter vector p and the measurement vector x, the dimension m of p, the dimension n of x, the maximum number of iterations, a pointer to a 3 element array whose elements specify certain minimization options, a pointer to an array into the elements of which various pieces of information regarding the result of minimization will be returned, a pointer to working memory, a pointer to a covariance matrix and a pointer to application-specific data structures that might be necessary for computing the function to be minimized and its derivatives.
This library has been built both for ARM processor, in order to be used in our system, and for x86 processors, in order to be used for the simulations on a Linux personal computer.

In our implementation the function `dlevmar_bc_dif` is used. The pa-

rameter "n" is boudend in the interval $[0.5, ..., 2.5]$ and R_s in $[0, ..., 100]$. Sometimes the solution provides negative values for R_{sh}. In these case the solution is rejected and another run is required.
In Appendix A it is possible to see the C source code of the file "ident5lib.c" where there is the implementation of the function:

```
int ident5pv(double *v5, double Vmpp, double Impp, double *V, double
*I, int nsam, double T, int ns, int rf_type);
```

This function is called by both the SW running on the control board (when compiled for ARM processors) and the simulator (when compiled for x86 processors). It provides through the vector "v5" the five parameters of the one-diode model found. It receives in input the values of the MPP (V_{mpp} and I_{mpp}), the $(nsam + 2)$ element vectors "I" and "V" with the $nsam$ samples and the values of I_{sc} and V_{oc}. The parameter "rf_type" allows us to select the reduced form type to be used during the identification process.

Performance evaluation

With the SW simulator described above, extensive simulations have been performed in order to evaluate the expected performances of the extraction procedure proposed in [Laudani et al., 2014a] implemented in our development system.
During these tests, the power of the noise affecting the measurements values was set to $25E^{-6}$ for the voltage measurements, and to $25E^{-6}$ for the those of

current. These values means there is the 99.99% of probability the maximum error in voltage measurements is ±0.1V, and ±0.02A for those of current.

The PV panel used in out tests is the same one used in [Laudani et al., 2014a], that is the Photowatt PWP201 working at 45°C, where the following parameters have been identified: $n = 1.317400$, $R_s = 1.239019\Omega$, $I_0 = 2.518888e-06\text{A}$, $I_{irr} = 1.032376\text{A}$, $R_{sh} = 745.644300\Omega$, $V_{oc} = 16.777003\text{V}$, $I_{sc} = 1.030659\text{A}$.

The number of samples was set to 26 and R_{LOAD} to 282Ω.

With these values, 1000 iterations were performed. During each iteration the SW simulated data acquisition, that is samples acquisition in the $I-V$ curve, MPP identification, and measurement of I_{sc} and V_{oc}. After that, the identification routine `ident5pv()` was called, both with RF #1 and RF #2 type, and the five parameters returned by this function evaluated.

The solutions with $R_{sh} < 0$ were discarded. In Table 3.1 there is the percentage of valid solutions provided by using RF #1 and RF #2. In the same table there is also the mean value of the root mean squared error (RMSE) calculated at each estimation. It is defined by the following:

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left[f_I(V_n,\theta) - f_I(V_n,\theta_0)\right]^2} \tag{3.27}$$

where f_I is defined by eq. (3.1), the five parameters θ_0 are those given in input to the simulator, θ are those coming out from the characterization process. The parameter N was set to 200.

It is interesting to note that, as expected, the RF #2 provided more accurate characterizations in presence of measurement noise. The reason of this relies

Table 3.1: Comparison of characterizations with RF #1 and RF #2

	RF #1	RF #2
valid solutions	85.6%	94.4%
$\overline{RMSE}$	0.004154	0.003027

to the less critical estimation of MPP required by RF #2.

In Fig. 3.1 and Fig. 3.2 there are the characterizations performed with the same input data, but obtained by using RF #1 and RF #2. In each figure there are three plots: the input data, the characteristic curve obtained with the five parameters in input, and the characteristic curve obtained with the estimated five parameters.

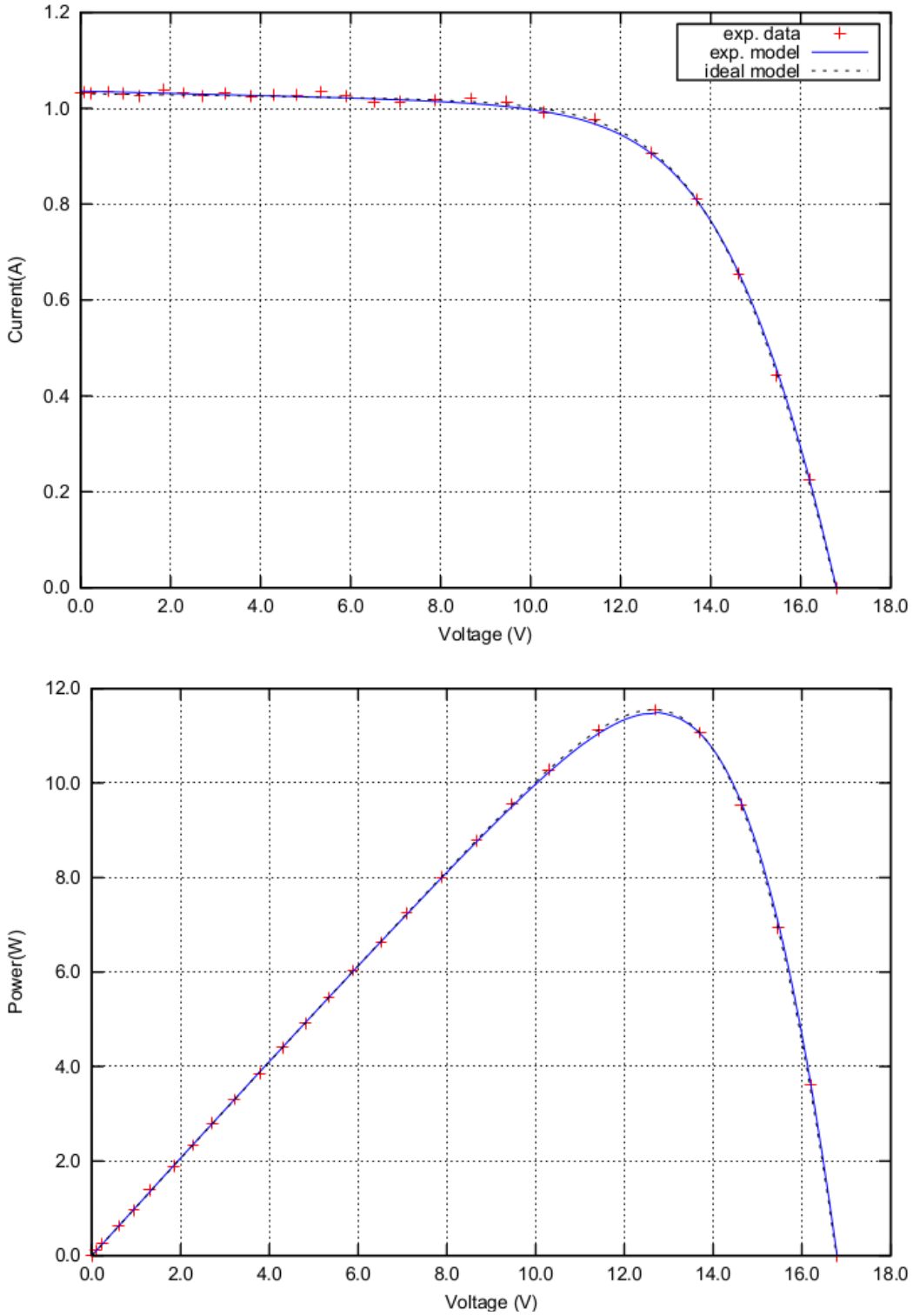


Figure 3.1: Characterization obtained with RF #1, $RMSE = 0.004675$

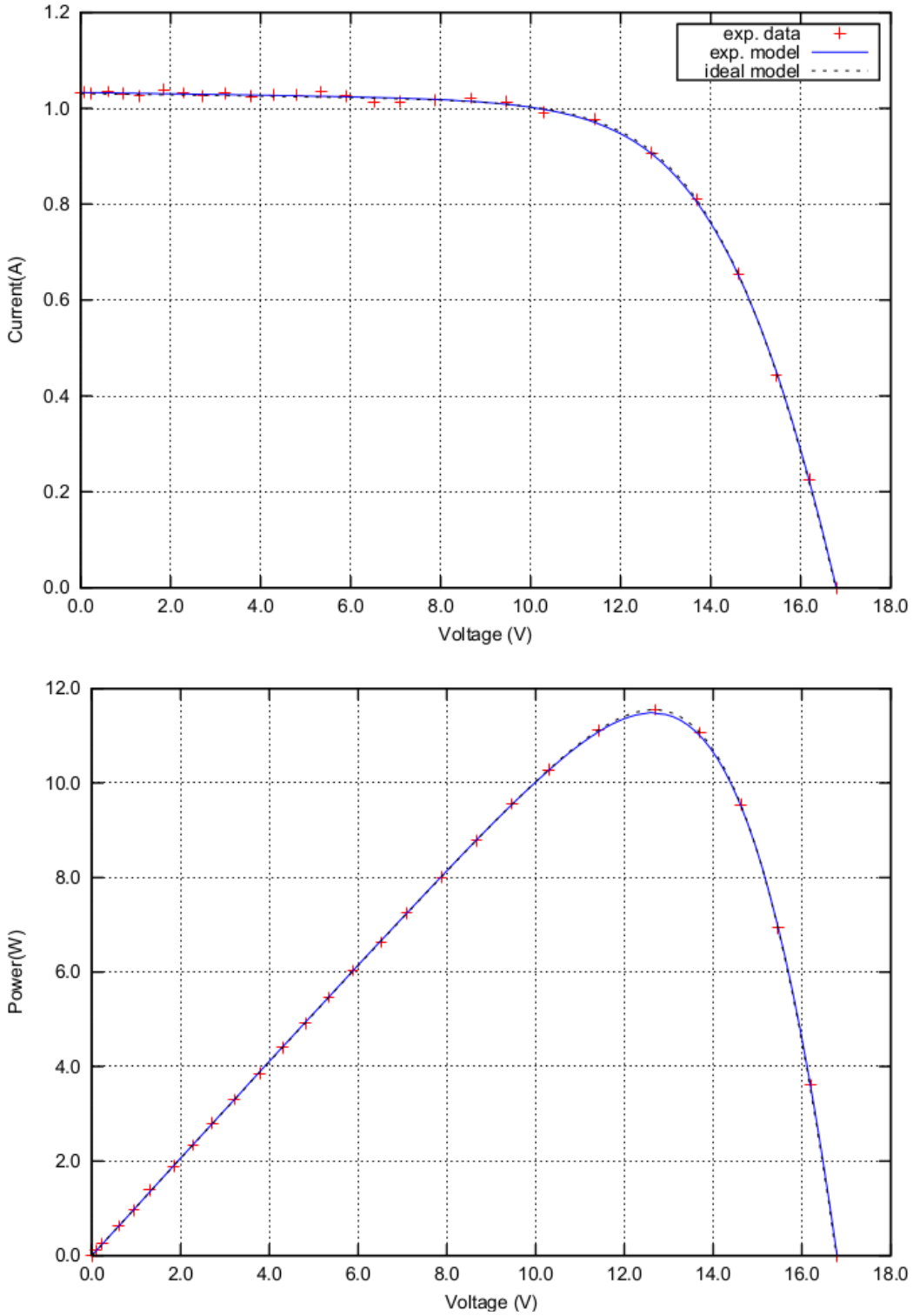


Figure 3.2: Characterization obtained with RF #2, $RMSE = 0.002901$

3.3.2 Some real PV panels characterizations

In this subsection some real characterizations obtained with our system are given. They have been obtained with the PV panels simulator Elgar ETS TerraSAS ETS60X14C. The experimental setup used for these characterizations is depicted in Fig. 3.3. Two characterizations are reported here: the first one has been obtained programming the PV simulator to act like the panel BP Solar BP585, the second one to simulate the behavior of the panel Kyocera KC200GT. The datasheet values for the former panel are in Table 3.2, whereas the values for the latter are in Table 4.1.

The characterizations have been obtained taking 50 points of the $I-V$ curve. For the panel BP585 we have set R_{LOAD} to 282Ω, whereas for the Kyocera to 28.2Ω. In both cases RF #2 provided the better results. The RMSE was 2.5E-3 for the panel BP585 and 5.0E-3 for the KC200GT.

Each characterization lasted about 38s. In Table 3.3 are collected some other significant data.

In Figs. 3.4 and 3.5 there are the characterizations for these panels. The red crosses indicate the experimental data, that is the voltages and current values provided by the acquisition unit. With these points the SW embedded into the system provided the five parameters of the one-diode model for each PV panel. With these parameters the blue continuous $I-V$ and $P-V$ curves have been plot. It is possible to see the accurateness of the data and the models provided by the instrument.

Table 3.2: Datasheet values for the PV panel BP Solar BP585

Parameter	Values @ STC
Maximum power P_{max}	85W
Short-circuit current I_{SC}	5.0A
Open-circuit voltage V_{OC}	22.1V
Maximum Power current I_{mpp}	4.72A
Maximum Power voltage V_{mpp}	18.0V
Temperature Coefficient of I_{SC} ($\alpha_{I_{SC}}$)	0.00065A/°C
Temperature Coefficient of V_{OC} ($\alpha_{V_{OC}}$)	-0.080V/°C
Number of cells	36

Table 3.3: Identification errors for the BP Solar BP585

Parameter	Datasheet values	Experimental values	error (%)
Maximum power P_{max}	85W	85.10W	0.1
Short-circuit current I_{SC}	5.0A	5.06A	1.2
Open-circuit voltage V_{OC}	22.1V	22.06V	0.2
Maximum Power current I_{mpp}	4.72A	4.66A	1.3
Maximum Power voltage V_{mpp}	18.0V	18.24V	1.3

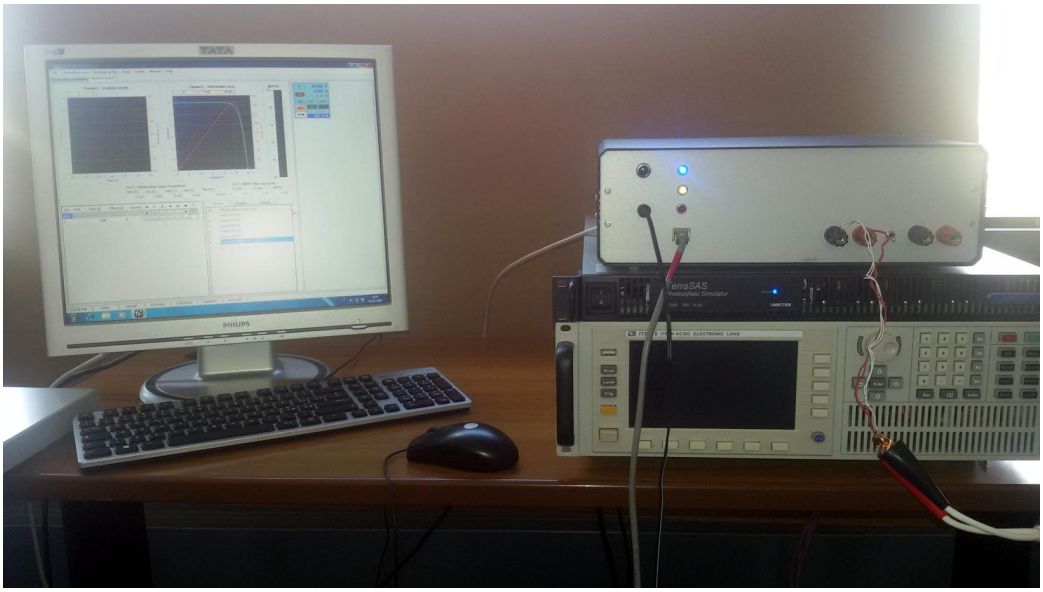


Figure 3.3: Experimental setup used to test the system. On the top there is our system, below it the PV panel simulator Elgar ETS TerraSAS ETS60X14C

.

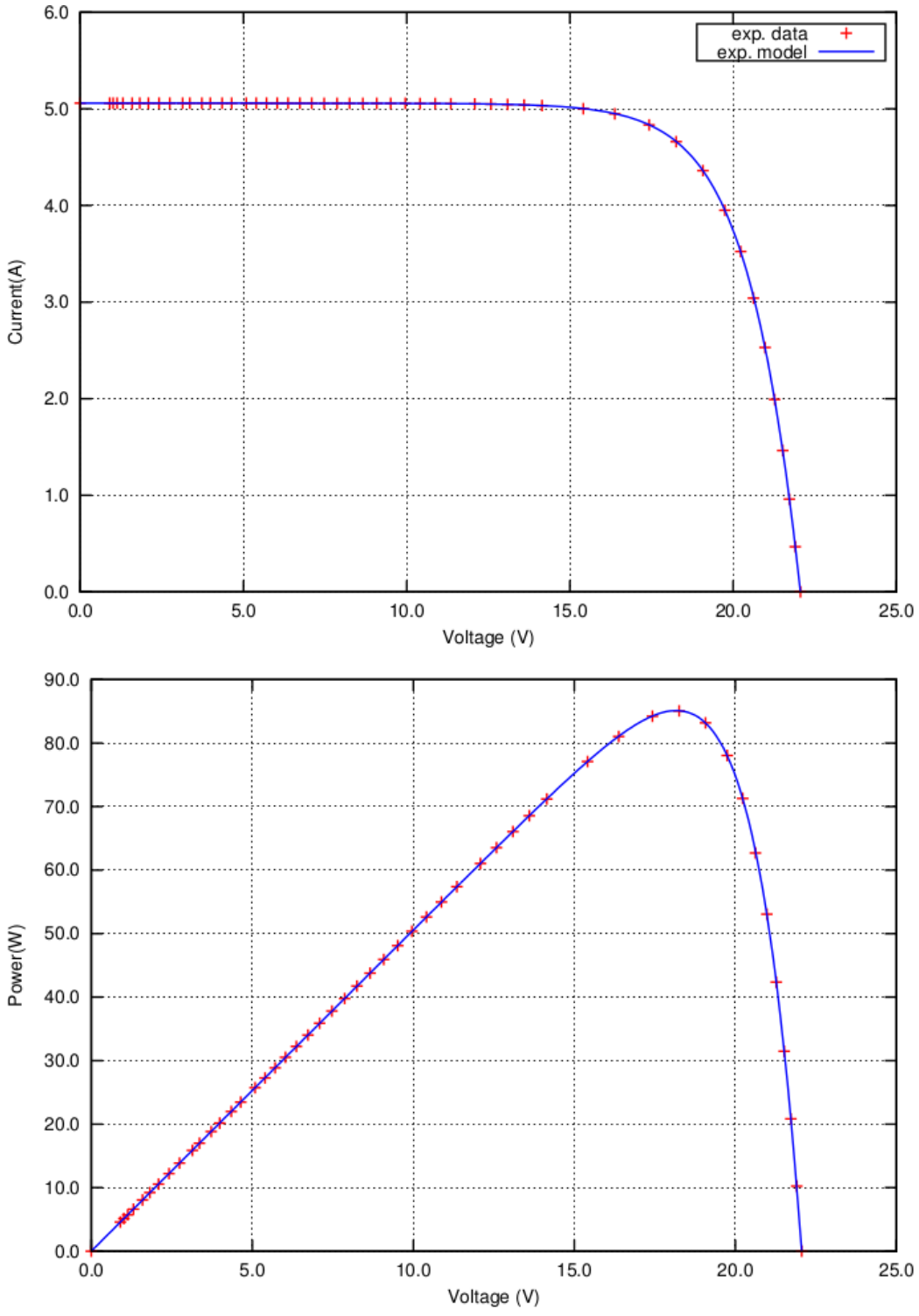


Figure 3.4: Characterization with the simulator acting like the panel BP585, RMSE=2.5E-3

.

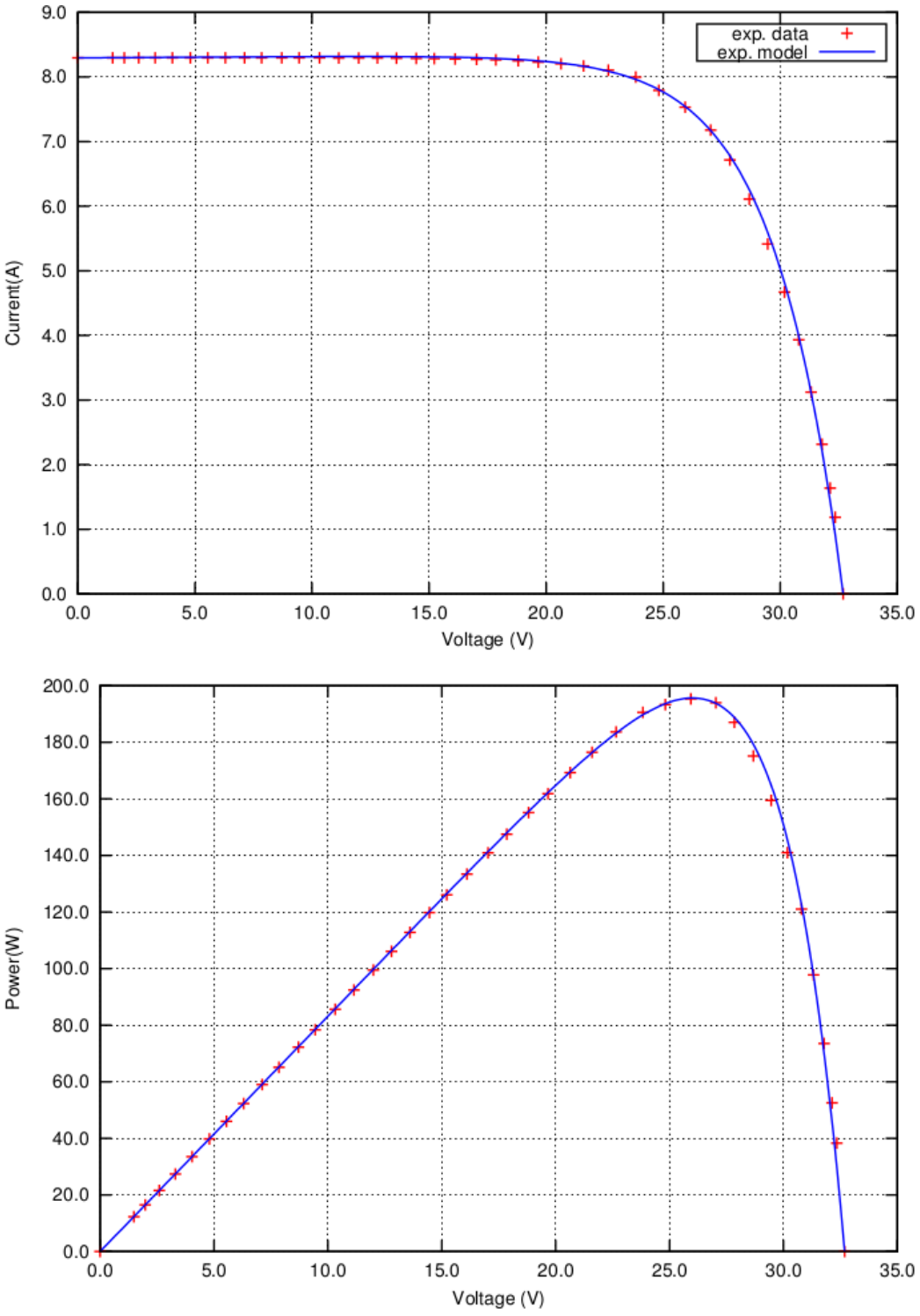


Figure 3.5: Characterization with the simulator acting like the panel KC200GT, RMSE=5.0E-3

.

Chapter 4

Additional use cases for the proposed system

4.1 DC-DC converter control algorithm verification

Putting the load switch in the 'external load' positions, it is possible to connect the output of the DC-DC converter to an external load, for example to a storage battery. In this case, the SW running in the microcomputer board could have the task to control the DC-DC converter according to a power control algorithm that it is required to investigate.

4.1.1 DC-DC converter enhancement

For PV panel characterization a DC-DC boost converter has been used as electronic load. When the instrument is used to study the control algorithm

of a DC-DC converter, however, it could be useful to have a DC-DC converter able to provide output voltages that could be lower than that in the input as well. For this reason, the boost converter described in Section 2.2 has been improved adding an asynchronous buck converter as schematized in Fig. 4.1.

The "buck and boost" topology used has been preferred over the "buck-boost" one since the latter one provides in output negative voltage.

With this solution, just adding another MOSFET (and its driving circuitry) and one diode, it is possible to have a buck and boost converter, above all, it is possible to use the big 10A-750µH inductor both for step-up and step-down operations.

The solution in Fig. 4.1 can act like a boost converter or like a buck converter. The original buck converter is enclosed in a box in the picture. When operating as a boost converter, the MOSFET M2 is driven by the PWM and M1 is always on, conversely, when acting like a buck converter, M1 is driven by the PWM and M2 is always off.

It is worth nothing that more sophisticated conversion topologies could be investigated driving together both M1 and M2.

4.1.2 The buck converter model

Since a mathematical model for the DC-DC boost converter has already been provided in Section 2.2.2, another one for the buck converter will be developed below. The block diagram in Fig. 4.1 is considered and it is assumed the MOSFET M2 is off and the threshold voltage V_γ of D2 is negligible. The

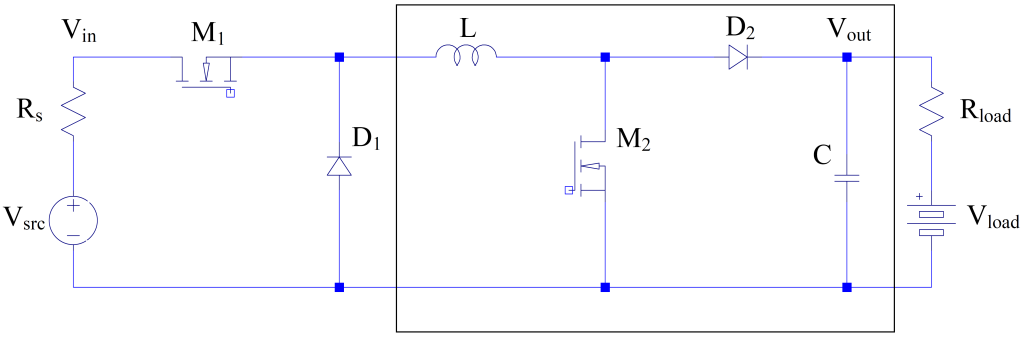


Figure 4.1: Simplified schematic of the buck and boost DC-DC converter.

load is represented with a DC generator with voltage V_{load} in series with a resistor. This is a good model when the load is a battery.

As for the boost converter, the first step is to evaluate the current $i(t)$ through the inductor L when M1 is on, that is when $t \in T_{ON} = [t_0+kT, t_0+kT+DT]$, with $k = 0, 1, 2, ...$, where T is the PWM signal period, and D is the duty cycle.

It is easy to demonstrate that:

$$i(t) = i_0 e^{-t/\tau_{on}} + \frac{V_{src} - V_{load}}{R_1}\left(1 - e^{-t/\tau_{on}}\right) \qquad t \in T_{ON} \tag{4.1}$$

When M1 is off, that is when $t \in T_{OFF} = [t_0 + kT + DT, t_0 + (k+1)T]$, with $k = 0, 1, 2, ...$, the current $i(t)$ is given by the following:

$$i(t) = i_0 e^{-t/\tau_{off}} + \frac{V_\gamma + V_{load}}{R_2}\left(1 - e^{-t/\tau_{off}}\right) \qquad t \in T_{OFF} \tag{4.2}$$

where $R_1 = R_s+R_L+R_{DS}+R_{load}$, $\tau_{on} = L/R_1$, $R_2 = R_L+R_{load}$, $\tau_{off} = L/R_2$, R_L is the inductor resistance, V_γ is the threshold voltage of the diodes, R_s is

the source resistance, and R_{DS} is static Drain-to-Source on-resistance of the MOSFET.

If I is the mean value of the current $i(t)$, because $i_{on}(t_k + DT) = I + \Delta I$, from eq. (4.1) it is possible to write:

$$\Delta I = \left(\frac{V_{src} - V_{load}}{R_1} - I \right) \frac{\left(1 - e^{-DT/\tau_{on}}\right)}{\left(1 + e^{-DT/\tau_{on}}\right)} \tag{4.3}$$

In the same way, but starting from eq. (4.2), it is possible to obtain:

$$\Delta I = \left(\frac{V_{\gamma} + V_{load}}{R_2} + I \right) \frac{\left(1 - e^{-(1-D)T/\tau_{off}}\right)}{\left(1 + e^{-(1-D)T/\tau_{off}}\right)} \tag{4.4}$$

Since ΔI given by eq. (4.3) must be the same of that given by eq. (4.4), comparing the two expressions it is possible to write:

$$\left(\frac{V_{src} - V_{load}}{R_1} - I \right) \times \alpha = \left(\frac{V_{\gamma} + V_{load}}{R_2} + I \right) \times \beta \tag{4.5}$$

where:

$$\alpha = \frac{\left(1 - e^{-DT/\tau_{on}}\right)}{\left(1 + e^{-DT/\tau_{on}}\right)}, \quad and \quad \beta = \frac{\left(1 - e^{-(1-D)T/\tau_{off}}\right)}{\left(1 + e^{-(1-D)T/\tau_{off}}\right)}. \tag{4.6}$$

The current I through the inductor L, that is also the current through the load, is therefore simply:

$$I_{out} = I = \frac{V_{src} - V_{load}}{R_1} \frac{\alpha}{\alpha + \beta} - \frac{V_{\gamma} + V_{load}}{R_2} \frac{\beta}{\alpha + \beta} \tag{4.7}$$

If I_{in} is the mean value of the current sunk in input by the converter, its value can be derived taking into account the power losses in the circuit:

$$V_{src}I_{in} - R_{load}I_{out}^2 - V_{load}I_{out} = \\ R_sI_{in}^2 + R_LI_{out}^2 + DR_{DS}I_{out}^2 + (1-D)\,V_\gamma I_{out} \quad (4.8)$$

where $R_sI_{in}^2$ is the is the power loss in the internal resistance of the power generator (actually this is not a power loss in the converter), $DR_{DS}I_{out}^2$ is the power loss in the MOSFET, $R_LI_{out}^2$ it the power loss due to inductor resistance, and, in the end, $(1-D)\,V_\gamma I_{out}$ is the power loss in the diode.
The solution of this quadratic equation provides the expression of I_{in}:

$$R_sI_{in}^2 - V_{src}I_{in} + \left(I_{out}^2\,(R_L + R_{load}) + V_{load}I_{out} + (1-D)\,V\gamma I_{out}\right) = 0 \quad (4.9)$$

Also in this case, the derived model has been proven to be very accurate by means of a comparison with the plots obtained with the circuit simulator LTspice IV.

4.1.3 Expected behavior

As already done with the model of the buck DC-DC converter, also in this case a C language SW simulator has been written in order to evaluate the behavior the converter in some realistic cases. The same PV panels (SUNTECH STP235-20/Wd and BP Solar SX310) used for the analysis of the boost converter will be used here for the DC-DC buck converter characterization, but in place of single PV panel SX310, a PV module made with two

PV panels connected in series is used to have an higher output voltage required to charge the battery especially in case of poor lighting. For the load, a simple battery model has been considered. It is just 12V ideal voltage generator connected in series with a 0.5Ω resistor.

The schematic diagram of the system used for the simulation is in Fig. 4.2. The box on the left encloses the model used for the PV panel, whereas the box on the right the DC-DC buck converter. In Fig. 4.3 there are the ripple and the efficiency estimated for both panels, in Fig. 4.4 there are the expected power losses in the DC-DC buck converter, whereas, in Fig. 4.5 there are the expected values for power and current provided to the external load. Unfortunately, due to lack of time, this upgrade to buck and boost converter has still not been realized.

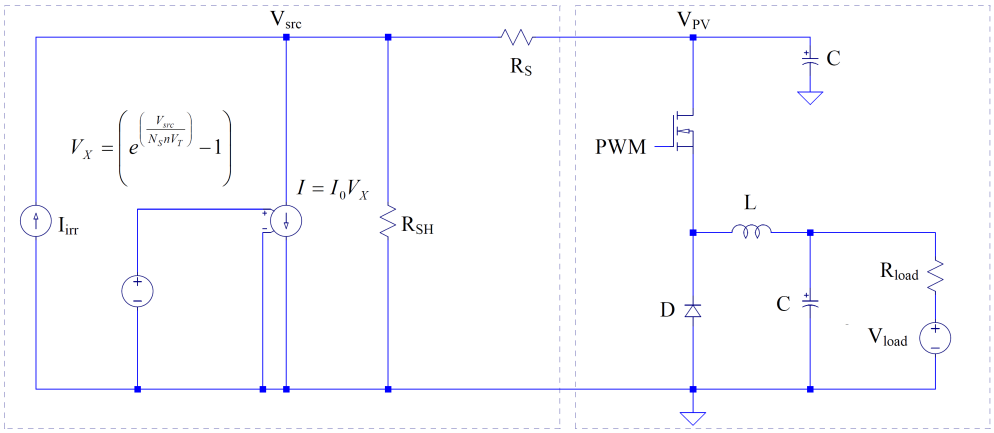


Figure 4.2: Schematic diagram of a DC-DC buck converter with in input a PV panel.

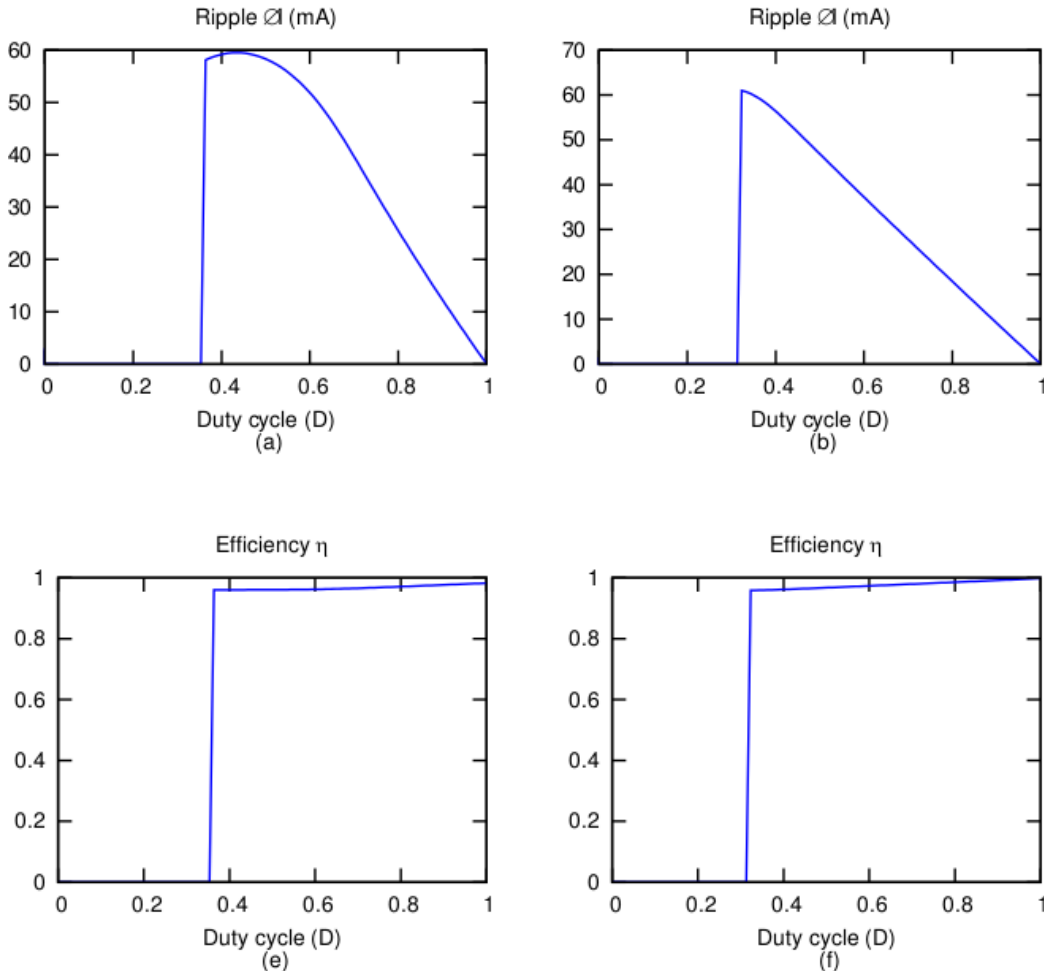


Figure 4.3: Ripple and efficiency of the buck converter with V_{load} = 12V and $R_{load} = 0.5\Omega$. The column on the right refers to the PV module made with two BP solar SX310 (10W) connected in series, whereas the plots of the left column are related to the PV module Suntech STP235-20/Wd (235W).

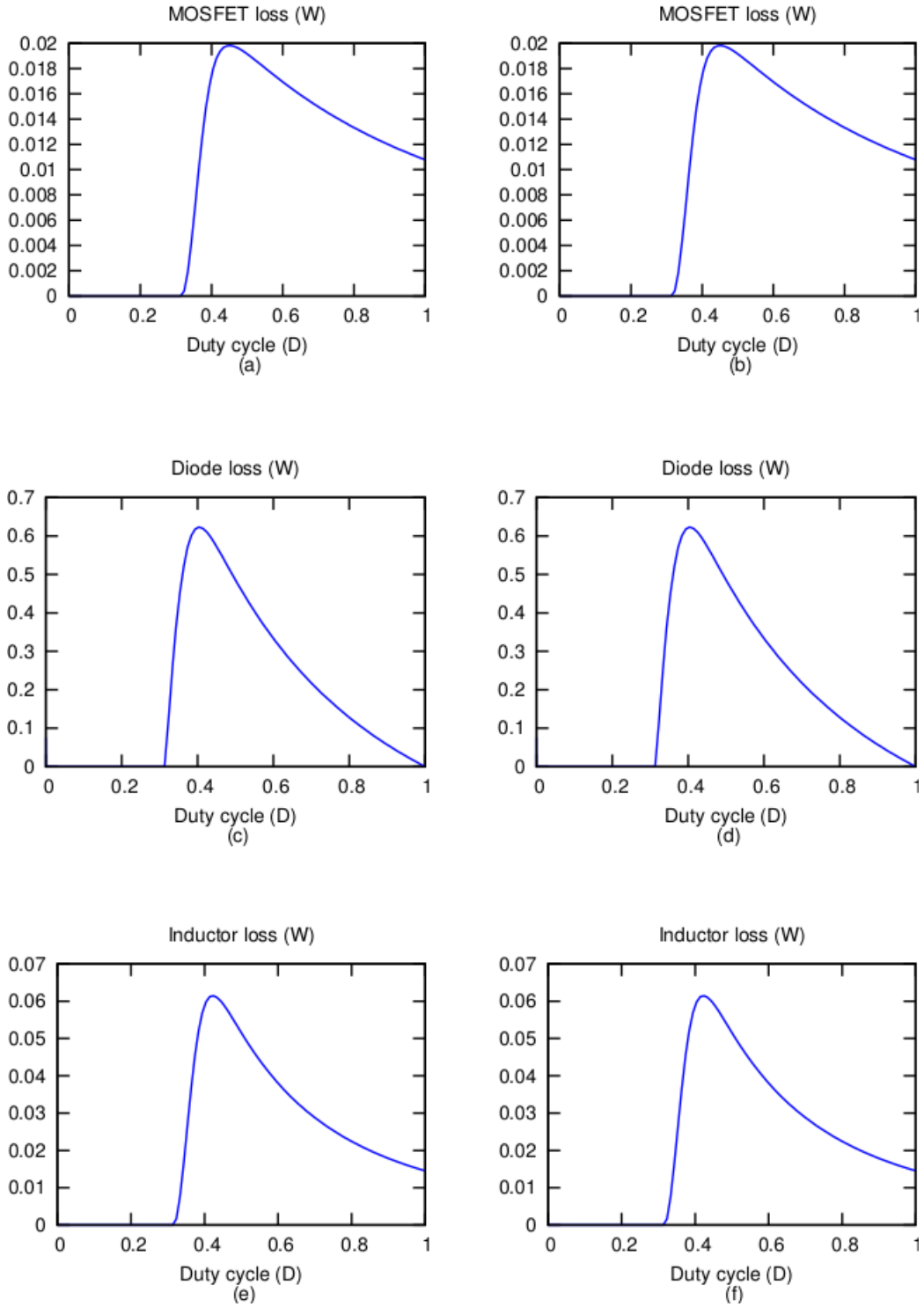
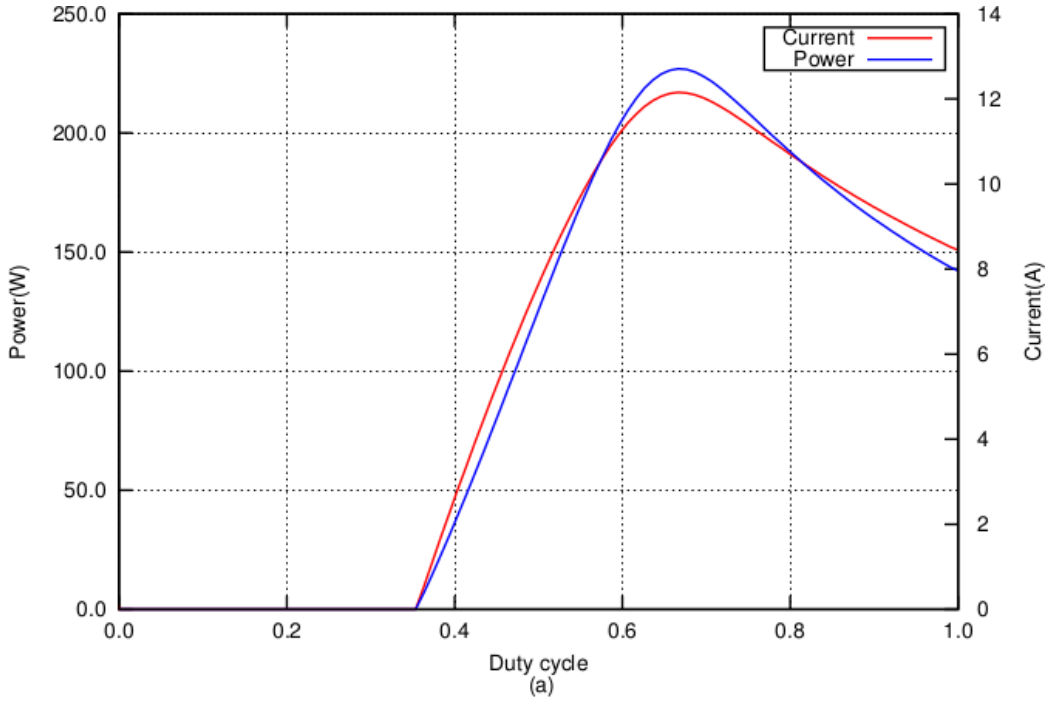


Figure 4.4: Power loss in the components of the DC-DC buck converter with $V_{load} = 12$V and $R_{load} = 0.5\Omega$. The column on the right refers to the PV module made with two BP solar SX310 (10W) connected in series, whereas the plots of the left column are related to the PV module Suntech STP235-20/Wd (235W).



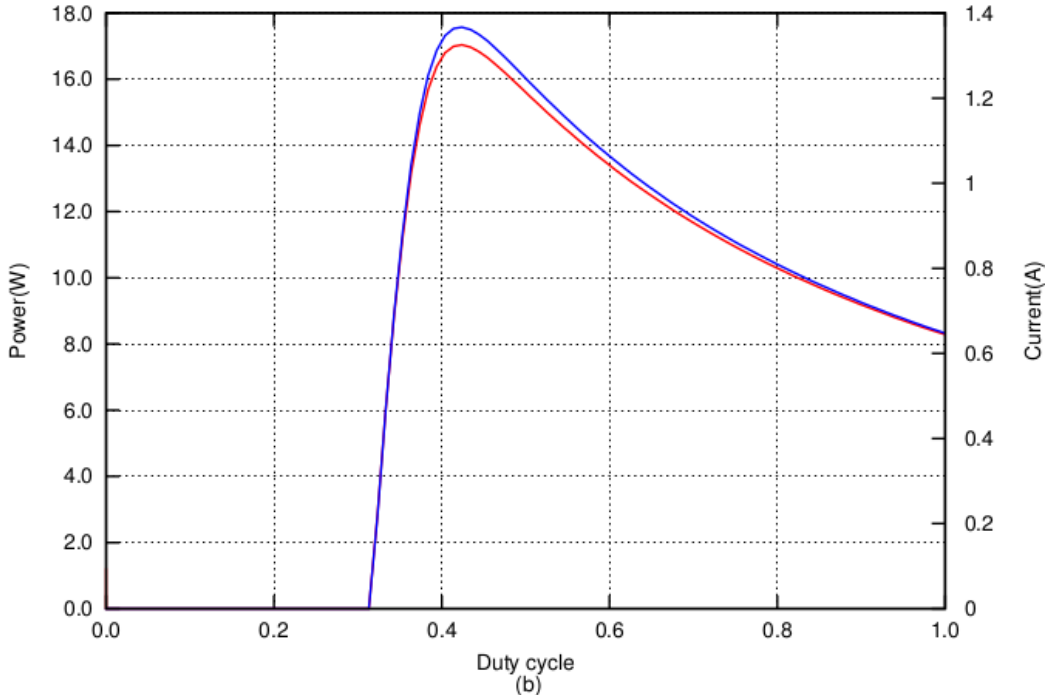


Figure 4.5: Current and power provided by the DC-DC buck converter to the load with $V_{load} = 12$V and $R_{load} = 0.5\Omega$. The plots above refer to the PV module Suntech STP235-20/Wd (235W), whereas those below are related to the PV module made with two BP solar SX310 (10W) connected in series.

4.2 MPPT algorithms verification

When dealing with a PV panels, one of the main task an engineer is expected to face, is the exploitation of the maximum power a PV panel can deliver for a given working condition. The Maximum Power Point (MPP) is a function of the temperature T and of the irradiance G, and, of course, of the of the panel itself. Since the behavior of PV panel is expected to change during the panel lifetime due to ageing, for example, tracking the MPP is a tricky task. This algorithm is expected to continuously adjust the duty cycle of the DC-DC converter quickly and precisely. Because the proposed instrument can collect values of voltage and current both in input and in output, in the next section it will be shown how it is useful to study the behavior of a Maximum Power Point Tracking (MPPT) algorithm.

4.2.1 Real use case example: a solar battery charger

In this section a solar battery charger is described. The proposed instrument can be used to have characterization of the PV panel used as already explained, but it will be shown how it can also be useful to have the charge and the discharge curves of the storage battery used. After that, the behavior of a neural network (NN) MPPT algorithm used to control the DC-DC conversion operations will be investigated. The block diagram of the system we want to study, is reported in Fig. 4.6. Only a PV panel and a battery are required in addition to the proposed system.

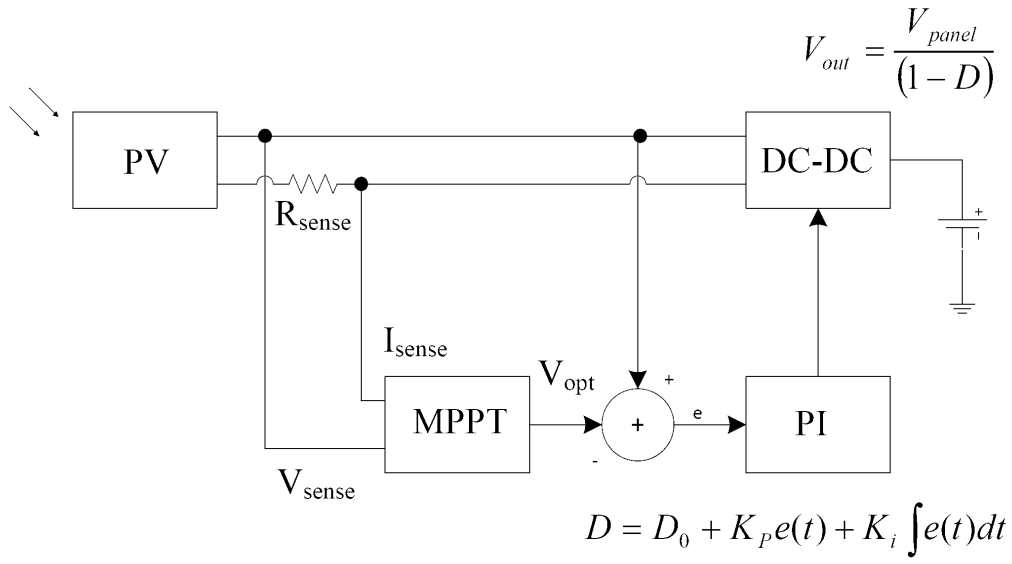


Figure 4.6: The block diagram of the solar battery charger.

PV panel characterization

The panel used for this demonstration, is made with 10 cells having the following datasheet values: $V_{oc} = 0.55$V, $I_{sc} = 2.0$A, $I_{mp} = 1.5$A, and $V_{mp} = 0.41$V. These values are enough to have characterization of the PV, but the proposed instruments was used anyway to plot the characteristic curves of the PV panel in the working condition. These curves are plotted in Fig. 4.7.

Battery characterization

For this task, a 12V-4Ah lead acid battery is considered. This battery, namely a YUASA Yucel Y4-12, is a maintenance free model often used in alarm systems, emergency lighting and other stand-by power applications where a solar

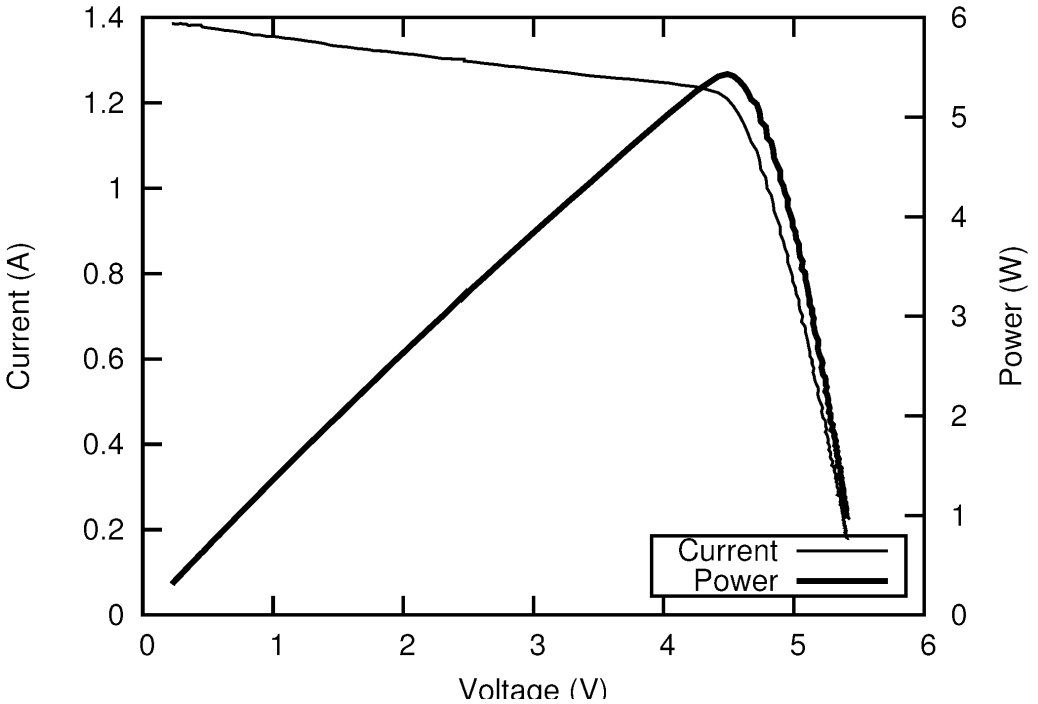


Figure 4.7: Experimental curves obtained with a little 10 cells panel and a 10W halogen lamp.

panel recharger could be considered.

To plot the battery charge curve, the proposed equipment has been configured as in Fig. 4.8. In input there is a 5V DC source and in the output the battery; the PI regulator, running in the control unit, has the task to keep constant the charge current adjusting the output voltage by means the duty cycle of the DC-DC boost converter.

To get the discharge curve, the instrument has been configured as in Fig. 4.9: the fully charged battery is connected in input and the internal resistor load is connected to the output, to convert into heat the power sourced by the battery. Also in this case, a PI regulator has the task to keep constant

the discharge current.

In Fig. 4.10 are plotted both the charge and discharge curves obtained with a current of 1A, that means a C/4 current, where C is the battery capacity. It is possible to observe that with these charge and discharge rates, the battery capacity is lower than the nominal one. It is worth nothing that very rarely the battery manufacturers provide this kind of data.

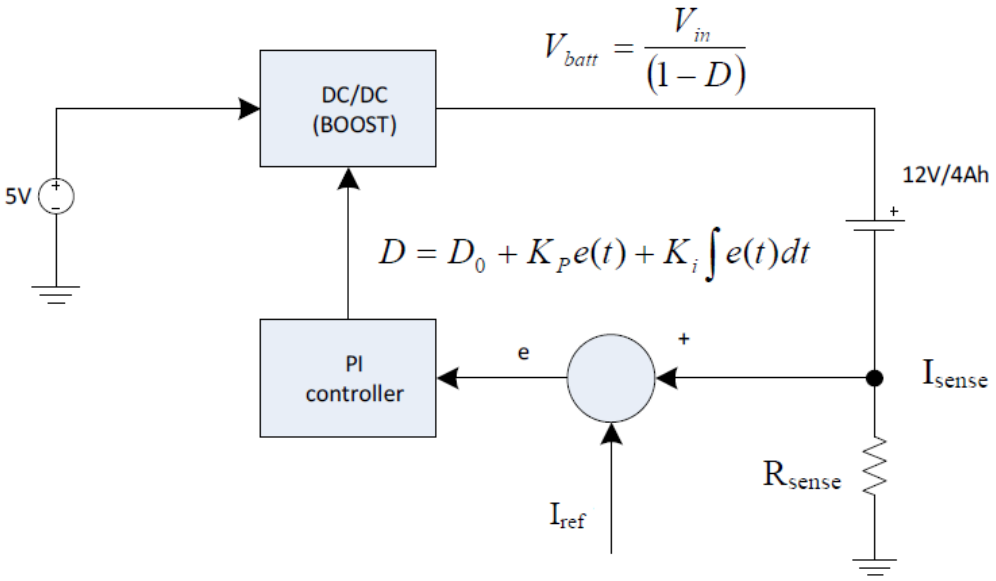


Figure 4.8: Battery charge configuration

DC-DC converter control algorithms verification

In this paper it is described the verification of a solar battery charger where the MPPT algorithm described in [Carrasco et al., 2013] is adopted. This neural network (NN) algorithm has been chosen because of its ability to

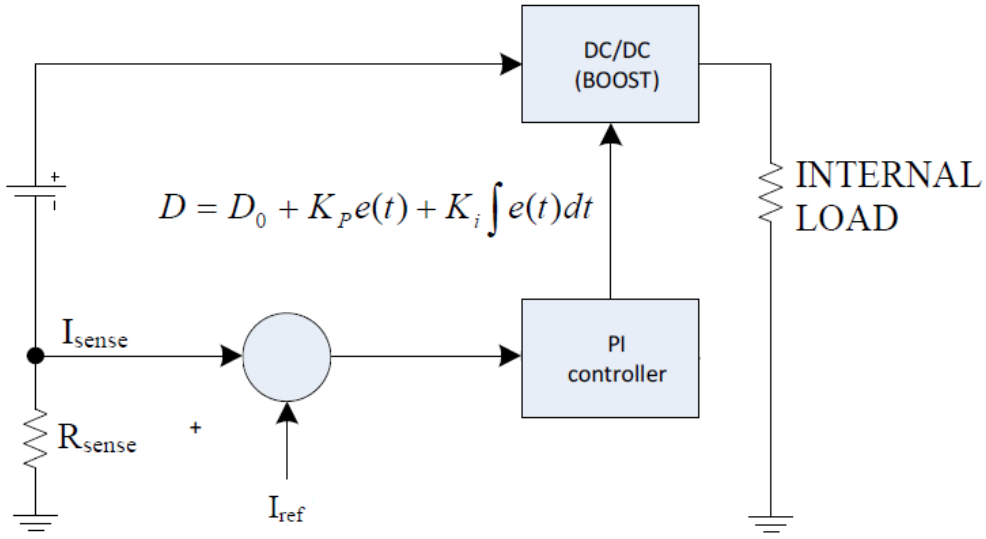


Figure 4.9: Battery discharge configuration

track rapid changes in atmospheric conditions. This algorithm provides the optimal value V_{opt} for the PV panel voltage starting from an single input pair (V, I) measured at the PV panel output. This is in contrast to other well-know MPPT algorithms such as the perturb and observe (P&O) and the incremental conductance (IC) [Masters, 2013] that iteratively approach the optimal solution. For each pair (V, I) the PV panel voltage is compared with V_{opt} and adjusted by means of the duty cycle D of the PWM controller. A proportional-integral (PI) controller accomplishes this task according to the block diagram in Fig. 4.6.

Using the NN MPPT algorithm the first thing to do is to train the NN in order to get the weights of the NN. As described in [Carrasco et al., 2013], 20 equidistant irradiance values from 50 to 1000 W/m^2 have been used. The

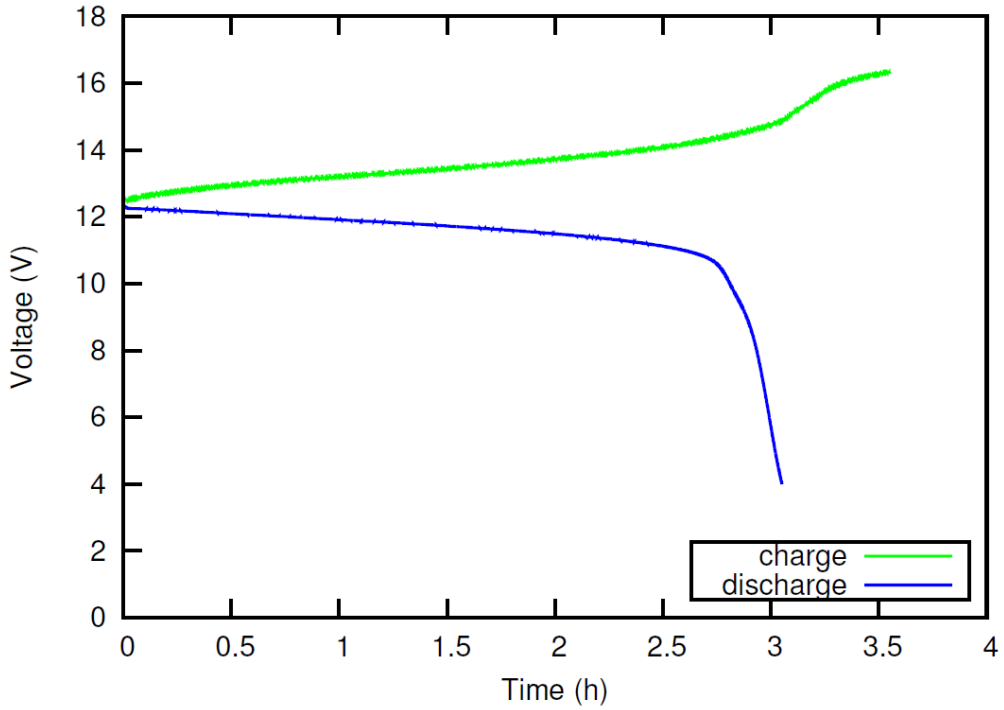


Figure 4.10: Charge and discharge curves obtained with a 1A (C/4) constant current for the battery YUASA Yucel Y4-12

panel is made with 10 cells having the following datasheet values: $V_{oc} = 0.55$V, $I_{sc} = 2.0$A, $I_{mp} = 1.5$A, and $V_{mp} = 0.41$V. The five 1-diode model parameters for this panel have been obtained according to the procedure proposed in [Laudani et al., 2014c]. With these parameters, we have prepared the input and the output vectors required to train the NN. In input there are 40 couples of values (I, V) for each G value, in output the corresponding V_{opt} values. This procedure is depicted in Fig. 4.12.

The NN used has a three-layer structure: two input neurons; eight neurons for the hidden layer; and one output neuron for the prediction of V_{opt}. It is shown in Fig. 4.11. It was created with the Matlab function `newff()`,

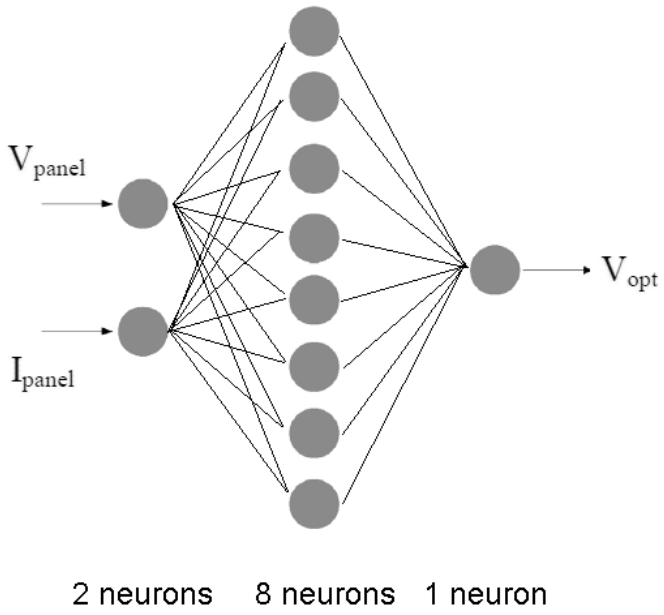


Figure 4.11: The neural network used in the MPPT controller

and trained with the function `train()` using the well known Levenberg-Marquardt algorithm. The mean square error (MSE) achieved with this NN was 1.0×10^{-5}.

After having obtained offline the NN weights, a proper SW routine to be run in the BBB was written with those. This routine, written in C language, is in charge to provide the V_{opt} where the PV panel provides the maximum power. The PI controller adjusts the duty cycle D to make the PV panel work in this point. Adjusting the duty cycle, the voltage in output from the DC-DC converter changes, and therefore the charge current too. Because the maximum power delivered by the PV panel used is about 10W, the charging current is not grater than 1A, that means a charging time of about 3.5 hours in case of a plenty of sun.

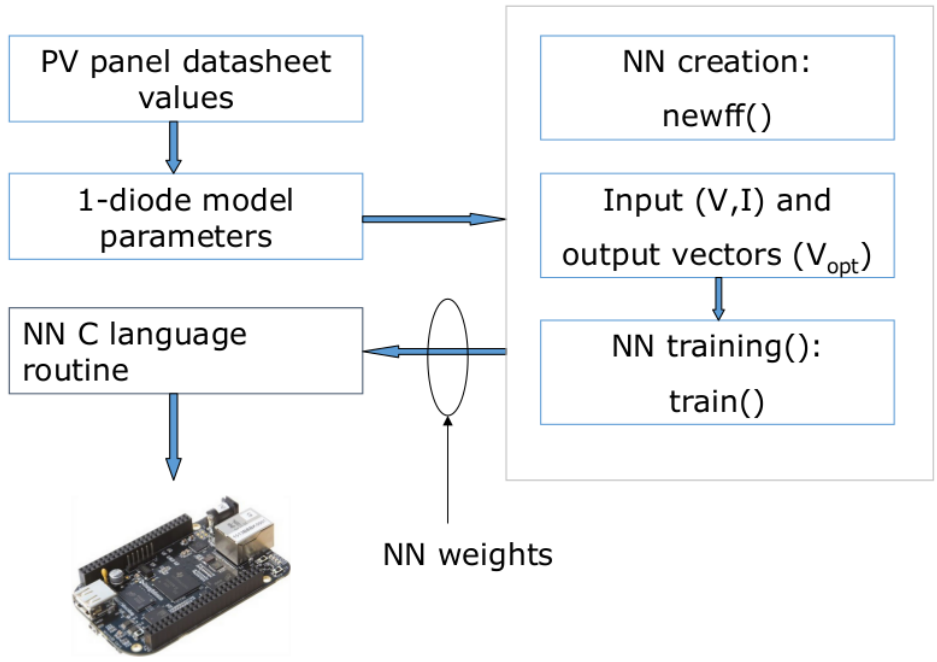


Figure 4.12: Estimation procedure of the NN weights

The charge procedure stops when the full battery condition is detected. For a lead acid battery this condition is detected by measuring the battery voltage. From the charge characteristic curve reported in Fig. 4.10, we have decided to stop the charge when the battery voltage reaches the value of 15V. Once the charge is stopped, it can be resumed when the battery voltage goes below a fixed threshold value. Other different control strategies could be evaluated with the proposed equipment as well.

To test the NN MPPT algorithm the experimental setup showed in Fig. 4.13 has been used. Besides the little PV panel, the 12V battery and the proposed equipment, it is possible to see the 400W halogen lamp used to light

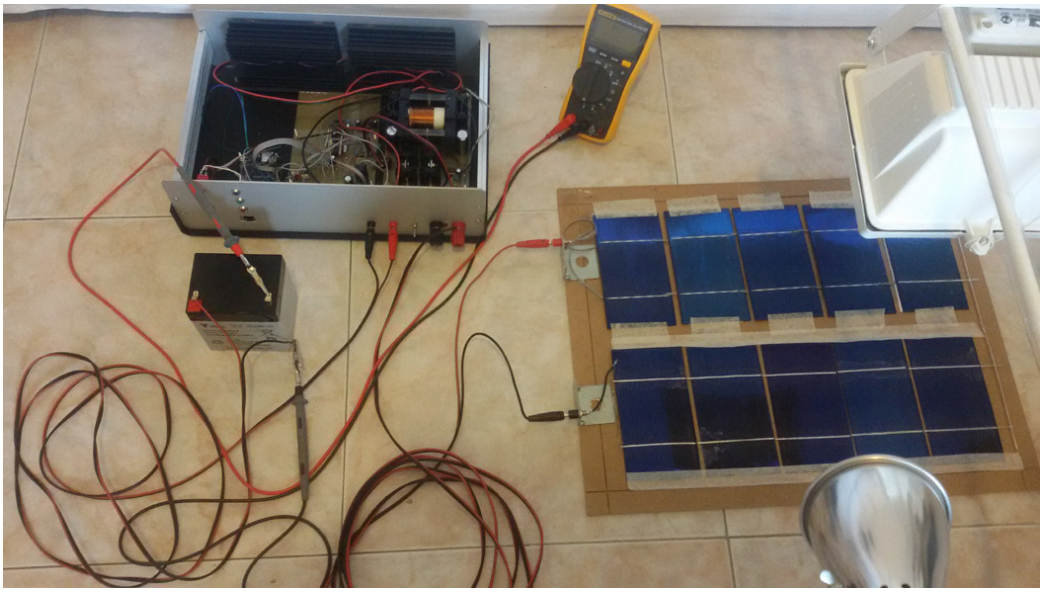


Figure 4.13: Experimental setup used for MPPT algorithm verification

.

the panel. When the lamp is on, the short circuit current of the panel is about 1.4A, that means the irradiance G is about 0.5 sun.

This value is representative of most of the real working conditions. With this setup some experiments have been performed, moving the panel and turning on and off the lamp. In Fig. 4.14 four of them have been reported. In each case it is plotted the V_{opt} provided by the NN MPPT algorithm and the effective panel voltage. The resolution of the proposed equipment is 50 measurements per second, that allowed us to examine the behavior of the system also during the fast turn on of the lamp.

The first case in Fig. 4.14 is about the turn on of the lamp. This is a very quick and big change in G, but the system demonstrated to be able to work in the new condition in less then 0.5s. When the 400W lamp was off, the charge current was about 10mA, due to the presence of another little lamp,

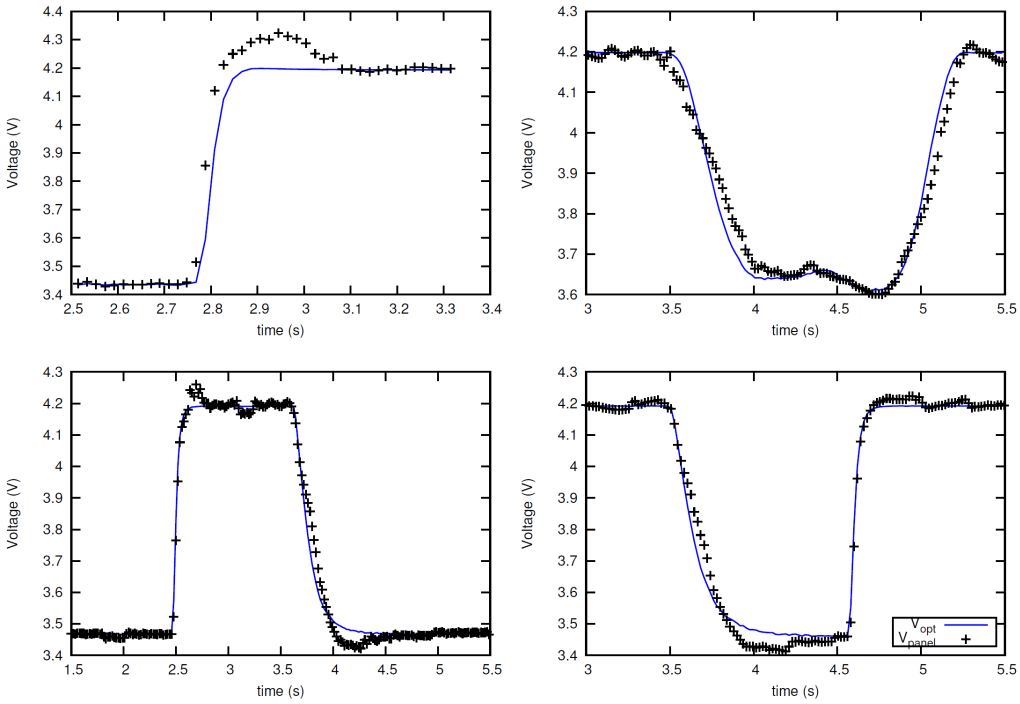


Figure 4.14: Experimental results obtained with the MPPT algorithm described in [Carrasco et al., 2013]. In (a) there is the case when the lamp is turned on, in (b) the panel has been moved under the lamp, in (c) and (d) the lamp has been turned on and off for about 1s.

but when the 400W lamp was turned on, it quickly reached the steady state value of about 350mA. In Fig. 4.14(b) the panel has been manually shifted under the lamp. Also in this case, the system followed the MPP very quickly and precisely. In the last two cases the lamp has been turned on (case (c)) and off (case (d)) for about 1s.

These simple experiments demonstrated that the NN MPPT algorithm in [Carrasco et al., 2013] is a good solution when fast changes in the irradiance G are expected.

In these experiments we have taken into account very fast changes in the ir-

radiance, but in a real solar battery charger, the temperature should also be involved since V_{opt} depends also on it. In this case, $V_{opt} = V_{opt}(I_{panel}, V_{panel}, T)$ and therefore a more complex NN is required, like showed in [Lozito et al., 2014]. The proposed equipment can still be used because it provides a lot of ways to connect itself to a remote temperature sensor.

In the end, in this section the prototype of a solar battery charger has been described to demonstrate the usefulness of the proposed device. The development of this project required only some simple additional laboratory equipments, but no hardware was required to be built, making the prototyping process of a final product faster and cheaper.

Part II

Photovoltaic panels monitoring

4.3 Introduction

Solar energy harvesting requires initial remarkable investments that must be carefully evaluated, considering the revenues that are expected to return in the following years [Kato et al., 1998]. Usually PV panels manufactures guarantee their panels for about 25 years. For this reason, the return of investment in solar panel plants must happen during the solar panels expected life. Unfortunately, it has been observed that some panels suffer from early degradation due to the outdoor operation [Munoz et al., 2011,Bastidas-Rodriguez et al., 2017,Accarino et al., 2013,De Soto et al., 2006]. This degradation reduces notably the available power [Kaplanis and Kaplani, 2011] as can be observed if a characterization of the PV panel is performed after degradation took place [Osterwald, 2003, Gaiotto et al., 2015]. It is possible to model the PV by means of the One-Diode model, that is a very common circuit model found in literature. Through this modelling, the process of degradation can be represented by a progressive increment of the series resistance and a progressive reduction of the shunt resistance. This means that a model identified from datasheet values [Laudani et al., 2014c, Laudani et al., 2013b] or by experimental curves [Laudani et al., 2014b] will lose accuracy over time. An accurately identified model is critical for several applications, among which Maximum Power Point Tracking [Vincheh et al., 2014, Mahamad and Saon, 2014, Lei et al., 2011, Kazmi et al., 2009] (especially for Neural Network based approaches [Laudani et al., 2014d, Lozito et al., 2014]) and solar irradiance measurement [Oliveri et al., 2017,Carrasco et al., 2017, Mancilla-David et al., 2014]. Constant monitoring of the process of degradation is a difficult issue (an interesting review on photovoltaic

degradation rate methodologies can be found in [Phinikarides et al., 2014]). Possible solutions revolving around maintenance and fault assessment relies on soft computing techniques such as Neural Networks (NN) [Chine et al., 2016, Li et al., 2012]. Other procedures, such as the one in this work aims to improve [Bastidas-Rodriguez et al., 2017], make use of a comparison between the panel expected behavior (i.e. the one from the model, not taking into account degradation) and the measured one (i.e. the one influenced by degradation). With respect to [Bastidas-Rodriguez et al., 2017], this work introduces two major advantages. First, the model identification is performed with an advanced technique able to reduce the number of parameters from 5 to 2 [Laudani et al., 2014c]. This yields a better accuracy and a simpler and deterministic procedure for model identification. Second, since the procedure for degradation estimation in [Bastidas-Rodriguez et al., 2017] requires the knowledge of the environmental quantities of temperature (T) and irradiance (G) for correct model computation, an analytic expression able to compute G from actual temperature and workpoint (v, i) is proposed. The proposed approach is tested twofold. First, the increased accuracy obtained by identification thorough [Laudani et al., 2014c] is validated by identifying the PV device Kyocera KC200GT from datasheet values (see Table). Second, the irradiance analytic expression is used to monitor a simulated degradation of the panel BP 3235T.

The technique presented hereafter has been accepted for publication in the journal "Solar Energy" and will published soon.

4.4 A model for a PV device and its degradation over time

A very common circuit model used in literature to represent the behavior of a PV device is the one-diode model, which is shown in Fig. 2.2. The model was originally conceived to represent a single PV cell, however, by a proportional modification of the parameters, it can represent a panel composed by N_S cells in series, and N_P strings in parallel. The characteristic of this circuit model is given in eq.(4.10).

$$i_{\mathrm{pv}} = I_{\mathrm{irr}} - I_0 \left[\exp\left(\frac{q(v_{\mathrm{pv}} + i_{\mathrm{pv}} R_{\mathrm{S}})}{N_{\mathrm{S}} n k T} \right) - 1 \right] - \frac{v_{\mathrm{pv}} + i_{\mathrm{pv}} R_{\mathrm{S}}}{R_{\mathrm{SH}}} \tag{4.10}$$

where $q = 1.602 \times 10^{-19}$ C is the electron's electric charge, $k = 1.3806503 \times 10^{-23}$ J/K is the Boltzmann constant, and n is the ideality factor, I_{irr} is the photo current or irradiance current generated when the cell is exposed to sunlight; I_0 is the diode saturation current or cell reverse saturation current. Eq. (4.10) is an implicit non-linear equation that must be solved numerically. The model is characterized by five parameters that define the circuit elements of Fig. 2.2. The five parameters are the independent generator current I_{irr}, the inverse saturation current for the diode I_0, the ideality factor for the diode n, the series resistance R_S and the shunt resistance R_{SH}. Those five parameters depend on the environmental conditions G, T, and their reference values measured at standard reference condition (SRC, $T_{\mathrm{ref}} = 25°\mathrm{C}$, $G_{\mathrm{ref}} = 1000\mathrm{W})/\mathrm{m}^2$), namely G_{ref}, T_{ref}, $I_{\mathrm{irr,ref}}$, $I_{0,\mathrm{ref}}$, $R_{\mathrm{SH,ref}}$, $R_{\mathrm{S,ref}}$, and n_{ref},

as described by eq. (4.11) through eq. (4.15), ([De Soto et al., 2006])

$$I_{\mathrm{irr}} = \frac{G}{G_{\mathrm{ref}}}(I_{\mathrm{irr,ref}} + \alpha_{\mathrm{T}}(T - T_{\mathrm{ref}})) \tag{4.11}$$

$$I_0 = I_{0,\mathrm{ref}} \left(\frac{T}{T_{\mathrm{ref}}}\right)^3 \exp\left(\frac{E_{\mathrm{g,ref}}}{kT_{\mathrm{ref}}} - \frac{E_{\mathrm{g}}}{kT}\right) \tag{4.12}$$

$$R_{\mathrm{SH}} = R_{\mathrm{SH,ref}} \left(\frac{G_{\mathrm{ref}}}{G}\right) \tag{4.13}$$

$$R_{\mathrm{S}} = R_{\mathrm{S,ref}} \tag{4.14}$$

$$n = n_{\mathrm{ref}} \tag{4.15}$$

In eq. (4.11) α_T is the temperature coefficient of the short–circuit current which represents the rate of change of the short–circuit current with respect to T. In eq. (4.12), E_{g} is the bandgap energy for silicon in eV expressed by:

$$E_{\mathrm{g}} = 1.17 - 4.73 \times 10^{-4} \times \frac{T^2}{T + 636}. \tag{4.16}$$

Identification of this model is an open problem in literature. Producers of PV devices usually report in their datasheet the following data:

- V_{OC}: The open circuit voltage
- I_{SC}: The short circuit current
- V_{mpp}: The maximum power voltage

- I_{mpp}: The maximum power current
- $\alpha_{V_{OC}}$: Temperature Coefficient of V_{OC}
- $\alpha_{I_{SC}}$: Temperature Coefficient of I_{SC}

Starting form these parameters, in [Bastidas-Rodriguez et al., 2017] and [Accarino et al., 2013] a procedure is proposed to identify the one-diode model by means of the following equations:

$$I_{irr} \approx I_{SC} \tag{4.17}$$

$$n = \frac{\alpha_{V_{OC}} - V_{OC}/T}{N_s V_t \left(\frac{\alpha_{I_{SC}}}{I_{irr}} - \frac{3}{T} - \frac{E_{gap}}{kT^2} \right)} \tag{4.18}$$

$$I_0 \approx I_{irr} e^{-\frac{V_{OC}}{N_s n V_t}} \tag{4.19}$$

$$\theta = \frac{V_{mpp} \left(2I_{mpp} - I_{irr}\right) \exp\left(\frac{V_{mpp}(V_{mpp} - 2N_s n V_t)}{N_S n^2 V_t^2} \right)}{N_s n I_0 V_t} \tag{4.20}$$

$$x = W\left(\theta\right) + 2\frac{V_{mpp}}{N_S n V_t} - \frac{V_{mpp}^2}{N_S n^2 V_t^2} \tag{4.21}$$

$$R_S = \frac{x N_S n V_t - V_{mpp}}{I_{mpp}} \tag{4.22}$$

$$R_{SH} = \frac{x N_S n V_t}{I_{irr} - I_{mpp} - I_0 \left(e^x - 1\right)} \tag{4.23}$$

Where $W(\theta)$ is the Lambert Function of argument θ. During PV panel lifetime, due to outdoor operations, panel parameters may change. The effect of the change is a reduction in V_{mpp} and I_{mpp} that can be taken into account with a reduction in R_{SH} and an increase in R_S. In [Bastidas-Rodriguez et al., 2017] it is proposed to evaluate this degradation online (i.e. in operative condition), by using the following expressions :

$$\Delta R_S = \frac{V\left(I_{mpp}\right) - V_{mpp}}{I_{mpp}} \tag{4.24}$$

and

$$\Delta R_{SH} = \frac{V_{mpp}}{I\left(V_{mpp}\right) - I_{mpp}} \tag{4.25}$$

The two indicators proposed express the degradation as a discrepancy between the actual panel performance and the model behavior (uninfluenced by the degradation). The former is expressed by the values of V_{mpp} and I_{mpp}. Those values are measured experimentally with a rather good precision if a good MPPT algorithm is implemented on for the PV device control. The latter are the values of $V(I_{mpp})$ and $I\left(V_{mpp}\right)$. Those values are the MPP coordinates on the I-V curve given by eq. (4.10). Concerning the solution of eq. (4.10), it is possible to do it numerically by using a simple SW routine employing, for example, the Newton's method, or by means the Lambert W function. In [Bastidas-Rodriguez et al., 2017] it is proven that eq. (4.24) and eq. (4.25) provide a good estimation of (4.22) and eq. (4.23) and that they have low sensitivity with respect to irradiance and temperature. When deriving $V(I_{mpp})$ and $I\left(V_{mpp}\right)$ from eq. (4.10) however, the effect of both

temperature and irradiance on the panel parameters must be taken into account. It is possible to accomplish this task thorough eq. (4.17)-eq. (4.23). In [Bastidas-Rodriguez et al., 2017] it is underlined that irradiance measurement of a PV module is expensive, therefore, the authors proposed to substitute in eq. (4.17) and eq. (4.23) the term G/G_{STC} with the approximation $I_{SC}/I_{SC,STC}$. Using this approximation is unprecise, and more importantly, requires to change the operating point of the panel in order to measure I_{SC}.

4.5 The proposed improvements

The indicators eq. (4.24) and eq. (4.25) described in the previous section suffer from two limitations: the first one is relative to the procedure proposed in [Bastidas-Rodriguez et al., 2017] and [Accarino et al., 2013] to obtain the five parameters of the one-diode model from the datasheet of the panel, that is quite approximate if compared to more accurate alternative approaches described in [Laudani et al., 2014c, Laudani et al., 2013b, Laudani et al., 2014f]. The second one is the requirement to change the operating point of the panel in order to measure I_{SC}, used to estimate G/G_{STC}. In the next subsections we propose a solution to overcome these limitations.

4.5.1 The first improvement

With the aim to increase the accuracy on the model identification, we use the procedure described in [Laudani et al., 2014c] instead of the one used in [Accarino et al., 2013] to obtain the five parameters of the one-diode model starting from the datasheet of the panel. The procedure in [Laudani et al.,

2014c] consists in splitting the five original parameters defining the one-diode model into two sets. The first set consists of independent parameters, and is composed by R_S and n. The second set consists of dependent parameters, and is composed by the remaning I_{irr}, I_0, R_{SH}. By manipulating the relationship obtained by evaluating eq.(4.10) in open circuit (OC), short circuit (SC) and maximum power (MP), three explicit relationships expressing the dependent parameters in function of the independent parameters are formulated. These relationships reduce the model complexity from a 5 parameters model to a 2 parameters model. Moreover, it is possible to manipulate the non-negativity of the independent parameters to obtain an upper boundary for the dependent parameters. The dependent parameters R_S and n can be found through a cost function obtained by imposing the null-slope of the P-V curve at maximum power point. Overall, the procedure proposed in [Laudani et al., 2014c] is extremely efficient, guaratees a physical solution (i.e. no negative parameters) and can be implemented with numerical techniques as simple as the bisection method.

4.5.2 The second improvement

The second improvement we are going to introduce is related to the necessity of changing the operating point for the PV array in order to measure I_{SC}, as used in the approximation $(I_{SC}/I_{SC,STC}) \approx (G/G_{STC})$. This necessity is due to the cost of an irradiance sensor. A possible approach, proposed in [Laudani et al., 2014d] propose linking the irradiance quantity to the workpoint of the PV device and its temperature. This approach requires a suitably trained Neural Network to be implemented in embedded environment. This is in

general a difficult task that, according to the architecture, might require computational optimizations [Laudani et al., 2015, Laudani et al., 2014e]. In [Carrasco et al., 2017], however, an interesting technique for sensing irradiance in PV panel is proposed. In this work it is shown that the irradiance G can be directly derived form any operating point of a PV panel, if its one-diode model is known, according the the following expression:

$$G = G_{ref} \frac{i_{pv} + I_{0,ref} \left(\frac{T_{pv}}{T_{ref}}\right)^3 \exp(A) \left[\exp(B) - 1\right]}{I_{irr,ref} \left[1 + \left(\frac{\alpha_{I_{SC}}}{I_{SC,ref}}\right)(T_{pv} - T_{ref})\right] + C} \tag{4.26}$$

where:

$$A = \frac{E_{g,ref}}{V_T} - \frac{E_g T_{ref}}{V_T T_{pv}} \tag{4.27}$$

$$B = \frac{T_{ref}\left(v_{pv} + R_{s,ref} i_{pv}\right)}{N_s n_{ref} V_T T_{pv}} \tag{4.28}$$

$$C = \left(\frac{v_{pv} + R_{s,ref} i_{pv}}{R_{sh,ref}}\right) \tag{4.29}$$

4.5.3 The monitoring procedure

Due to the ageing of the panel, the estimations given eq. (4.26) becomes increasingly inaccurate over time. It is easy to prove that the variations on R_{sh} affect very little the accuracy in G estimation, whereas variations on R_s have a great impact. In Fig. 4.15 three curves are plotted: the first one

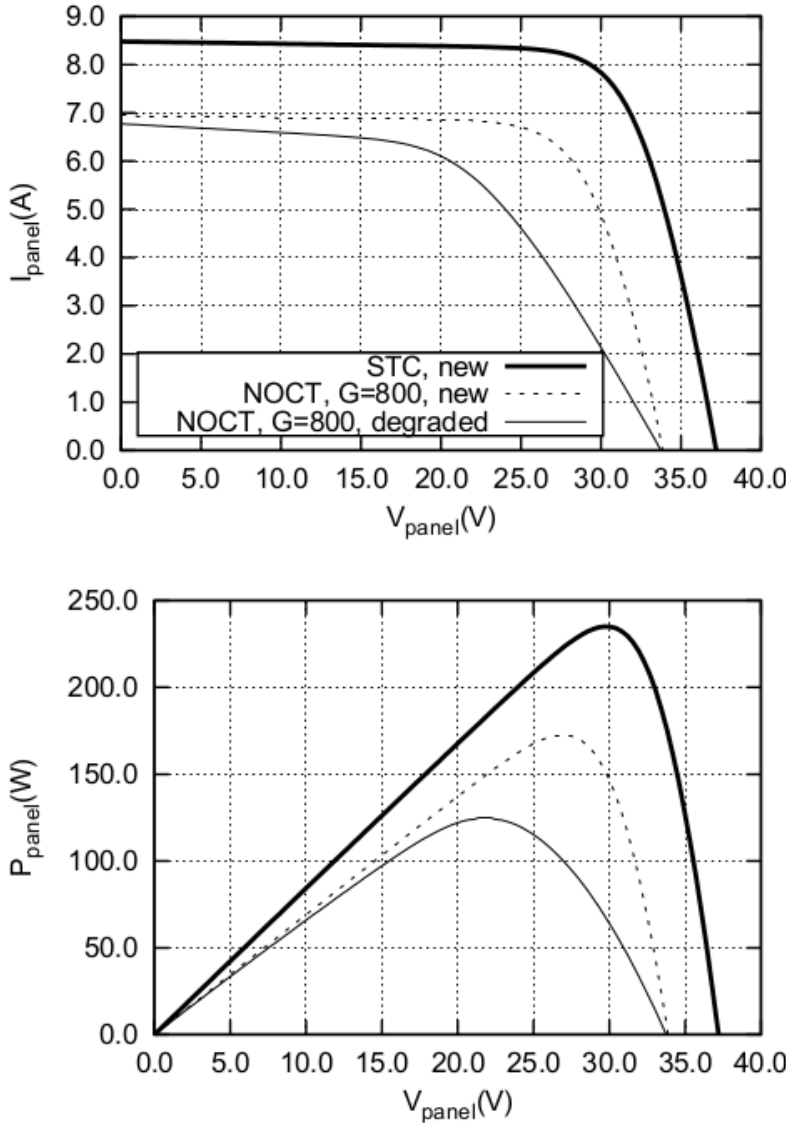


Figure 4.15: I-V curve degradation for BP 3235T panel. The degraded panel related plot has been obtained with $R'_s = 4R_s$ and $R'_{sh} = R_{sh}/4$.

refers to the a new panel at STC conditions, the second to a new panel at Nominal Operation Cell Temperature (NOCT) of 47°C and $G = 800\text{W}/m^2$. In these conditions its R_s and R_{sh} changes a little according to eq. (4.14) and eq. (4.13) (actually only R_{sh} depends on G). Let us call R_s and R_{sh} the series and the shunt resistors of the panel on these conditions.

The third plot still refers to a PV panel at NOCT and $G = 800\text{W}/m^2$, but it is degraded having $R'_s = 4R_s$ and $R'_{sh} = R_{sh}/4$. We will use hereafter the apex in a parameter to indicate a degraded panel.

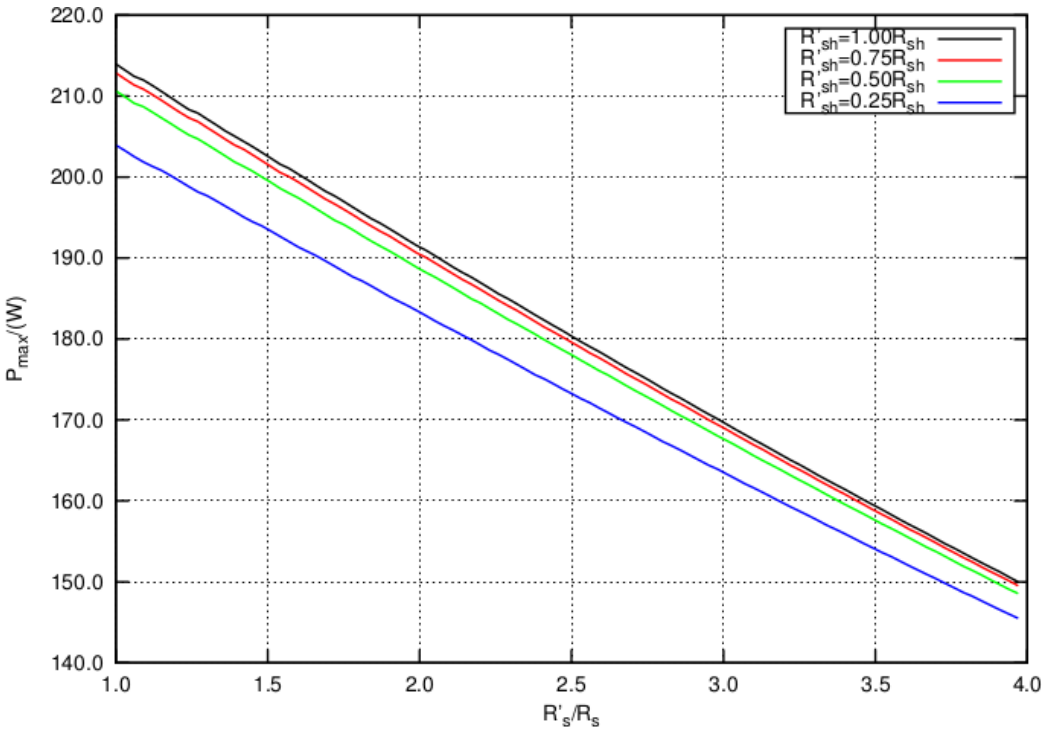


Figure 4.16: Expected P_{max} reduction for the PV panel BP 3235T as a function of R'_s and R'_{sh}

In Figs. 4.16 and 4.17 there is the expected maximum power reduction

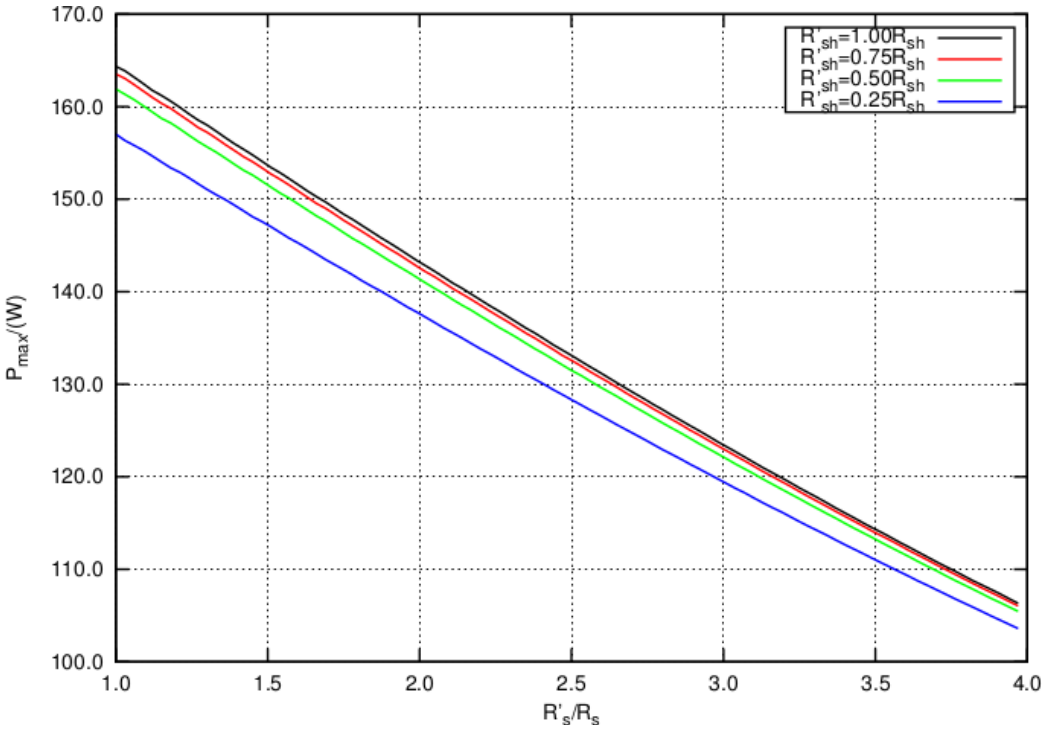


Figure 4.17: Expected P_{max} reduction for the PV panel KC200GT as a function of R'_s and R'_{sh}

respectively for the PV panels BP 3235T ($V_{OC} = 37.2$V, $I_{SC} = 8.48$A, $V_{mpp} = 29.8$V, $I_{mpp} = 7.89$A, $\alpha_{I_{SC}} = 8.904e-3$A/°C, $\alpha_{V_{OC}} = -0.00360$V/°C, $N_S = 60$) and KC200GT (see Table 4.1) as a function of ΔR_S and ΔR_{SH}. It is interesting to note that variations of R_S are more significant than variations on R_{SH}.

To take into account the ageing of the PV panel in eq. (4.26), the estimations provided by eq. (4.24) can be used to adjust the PV model used in eq. (4.26). The flowchart for the proposed monitoring procedure can be seen in Fig. 4.18. The PV source is equipped with a suitable controller able to give a digital measurement of panel voltage (v), current (i) and temperature (T), and track

accurately the Maximum Power Point thorough a suitable algorithm.
The online procedure follows this continuous loop:

- The (v, i, T) triplet is used to compute the irradiance G along with the parameters $\gamma'_{ref} = \left\{I'_{irr,ref}, I'_{0,ref}, R'_{S,ref}, R'_{SH,ref}, n'_{ref}\right\}$ from the identified Five Parameters Model.

- The computed irradiance is used by the model to compute the $I(V_{mpp})$ and $V(I_{mpp})$ terms.

- The $I(V_{mpp})$ and $V(I_{mpp})$ terms are compared against the panel V_{mpp} and I_{mpp} in the panel age estimator block, where the corrections ΔR_S and ΔR_{SH} are computed by means of eq. (4.24) and eq. (4.25). Actually, only ΔR_S is required since the influence of ΔR_{SH} in eq.(4.26) is negligible. Furthermore, R_S does not depends of T and G, so ΔR_S provided by eq. (4.24) can be used to adjust the model of the panel at SRC required by eq.(4.26).

- The model parametersγ'_{ref} are updated to reflect the occurred degradation.

4.6 Validation

In this section the validation of the improvements proposed in this work will be presented. For the first improvement, an identification test was performed on a typical power-producing panel, the Kyocera KC200GT. The producer datasheet values are shown in Table 4.1.

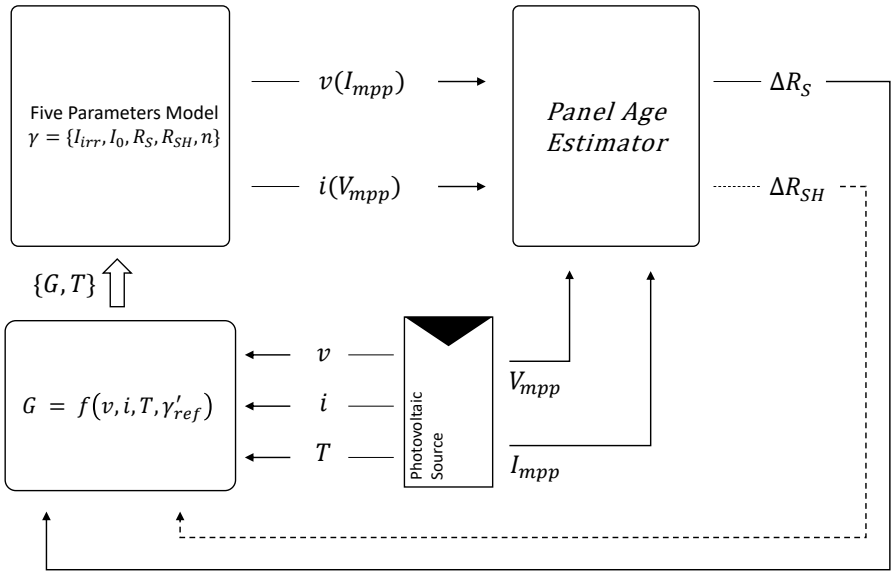


Figure 4.18: Schematic representation of the procedure for on-line monitoring of panel degradation

Table 4.1: Datasheet values for the PV panel Kyocera KC200GT

Parameter	Values @ STC	Values @ NOCT and $G = 800W/m^2$
Maximum power P_{max}	200W	142W
Short-circuit current I_{SC}	8.21A	6.22A
Open-circuit voltage V_{OC}	32.9V	29.9V
Maximum Power current I_{mpp}	7.61A	6.13A
Maximum Power voltage V_{mpp}	26.3V	23.2V
Temperature Coefficient of I_{SC} ($\alpha_{I_{SC}}$)	0.0032A/°C	-
Temperature Coefficient of V_{OC} ($\alpha_{V_{OC}}$)	-0.1230V/°C	-
Number of cells	54	-

Table 4.2: Parameters obtained with the methods proposed in [Accarino et al., 2013] and [Laudani et al., 2014c]

Parameter	Method [Accarino et al., 2013]	Method [Laudani et al., 2014c]
$I_{\text{irr,ref}}$ [A]	8.21	8.2267
$I_{0,\text{ref}}$ [A]	$2.1546 \cdot 10^{-9}$	$5.1857 \cdot 10^{-9}$
$R_{\text{P,ref}}$ [Ω]	157.536	163.4774
$R_{\text{S,ref}}$ [Ω]	0.2844	0.3324
n_{ref}	1.0755	1.0105801

Table 4.3: Parameters @ $T = 47°\text{C}$ and $G = 800\text{W}/m^2$ estimation obtained with [Accarino et al., 2013] and [Laudani et al., 2014c]

Parameter	Datasheet	Method [Accarino et al., 2013]	Method [Laudani et al., 2014c]
V_{mpp}	23.2V	22.98V	23.48V
I_{mpp}	6.13A	6.16A	6.14A
I_{sc}	6.62A	6.61A	6.62A
V_{oc}	29.9V	29.48V	29.85V

By using the two methodologies proposed in [Accarino et al., 2013] and [Laudani et al., 2014c] the panel was identified starting from datasheet values. The parameters obtained are reported in Table 4.2.
After that, these parameters have been adjusted according to eq. (4.11) through eq. (4.15) to take into account the working condition $T = 47°\text{C}$ and $G = 800\text{W}/m^2$, after that, the parameters I_{SC}, V_{OC}, V_{mpp}, and I_{mpp} reported in the datsheet of the panel (also reported in Table 4.1), have been predicted. Table 4.3 and Fig. 4.19 show the improvements obtained by using the reduced forms [Laudani et al., 2014c] based approach against the one described in [Accarino et al., 2013].

To verify the goodness of the second improvement proposed in this work

(and the relative procedure), a simulation was performed introducing a progressive degradation in the PV panel BP 3235T according to the following rules:

$$R'_s = R_s + 3R_s\left(\frac{i}{100}\right), \quad i = 0, ..., 100 \tag{4.30}$$

$$R'_{sh} = R_{sh} - 0.75R_{sh}\left(\frac{i}{100}\right) \tag{4.31}$$

where R_s and R_{sh} are the PV panel parameters at a given working conditions without degradation, and R'_s and R'_{sh} with degradation. At each step i, a progressive degradation is introduced in the PV according to eq. (4.30) and eq. (4.31). After that eq. (4.26) is evaluated and the resulting value G is used, along with the temperature T, to evaluate ΔR_S and ΔR_{SH} by means of eq. (4.24) and eq. (4.25) respectively. The value of eq. (4.24) is used to adjust the PV model used in eq. (4.26) for the next iteration. During all the simulation the irradiance was kept constant to the value of 800W/m^2. In Fig.4.20(b) is plotted the value of the irradiance given by eq. (4.26). Note that this value was very accurate across all the simulated panel life thanks to the accuracy of R_S degradation provided by eq. (4.24). In Fig.4.20(a) the actual and the estimated variations on R_S are plotted. After 100 iterations, the value of R'_s and R'_{sh} are respectively four times and one fourth of the original values.

4.6.1 Computational load

In the previous section it was shown how it is possible to have a very good estimation of G performing 100 iterations of the proposed procedure to track the degradation of a panel during all its lifetime. Assuming a PV panel lifetime equal to 25 years, it means each iteration should be performed every three months.

During each it iteration its required to evaluate:

- Computation of G by means eq. (4.26)
- Computation of $I(V_{mpp})$ and $V(I_{mpp})$
- Computation of ΔR_S by means of eq. (4.24)

The first item requires few single precision floating point operations. The exponential function required in this step could be easily obtained by means a simple SW routine implementing the Taylor series approximating this function.

The second step requires the solution of the eq. (4.10). This task could be accomplished providing a numerical solution of this equation for example by using the well known Newton's method that usually provides a good approximation in a few cycles.

The last step, that is the computation of the eq. (4.24), requires only two floating point operations.

In the end, only few lines a C source code are required to implement the proposed procedure. Since it is required to run it about once every three months, almost any controller without a floating point operation unit (FPU) could

be used to implement the proposed procedure, making use of SW emulation of the FPU.

4.6.2 Applications

The five parameters model is crucial for several applications that revolves around knowing the exact current-voltage characteristic of a PV device. The model is used to estimate power production in large scale plants [Picault et al., 2010] and assess mismatch losses [Lorente et al., 2014, Peled and Appelbaum, 2016]. For control purposes, the model has a central role for irradiance measurement and MPPT algorithms. The first case is, in general, an application of eq. 4.26. However, other techniques that assess irradiance through non-analytic approaches (such as [Mancilla-David et al., 2014, Oliveri et al., 2017]) benefits from an updated model as well since the training dataset needs to be updated periodically to reflect the device degradation. The second possible application is the development of a model based MPPT algorithm. Although the classic approach found in literature for MPPT relies on a direct approach (e.g. the Perturbe & Observe method) this strongly limits the performances of the tracking in presence of rapidly variable environmental conditions (although several research was done on the topic, see [Tey and Mekhilef, 2014, Li, 2014, Radjai et al., 2014]). Indeed, if an accurate model and environmental conditions measurements G, T were constantly available, it could be possible to compute the MPP by simply calculating the current-voltage characteristic. However, more refined techniques involving manipulations of the model equations have been proposed in literature such as [Batzelis et al., 2015, Farivar et al., 2013]. For these techniques the avail-

ability of an up-to-date model is obviously crucial. For numerical techniques such as [Laudani et al., 2014d,Lozito et al., 2014] the advantage relies, again, in having an up-to-date dataset to perform constant on-line training.

4.7 Conclusion

In these sections a novel procedure for on-line monitoring of a photovoltaic module degradation over time have been presented. The procedure is based on a continuous comparison of the device behavior against its identified circuit model. When degradation of the panel occurs, the behavior of the device starts to diverge from the behavior of the model. This divergence is used to update the model itself using two expressions described in [Bastidas-Rodriguez et al., 2017]. The expressions provide a differential increase of the parameter R_S and a differential decrease of the parameter R_{SH}. With respect to the procedure presented in [Bastidas-Rodriguez et al., 2017] and [Accarino et al., 2013], the procedure proposed offers two major novelties. The first is the use of an advanced identification technique able to identify the model with high accuracy. The second is the use of an analytic expression for the solar irradiance, used to evaluate the model output with great accuracy without need to change the operating point of the panel. The resulting procedure is completely online and requires little computational power to be implemented. This makes the procedure very interesting for direct integration in commercial MPPT controllers: the information of irradiance and temperature, along with a circuit model that is kept up to date in terms of degradation, allows tracing the I-V and P-V curve of the PV device in real time. On such curve

it is easy to identify the theoretical MPP, allowing an efficient initialization of the MPPT algorithm.

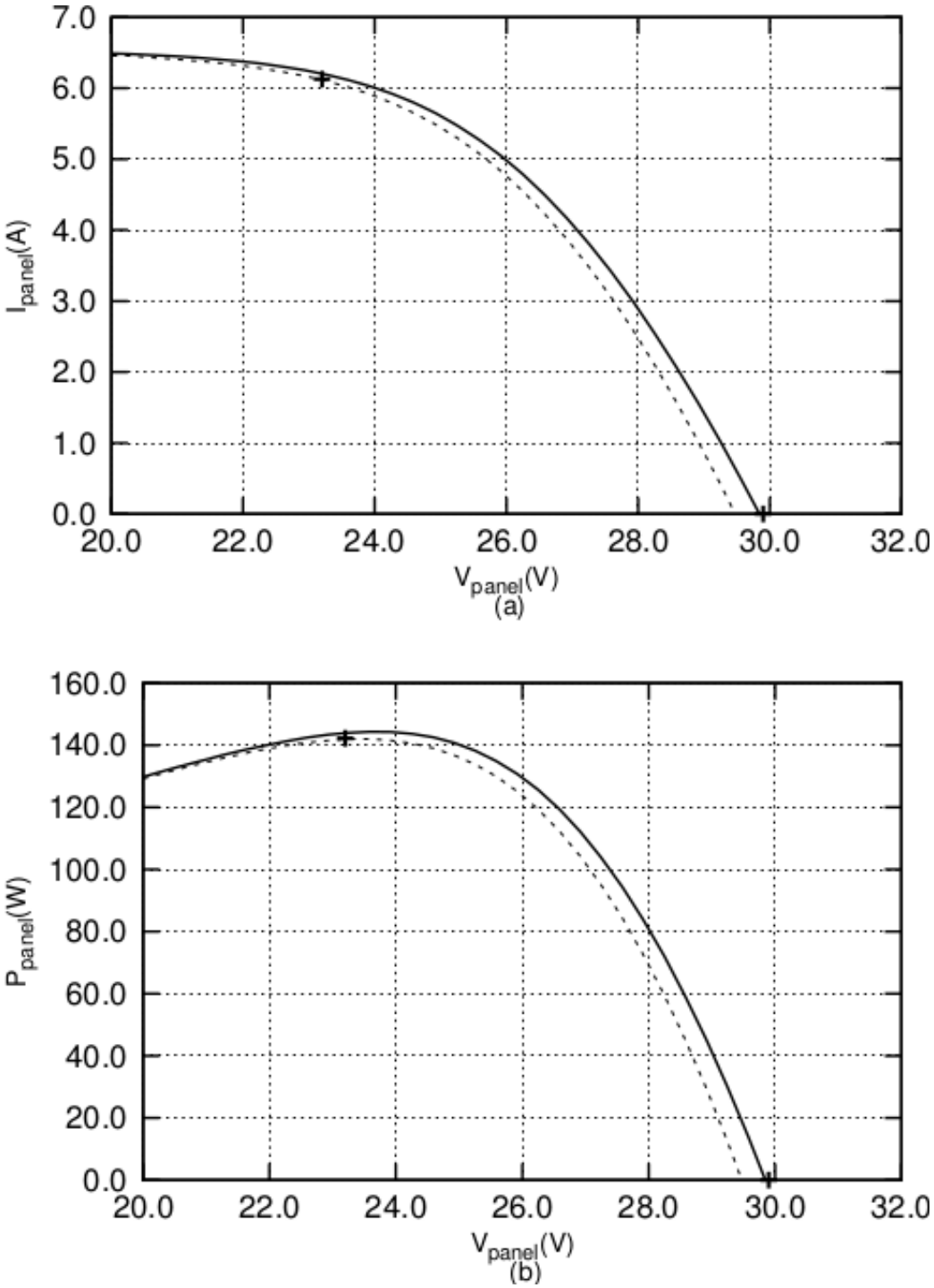


Figure 4.19: Comparison between the two methods on the $I-V$ curve (a) and $P-V$ curve (b) at $G = 800W/m^2$ and T=47°C. The '+' points refer to the values provided within the datasheet. Dashed line was obtained with method [Bastidas-Rodriguez et al., 2017], full line with method [Laudani et al., 2014c].

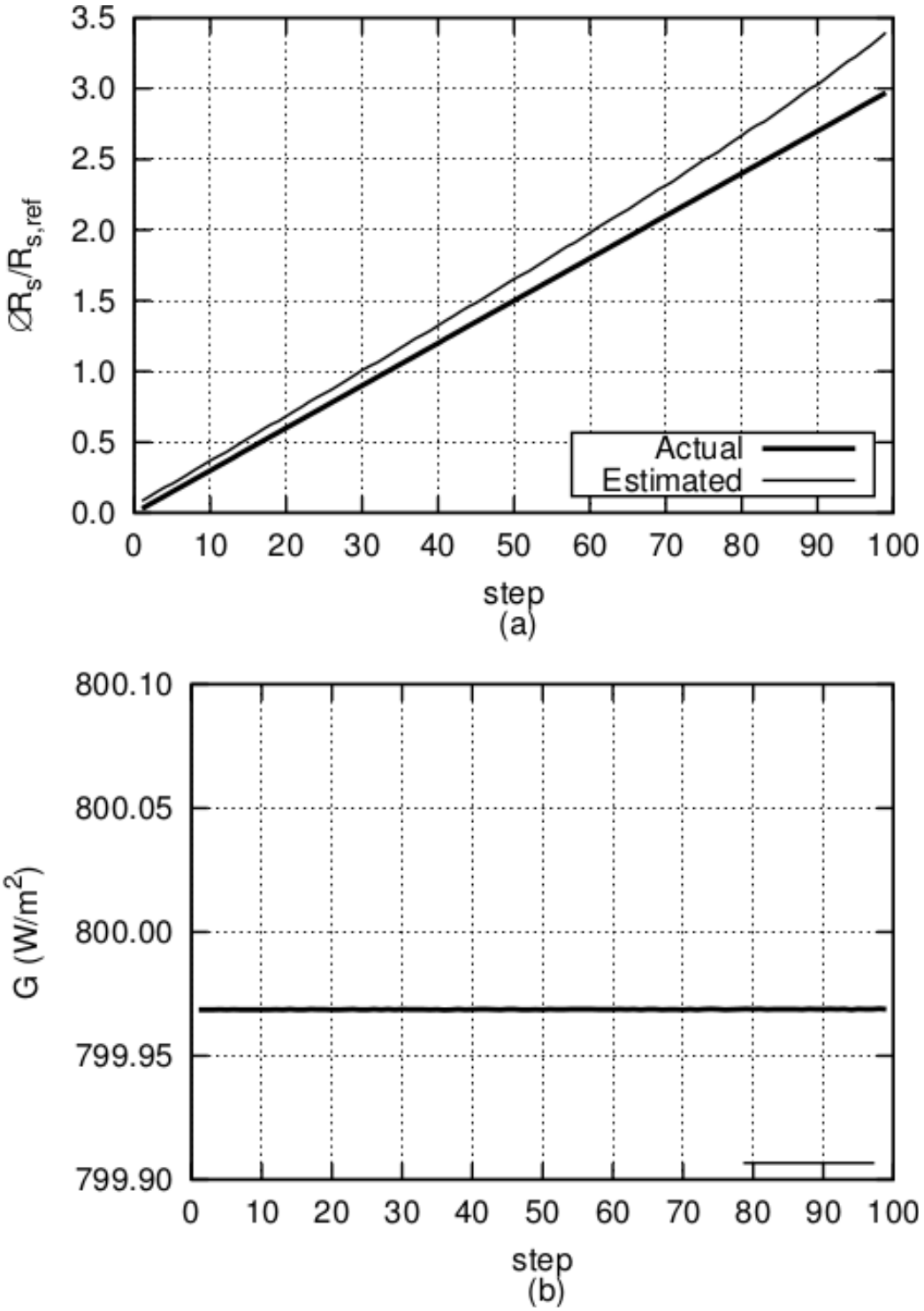


Figure 4.20: Actual and estimated variation on R_S (a) and estimated irradiance (b)

Conclusions and Future Developments

The work proposed on this PhD thesis is composed by contributions in two different areas concerning the exploitation of solar energy by means of PV panels. The first one is mainly related to PV panels characterization. The work developed about this item led to the realization of a prototype of a development system, useful for PV panels characterizations, that demonstrated to be very accurate with PV panels up to 300W. It was shown how complex algorithms like those proposed in [Laudani et al., 2014a] can be easily be embedded into it. The high accuracy and the processing capability in conjunction with the wide range of connectivity options provided by the developed system, allow a lot of future developments making use of it. First of all, the study of new models required for new PV panels type, for example the organic ones. Other characterizations different than the usual $I - V$ curves are also possible: for example it is possible the see how the MPP, or some model parameters, change according to temperature, irradiance or even ageing.

No additional hardware is required to be developed and be built for these

tasks. It is required only to write some SW routines. About this item, it would be useful to write some dynamic link libraries (dll) to make possible to control the system by using the Matlab environment. The realization of some graphic interfaces for the work cases described in this thesis work would be useful as well to complete the proposed system.

In the second part of this thesis work, it is described a new technique to keep up to date the model of a PV panel when this change because of the ageing. In many applications it is required to have an up to date model, for this reason a lot of future work should be possible exploiting this new technique. In [Faba et al., 2017] it is shown how an accurate PV panel monitoring is possible exploiting this possibility. In the future, to verify and possibly to improve this work, it would be interesting to study some real degraded panels.

Another applications is the irradiance measurement by means of the PV panel itself. In [Carrasco et al., 2017] it is shown how this task is possible. The procedure proposed in this work, however, is jeopardized by the PV panel ageing. With the contribution described in this thesis work, this limitation can be removed allowing other applications that come from the knowledge of the irradiance. The first one is the development of a model based MPPT algorithm. With an up to date model and the knowledge of temperature and irradiance it is possible know the working condition of a PV panel that allow to sunk from it the maximum power it can deliver.

The behavior of an MPPT algorithm is another use case of the proposed system, described in this thesis work.

Appendices

Appendix A

A low-ripple switched-capacitor voltage regulator with decoupling capabilities

A.1 Introduction

When an electrical circuit operates in the presence of a rapidly changing magnetic field, induced currents in the circuit are involved and must be taken into account since they may significantly affect the expected behavior of the electrical circuit. For this reason big attention must be paid when designing the layout of a circuit [Johnson et al., 1993], and galvanic isolation if often required [SLLA284A, 2014] to break the loops where induced currents can be generated.

In addition to induced currents, conducted EMI should also be taken into account: disturbances that are generated in some noisy subsystems, can reach

sensitive subsystems through the connections between circuits. Examples of these connections are the power rails. Usually, passive filters are used to stop this kind of interferences [Ott and Ott, 1988]. The operation of preventing transmission of noise from one circuit to another is usually referred to as 'decoupling'. Poor decoupling can greatly reduce circuit performance and is, in general, a topic often overlooked. For this reason, chip vendors usually provides useful guides, for example [MT-101, 2009] and [SPRA906, 2003], to help designers to understand the importance of this operation. For the same reason [Kundert, 2004] has been written.

Sometime it is the power supply unit itself that introduces noise in the dc voltage it provides. It is the case of, for example, switchmode power supplies (SMPS). Also in this case analog filters are used to reduce the noise annoying the power provided [Billings and Morey, 2011], however, the realization of a good filter is a tricky task that should also take into account parasitic effects of components and printed circuit board (PCB) tracks, furthermore, due to the parasitic in the filters, filters cannot attenuate high-frequency noises efficiently. In [Wang et al., 2005b] some techniques to cancel them and improve EMI filter performance are presented. Passive filters are not only difficult to design, but often require bulky components. To reduce both parasitic effects and space requirements, thanks to advances in electromagnetic integration technologies, integrated EMI filters are possible. In [Zhenyang et al., 2015] this kind of filters is reviewed. In [Wang et al., 2005a] the effects of filter parasitics are investigated and filter design techniques are proposed to mitigate the effects of these both in the conventional discrete filters and in those integrated.

In [Omata and Shimizu, 2016] the EMI noise radiated from the dc cable of photovoltaic power conditioner systems (PCSs) is studied. Furthermore the authors describe how the performances of passive EMI filters, connected to both the dc and the ac sides of the PCSs, are reduced when they are connected to the PCS because of resonant phenomena. In order to reduce these problems, they propose a new design method for these EMI filters.

To overcome some of the limitations of the passive EMI filters, active EMI filtering (AEF) is getting more and more interesting. Active filtering provides noise reduction introducing a synthetic signal able to suppress the noise. This signal could be generated by means of analog or digital circuits. In [Chen et al., 2006] AEF is proposed to replace large passive EMI filters by means of small passive components and active op-amp circuitry for the input filters of integrated power electronics modules when improved attenuation is required at relatively low frequencies. In [Hamza and Qiu, 2013] EMI noise attenuation is obtained by mean of a digital circuit based on a Field Programmable Gate Array (FPGA). Other examples of AEF applications can be found in [Ji et al., 2016] and [Yang et al., 2014]. Usually active filtering is used to mitigate EMI propagation towards other devices in order to satisfy EMI regulatory compliance, namely EN55022 in Europe and FCC part 15 in the USA.

In this paper, a low-ripple switched-capacitor voltage regulator is proposed in order to reduce the propagation of conducted EMI and the generation of those induced, thus introducing a new concept of active decoupling filtering. To have the required low-ripple DC-DC switched capacitor converter, a dual phase converter is proposed. Multi phase operations are

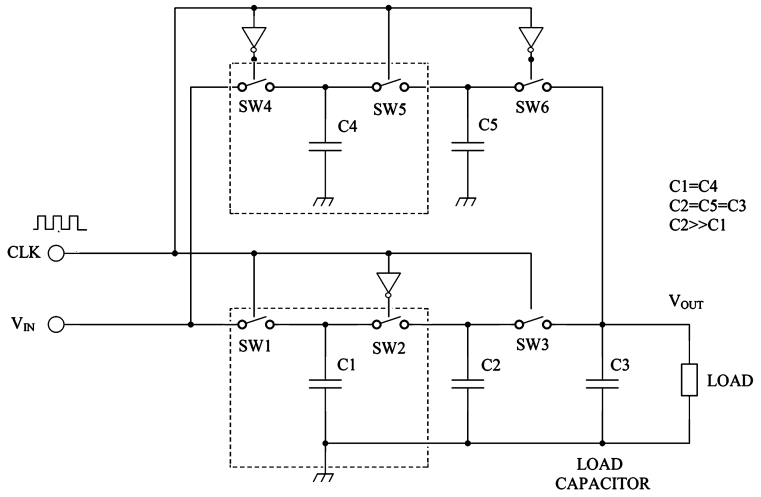


Figure A.1: The proposed switched capacitor DC-DC converter.

usually used to have a low ripple, as for example in [Luo et al., 2008], but in the proposed converter a double fly capacitor architecture is introduced. In comparison with other low-ripple solutions, like that in [Bang et al., 2016], the proposed one is much simpler.

The paper will be structured as follows. In the first paragraph, the proposed switched capacitors converter will be introduced, then, an exact mathematical model for it will be derived. Lastly, its decoupling capability will be described. Conclusions will follow.

A.2 The proposed switched capacitor DC-DC converter

In Fig. A.1 the schematic of the proposed DC-DC converter is reported. Actually, two converters are present: the upper one, composed by the elec-

tronic switches SW4, SW5, and SW6, and the capacitors C4 and C5, and the lower one, implemented by SW1, SW2, SW3, C1, and C2. They work in opposite phase, therefore, when the clock signal is high, the load capacitor C3 is loaded by the lower converter, vice-versa, when the clock signal is low, the load capacitor is loaded by the upper converter. Since the two converters are identical, only the lower one will be described.

In Fig. A.1 there are two boxes: they enclose the two building blocks where DC-DC conversion is performed. The circuit in these blocks acts like a variable resistor. It is easy to show that its equivalent resistance is given by the following:

$$R_{eq} \cong \frac{1}{fC} \tag{A.1}$$

where f is the clock frequency and C is the capacitance of the "fly" capacitor C1 (or C4). Adjusting the frequency, the output voltage can be changed. It performs like a linear DC-DC regulator. Most often a switched capacitor DC-DC regulator is used for currents up to 100mA. Assuming a voltage dropout through the converter of 2V, it results an equivalent resistance of 20Ω. Typically the switching frequency spans between 20kHz and 2MHz. Assuming a switching frequency of 500kHz, eq. (A.1) gives for the fly capacitor the value of 100nF. This value is quite low, meaning that if C1 is discharged directly in the load capacitor (that is usually bigger or equal to 10µF), it would produce an undesired spike in the output voltage. For this reason another fly capacitor C2 has been added. This capacitor has the same value of the load capacitor. Because of this, when it is discharged in the load capacitor, only a little ripple affecting the output voltage is produced.

Since the proposed converter is made with two parallel converters, and because the equivalent series resistance due to the second fly capacitors can be neglected, an approximate expression of the equivalent series resistance of the whole converter can be the following one:

$$R_{eq} \cong \frac{1}{2fC} \tag{A.2}$$

The converter in Fig. A.1 is an ideal circuit, but in a more realistic model the resistance of the switches, R_{SW} from now on, and the equivalent series resistances (ESR) of the capacitors should be considered. In Fig. A.2 the voltages at the junctions of the capacitors are plotted. They have been obtained with the Linear Technologies circuit simulator LTspice IV and the following values: $V_{IN} = 3.6$V, $C1 = 100$nF, $C2 = 10\mu$F and $ESR = 2.8\Omega$, $C3 = 10\mu$F and $ESR = 0.1\Omega$ (ceramic type), and $R_{SW} = 2\Omega$. The switching frequency f_{CK} has been calculated by means of eq. (A.2) in order to have $V_{OUT} = 2.0$V. The value given by eq. (A.2) is $f = 312$kHz, but since this value does not take into account the ESRs of the capacitors and the resistance of the switches, it is just a rough estimate. For this reason it was experimentally adjusted to 500kHz in order to have a better value for V_{OUT}. With these values, the simulation provided $V_{OUT} = 1.98$V and therefore $I_{OUT} = 98.9$mA. The plots of the Fig. A.2 show the usefulness of the double fly capacitors and of the dual phase operations. The voltage swing in the first fly capacitors, C1 and C4, is very high because of the low value of them, whereas the ripple in the second fly capacitors, C2 and C5, is very little. More in details, it is shown how the ripple in C2 and C4 is very low when they are connected to the output capacitor C5. Their voltage is with a very

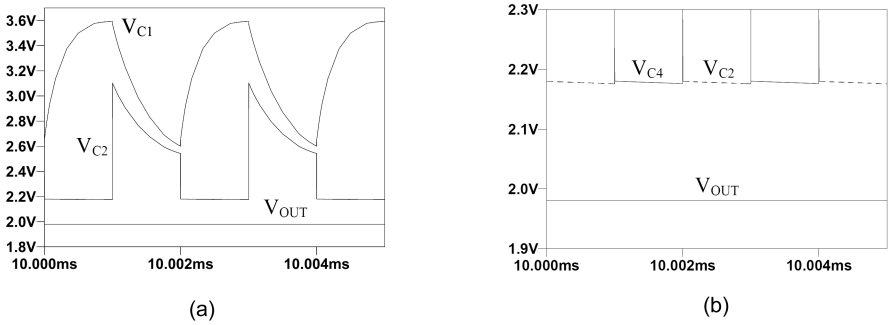


Figure A.2: Voltages with $V_{IN} = 3.6$V, $C1 = 100$nF, $C2 = 10\mu$F and $ESR = 2.8\Omega$, $C3 = 10\mu$F ceramic, $R_{SW} = 2\Omega$ and $f = 500$kHz

good accuracy the output voltage minus the voltage drop through SW3 and SW6.

A.2.1 An exact model for the proposed converter

Equation eq. (A.2) provides very rough estimates of R_{eq} and does not provide any information about the output ripple. For this reason, a better model is required. Our aim is to provide an exact model that can describe the behaviour of the converter in every conditions, for example during the start-up phase, at steady state condition or when a load change occurs. To provide this model, it is useful to identify in the schematic in Fig. A.1 the three different sub-circuits that are defined when the switches SW1, SW5 and SW3 are closed and SW2, SW4, and SW6 open and vice versa. The representations of these three sub-circuits in the Laplace domain are depicted in Fig. A.3.

The first sub-circuit is established when SW1 (or SW4) is closed and SW2 (or SW5) is open. In this condition the fly capacitor C1 (or C4) is charged. The equivalent sub-circuit in Laplace domain is in Fig. A.3(a). The equation

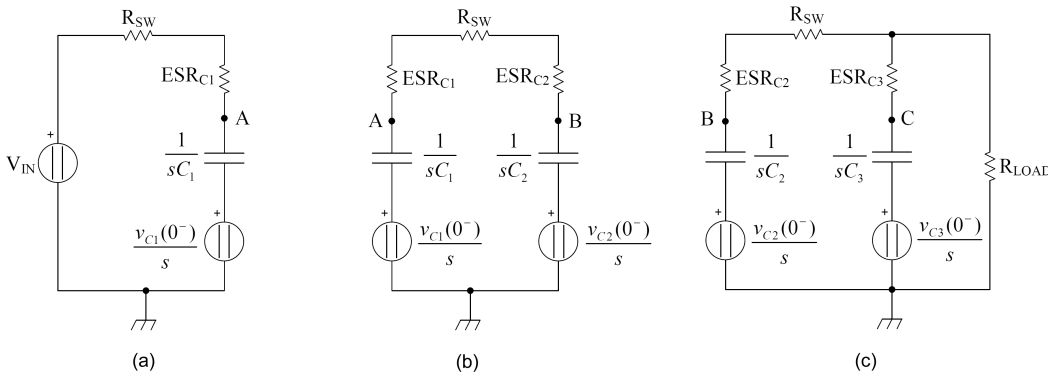


Figure A.3: Laplace domain representations of the three sub-circuits identifiable in the proposed converter.

for the loop in the s-domain is simply:

$$\frac{V_{IN}}{s} = I(s)\left[R_1 + \frac{1}{sC_1}\right] + \frac{v_{C1}(0^-)}{s} \tag{A.3}$$

that easily leads to the solution:

$$v_A(t) = V_{IN}\left[1 - \exp\left(\frac{-t}{R_1C_1}\right)\right] + v_{C1}(0^-)\exp\left(\frac{-t}{R_1C_1}\right) \tag{A.4}$$

Where R_1 takes into account both the switch resistance and the ESR of the first fly capacitor C1 and $v_{C1}(0^-)$ is the voltage of the capacitor when the switch S1 is closed.

The second sub-circuit is established when SW2 is closed and SW1 and SW3 are open. In this condition the first fly capacitor C1 is discharged into C2. The equation for the only loop of this circuit in the Laplace domain is given by:

$$\frac{v_{C1}(0^-)}{s} - I(s)\left(\frac{1}{sC_1} + R_2 + \frac{1}{sC2}\right) - \frac{v_{C2}(0^-)}{s} = 0 \tag{A.5}$$

with $R_2 = ESR_{C1} + R_{SW} + ESR_{C2}$.

In the time domain, we can easily obtain:

$$i(t) = \left(\frac{v_{C1}(0^-) - v_{C2}(0^-)}{R_2}\right) \exp\left[-\left(\frac{1}{R_2C_1} + \frac{1}{R_2C_2}\right)t\right] \tag{A.6}$$

What we need, however, is not the current through the loop, but the voltages of the capacitors, that is $v_A(t)$ and $v_B(t)$. The voltage in the junction B is given by:

$$V_B(s) = \frac{v_{C2}(0^-)}{s} + I(s)\frac{1}{sC_2} \tag{A.7}$$

That is, in the time domain:

$$v_B(t) = v_{C2}(0^-) + \frac{1}{C_2}\int i(t)dt + k \tag{A.8}$$

Introducing the expression of $i(t)$ and establishing the condition $v_B(0) = v_{C2}(0^-)$ to determine the constant k, we get to the expression:

$$v_B(t) = v_{C2}(0^-) + \big(v_{C1}(0^-) - v_{C2}(0^-)\big)\,\frac{C_2}{C_1 + C_2}\left[1 - \exp\left[-\left(\frac{1}{R_2C_1} + \frac{1}{R_2C_2}\right)t\right]\right] \tag{A.9}$$

Proceeding in the same way, we have the expression of the voltage at junction A:

$$v_A(t) = v_{C1}(0^-) - \left(v_{C1}(0^-) - v_{C2}(0^-)\right) \frac{C_2}{C_1 + C_2} \left[1 - \exp\left[-\left(\frac{1}{R_2 C_1} + \frac{1}{R_2 C_2}\right) t\right]\right] \tag{A.10}$$

The third sub-circuit, in the end, is established when SW3 is ON and SW2 is OFF. Its equivalent circuit is the Laplace domain is reported in Fig. A.3(c). The study of this sub-circuit requires to solve two simultaneous equations referring to the two loops in the circuit:

$$\begin{cases} \frac{v_{C2}(0^-)}{s} - I_1(s)\left(\frac{1}{sC_2} + \frac{1}{sC_3} + R_3 + ESR_{C3}\right) + I_2(s)\left(\frac{1}{sC_3} + ESR_{C3}\right) - \frac{v_{C3}(0^-)}{s} \\ \frac{v_{C3}(0^-)}{s} - I_2(s)\left(\frac{1}{sC_3} + ESR_{C3} + R_{LOAD}\right) + I_1(s)\left(\frac{1}{sC_3} + ESR_{C2}\right) \end{cases} \tag{A.11}$$

where:

$$R_3 = ESR_{C2} + R_{SW} \tag{A.12}$$

After some steps, it is possible to obtain the following expression:

$$I_2(s) = \frac{As + B}{Cs^2 + Ds + 1} \tag{A.13}$$

where:

$$A = v_{C3}(0^-)C_2C_3R_3 + C_2C_3v_{C2}(0^-)ESR_{C3}$$

$$B = v_{C2}(0^-)C_2 + v_{C3}(0^-)C_3$$

$$C = C_2C_3\left(ESR_{C2} + R_{SW}\right)\left(ESR_{C3} + R_{LOAD}\right) + C_2C_3R_{LOAD}ESR_{C3}$$

$$D = C_2R_3 + C_3\left(ESR_{C3} + R_{LOAD}\right) + C_2R_{LOAD}$$

To have the expression of $I_2(s)$ in the time domain it is useful to decompose it into partial fractions, that is to put eq. (A.13) in the form:

$$I_2(s) = \frac{k_1}{(s - s_1)} + \frac{k_2}{(s - s_2)} \tag{A.14}$$

Where s_1 and s_2 are the radixes of the quadratic equation $s^2 + (D/C)\,s + (1/C) = 0$ and k_1 and k_2 are given equaling eq. (A.13) and eq. (A.14). This operation leads to the following values:

$$k_2 = \frac{\left(\frac{B}{C} + s_2\frac{A}{C}\right)}{(s_2 - s_1)}; \quad k_1 = \frac{A}{C} - k_2. \tag{A.15}$$

In the time domain, the expression of $i_2(t)$ can now be simply derived:

$$i_2\left(t\right) = k_1\exp(s_1t) + k_2\exp(s_2t) \tag{A.16}$$

The output voltage is therefore $v_{OUT} = R_{LOAD}i_2(t)$.
The next step is to calculate the voltage of the fly capacitor C_2, that is $v_B(t)$, and that of the load capacitor C_3, that is $v_C(t)$:

$$v_B(s) = \frac{v_{C2}(0^-)}{s} - I_1(s)\frac{1}{sC_2} = \frac{v_{C2}(0^-)}{s} - \frac{v_{C2}(0^-)}{sC_2(1+sR_3C_2)} + \frac{I_2(s)}{(1+sR_3C_2)} \tag{A.17}$$

Making the decomposition into partial fractions and tacking the Laplace inverse transformation, we get to the following expression:

$$v_B(t) = v_{C2}(0^-)\exp(ts_3) + \frac{R_{LOAD}}{R_3C_2}\left[(h+l)\exp(ts_3) - h\exp(ts_1) - l\exp(ts_2)\right] \tag{A.18}$$

where:

$$s_3 = \frac{-1}{R_3C_2}; \quad h = \frac{-k_1}{(s_1-s_3)}; \quad l = \frac{-k_2}{(s_2-s_3)}. \tag{A.19}$$

The expression of $v_C(t)$ can be simply stated as:

$$v_C(t) = v_{LOAD}(t) + ESR_{C3}\left[i_{LOAD}(t) - \frac{v_B(t) - v_{LOAD}(t)}{R_{SW} + ESR_{C2}}\right] \tag{A.20}$$

After having derived all the equations for the sub-circuits in the converter, it is easy to write a procedure to accurately simulate the behaviour of the proposed DC-DC converter. Each clock period, the converter assumes two different configurations, according to the value of the CLK signal:

- **CLK=1:** SW1=ON, SW2=OFF, SW3=ON; SW4=OFF, SW5=ON, SW6=OFF

During this half period, the capacitor C1 is loaded with the input voltage V_{IN}

according to eq. (A.4), at the same time the fly capacitor C4 is discharged into C5 according to eq. (A.9) and eq. (A.10) and the fly capacitor C2 is discharged into the load capacitor C3 and the load according to eq. (A.16), eq. (A.18) and eq. (A.20).

- **CLK=0:** SW1=OFF, SW2=ON, SW3=OFF; SW4=ON, SW5=OFF, SW6=ON

During this half period, the situation is exactly the same of the first half, but C4 (instead of C1) is loaded with the input voltage V_{IN}, the fly capacitor C1 is discharged into C2, and the fly capacitor C5 is discharged into the load capacitor C3 and the load.
In Table A.1 the complete mathematical model for the proposed converter is reported. In this table, the sub-index U is used when referring to the upper half converter, and the sub-index L when referring to the lower one. With this notation, $v_{A_L} = v_{C1}$, $v_{A_U} = v_{C4}$, $v_{B_L} = v_{C2}$, $v_{B_U} = v_{C5}$ and $v_C = v_{C3}$.

The validity of this model has been verified comparing the output voltage predicted by a SW routine implementing the algorithm stated here with that produced by the Linear Technologies circuit simulator LTspice IV.

A.2.2 Comparison with other devices.

To evaluate the performances of the proposed DC-DC converter, it was compared with a good commercial device, namely the Linear Technologies LTC1503-2. This SC converter is able to provide a fixed output of 2V with an input voltage spanning between 2.4V to 6V. To achieve high efficiency, it uses the fractional conversion technique, that means three different con-

Table A.1: Summary of the voltage regulator model

$Initialization(k=0):$
$v_{A_L}(t_0)=v_{B_L}(t_0)=v_{C_L}(t_0)=v_{A_U}(t_0)=v_{B_U}(t_0)=v_C(t_0)=0$
$C_1=C_4; C_2=C_5$
$R_1=R_{SW}+ESR_1$
$R_2=ESR_1+R_{SW}+ESR_2$
$R_3=ESR_2+R_{SW}$
$C=C_2C_3\left(ESR_{C2}+R_{SW}\right)\left(ESR_{C3}+R_{LOAD}\right)+C_2C_3R_{LOAD}ESR_{C3}$
$D=C_2\left(ESR_{C2}+R_{SW}\right)+C_3\left(ESR_{C3}+R_{LOAD}\right)+C_2R_{LOAD}$
$s_1=\frac{-(D/C)+\sqrt{(D/C)^2-4(1/C)}}{2};\quad s_2=\frac{-(D/C)-\sqrt{(D/C)^2-4(1/C)}}{2};\quad s_3=-1/(R_3C_1)$

$t_k<t\le t_k+\frac{T}{2};\quad$ k=1,2...

$v_{A_L}(t)=V_{IN}\left[1-\exp(\frac{-t}{R_1C_1})\right]+v_{A_L}(t_k)\exp(\frac{-t}{R_1C_1})$
$v_{A_U}(t)=v_{A_U}(t_k)-\left(v_{A_U}(t_k)-v_{B_U}(t_k)\right)\frac{C_5}{C_4+C_5}\left[1-\exp\left[-\left(\frac{1}{R_2C_4}+\frac{1}{R_2C_5}\right)t\right]\right]$
$v_{B_U}(t)=v_{B_U}(t_k)+\left(v_{A_U}(t_k)-v_{B_U}(t_k)\right)\frac{C_5}{C_4+C_5}\left[1-\exp\left[-\left(\frac{1}{R_2C_4}+\frac{1}{R_2C_5}\right)t\right]\right]$
$v_{B_L}(t)=v_{B_L}(t_k)\exp\left(ts_3\right)+\frac{R_{LOAD}}{R_3C_2}\left[(h+l)\exp(ts_3)-h\exp(ts_1)-l\exp(ts_2)\right]$
$i_{OUT}\left(t\right)=k_1\exp(s_1t)+k_2\exp(s_2t)$
$v_{OUT}=i_{OUT}R_{LOAD}$
$v_C(t)=v_{OUT}(t)+ESR_{C3}\left[i_{OUT}\left(t\right)-\frac{v_{B_L}(t)-v_{OUT}(t)}{R_{SW}+ESR_{C2}}\right]$

$Where:$
$A=v_C(t_k)C_2C_3\left(ESR_{C2}+R_{SW}\right)+C_2C_3v_{B_L}(t_k)ESR_{C3}$
$B=v_{B_L}(t_k)C_2+v_C(t_k)C_3$
$k_2=\left[(B/C)+s_2\left(A/C\right)\right]/\left(s_2-s_1\right);\quad k_1=(A/C)-k_2$
$h=-k_1/(s_1-s_3);\quad l=-k_2/(s_2-s_3)$

$t_k+\frac{T}{2}<t\le t_k+T;\quad$ k=1,2...

$v_{A_U}(t)=V_{IN}\left[1-\exp(\frac{-t}{R_1C_1})\right]+v_{A_U}(t_k+\frac{T}{2})\exp(\frac{-t}{R_1C_1})$
$v_{A_L}(t)=v_{A_L}(t_k+\frac{T}{2})-\left(v_{A_L}(t_k+\frac{T}{2})-v_{B_L}(t_k+\frac{T}{2})\right)\frac{C_2}{C_1+C_2}\left[1-\exp\left[-\left(\frac{1}{R_2C_1}+\frac{1}{R_2C_2}\right)t\right]\right]$
$v_{B_L}(t)=v_{B_L}(t_k+\frac{T}{2})+\left(v_{A_L}(t_k+\frac{T}{2})-v_{B_L}(t_k+\frac{T}{2})\right)\frac{C_2}{C_1+C_2}\left[1-\exp\left[-\left(\frac{1}{R_2C_1}+\frac{1}{R_2C_2}\right)t\right]\right]$
$v_{B_U}(t)=v_{B_U}(t_k)\exp\left(ts_3\right)+\frac{R_{LOAD}}{R_3C_5}\left[(h+l)\exp(ts_3)-h\exp(ts_1)-l\exp(ts_2)\right]$
$i_{OUT}\left(t\right)=k_1\exp(s_1t)+k_2\exp(s_2t)$
$v_{OUT}=i_{OUT}R_{LOAD}$
$v_C(t)=v_{OUT}(t)+ESR_{C3}\left[i_{OUT}\left(t\right)-\frac{v_{B_U}(t)-v_{OUT}(t)}{R_{SW}+ESR_{C2}}\right]$

$Where:$
$A=v_C(t_k+\frac{T}{2})C_5C_3\left(ESR_{C5}+R_{SW}\right)+C_5C_3v_{B_U}(t_k+\frac{T}{2})ESR_{C3}$
$B=v_{B_U}(t_k+\frac{T}{2})C_5+v_C(t_k+\frac{T}{2})C_3$
$k_2=\left[(B/C)+s_2\left(A/C\right)\right]/\left(s_2-s_1\right);\quad k_1=(A/C)-k_2$
$h=-k_1/(s_1-s_3);\quad l=-k_2/(s_2-s_3)$

figurations are possible according to the voltage level in input. When the input voltage is below 3V, it works exactly like the charge-pump step down converter with the same efficiency of a linear regulator. For the comparison, we adopted the same working conditions described in the datasheet of the device, that are: $V_{IN} = 3.6$V, $V_{OUT} = 2.0$V, $I_{OUT} = 100$mA and $C_{LOAD} = 10\mu$F with very low ESR (ceramic type capacitor). It works at a fixed frequency of 600kHz. In the proposed converter, it was assumed $R_{SW} = 2\Omega$, $C1 = C4 = 100$nF, $C2 = C5 = 10\mu$F aluminum electrolytic type, with $ESR = 2.8\Omega$. ESR_{CLOAD} was set to 0.1Ω, whereas the switching frequency was set to 500kHz. It should be noted here that, in a real device, a control loop is needed to set the right working frequency to obtain the desired output voltage. With the chosen components, the model in Table A.1 predicted the output voltage plotted in In Fig. A.4. Its ripple is about 0.2mV that is significantly lower than that one reported in the datasheet of the LTC1503-2, which is about 20mV. Other commercial devices may easily show an even greater output ripple.

The low ripple SC voltage regulator proposed in [Bang et al., 2016], that is much more complex than the proposed one, exhibits an output ripple of $6 - 16$mV at 2.3V input and 1V output with a load current $11 - 142$mA.

It is worth nothing here that the proposed converter can provide low output ripple even when using cheap capacitors with high ESR. On the other hand, if higher ripple is acceptable, smaller capacitors can be used to reduce cost and space.

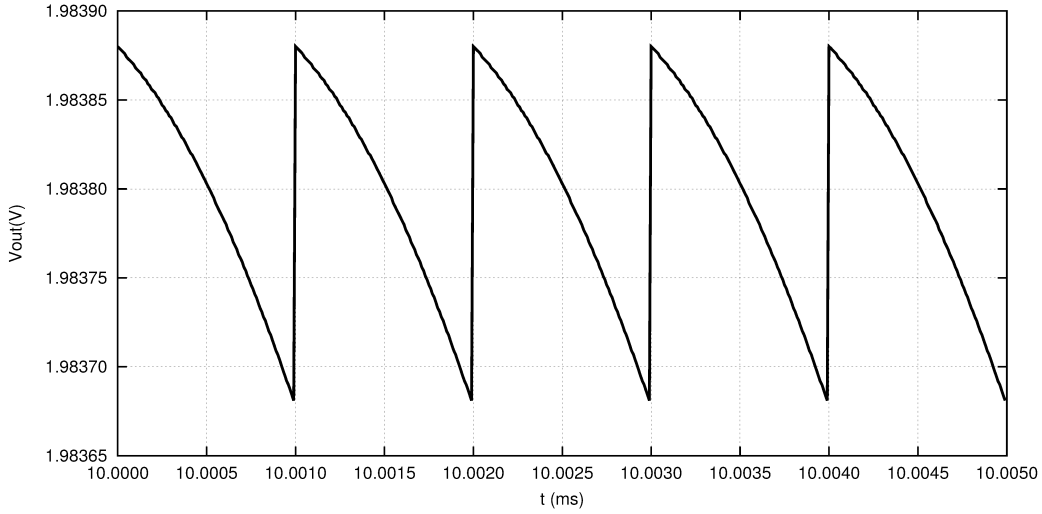


Figure A.4: The output voltage of the proposed converter with $V_{IN} = 3.6$V, $V_{OUT} = 2.0$V, and $I_{OUT} = 100$mA. The output ripple is about 0.2mV

A.3 Decoupling capabilities of the proposed converter

The proposed converter, besides featuring a very low output ripple, has the interesting capability to filter differential noise affecting the input voltage, preventing it to reach the load. This characteristic could be exploited without involving DC-DC regulation. In this case, the first fly capacitors used to adjust the output voltage could be removed and, therefore, the regulator of Fig. A.1 become like that in Fig. A.5 and can be referred to as an "active filter" rather than a voltage regulator.

The mathematical model derived for the DC-DC converter can be easily adapted for the circuit in Fig. A.5. In this case, we have the following working conditions:

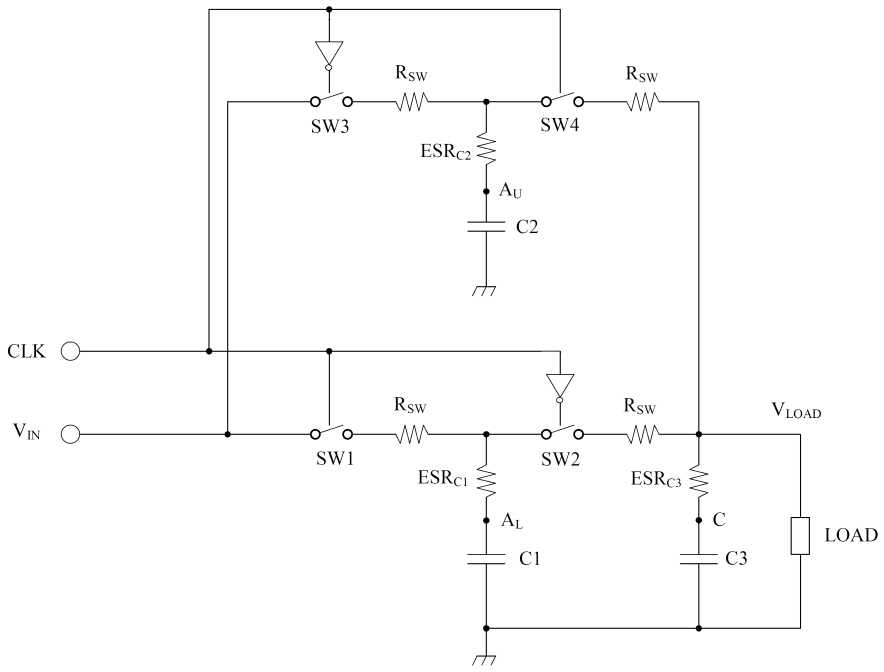


Figure A.5: The inductor-less proposed active filter for differential noise in power lines.

- **CLK=1:** SW1=ON, SW2=OFF, SW3=OFF; SW4=ON

During this half period, the capacitor C1 is loaded with the input voltage V_{IN} according to eq. (A.4), at the same time the fly capacitor C2 is discharged into the load capacitor C3 and the load according to eq. (A.16), eq. (A.18) and eq. (A.20).

- **CLK=0:** SW1=OFF, SW2=ON, SW3=ON; SW4=OFF

During the second half of clock period, C2 is loaded and C1 is discharged into the load capacitor C3 and the load itself. In Table A.2 the complete mathematical model for the filter in Fig. A.5 is reported.

Table A.2: Summary of the decoupling filter model

$Initialization(k=0):$
$v_{A_L}(t_0) = v_{B_L}(t_0) = v_{C_L}(t_0) = v_{A_U}(t_0) = v_{B_U}(t_0) = v_C(t_0) = 0$
$C_1 = C_2$
$R_1 = R_{SW} + ESR_1$
$R_2 = ESR_1 + R_{SW} + ESR_2$
$R_3 = ESR_2 + R_{SW}$
$C = C_2C_3\left(ESR_{C2} + R_{SW}\right)\left(ESR_{C3} + R_{LOAD}\right) + C_2C_3R_{LOAD}ESR_{C3}$
$D = C_2\left(ESR_{C2} + R_{SW}\right) + C_3\left(ESR_{C3} + R_{LOAD}\right) + C_2R_{LOAD}$
$s_1 = \frac{-(D/C)+\sqrt{(D/C)^2-4(1/C)}}{2}; \quad s_2 = \frac{-(D/C)-\sqrt{(D/C)^2-4(1/C)}}{2}; \quad s_3 = -1/(RC_1)$

$t_k < t \leq t_k + \frac{T}{2}; \quad$ k=1,2...

$v_{A_L}(t) = V_{IN}\left[1 - \exp(\frac{-t}{R_1C_1})\right] + v_{A_L}(t_k)\exp(\frac{-t}{R_1C_1})$
$v_{A_U}(t) = v_{B_L}(t_k)\exp\left(ts_3\right) + \frac{R_{LOAD}}{R_3C_2}\left[(h+l)\exp(ts_3) - h\exp(ts_1) - l\exp(ts_2)\right]$
$i_{OUT}\left(t\right) = k_1\exp(s_1t) + k_2\exp(s_2t)$
$v_{OUT} = i_{OUT}R_{LOAD}$
$v_C(t) = v_{OUT}(t) + ESR_{C3}\left[i_{OUT}\left(t\right) - \frac{v_{B_L}(t)-v_{OUT}(t)}{R_{SW}+ESR_{C2}}\right]$

$Where:$
$A = v_C(t_k)C_2C_3\left(ESR_{C2} + R_{SW}\right) + C_2C_3v_{B_L}(t_k)ESR_{C3}$
$B = v_{B_L}(t_k)C_2 + v_C(t_k)C_3$
$k_2 = \left[(B/C) + s_2\left(A/C\right)\right]/\left(s_2 - s_1\right); \quad k_1 = (A/C) - k_2$
$h = -k_1/(s_1 - s_3); \quad l = -k_2/(s_2 - s_3)$

$t_k + \frac{T}{2} < t \leq t_k + T; \quad$ k=1,2...

$v_{A_U}(t) = V_{IN}\left[1 - \exp(\frac{-t}{R_1C_1})\right] + v_{A_U}(t_k + \frac{T}{2})\exp(\frac{-t}{R_1C_1})$
$v_{A_L}(t) = v_{B_U}(t_k)\exp\left(ts_3\right) + \frac{R_{LOAD}}{R_3C_1}\left[(h+l)\exp(ts_3) - h\exp(ts_1) - l\exp(ts_2)\right]$
$i_{OUT}\left(t\right) = k_1\exp(s_1t) + k_2\exp(s_2t)$
$v_{OUT} = i_{OUT}R_{LOAD}$
$v_C(t) = v_{OUT}(t) + ESR_{C3}\left[i_{OUT}\left(t\right) - \frac{v_{B_U}(t)-v_{OUT}(t)}{R_{SW}+ESR_{C2}}\right]$

$Where:$
$A = v_C(t_k + \frac{T}{2})C_5C_3\left(ESR_{C5} + R_{SW}\right) + C_5C_3v_{B_U}(t_k + \frac{T}{2})ESR_{C3}$
$B = v_{B_U}(t_k + \frac{T}{2})C_5 + v_C(t_k + \frac{T}{2})C_3$
$k_2 = \left[(B/C) + s_2\left(A/C\right)\right]/\left(s_2 - s_1\right); \quad k_1 = (A/C) - k_2$
$h = -k_1/(s_1 - s_3); \quad l = -k_2/(s_2 - s_3)$

Using this model, it is possible to analyze the filter behavior in the frequency domain. To accomplish this task, a sinusoidal tone, with an amplitude of $10V_{pp}$ and fixed frequency f, was added to the 5V input voltage. The phase of the tone was randomly set. The ripple in output was then compared with the amplitude of the tone in input, providing in this way the attenuation of the filter at frequency f. Repeating the measure with f going from 0Hz to 1GHz, with 1kHz steps, it was possible to plot the graph in Fig. A.6. The plot refers to the following test conditions: $V_{IN} = 5V$, $R_{LOAD} = 50\Omega$, $C1 = C2 = 10\mu F$ and $ESR = 2.8\Omega$, $C_{LOAD} = 10\mu F$ and $ESR = 0.1\Omega$, $R_{SW} = 2.0\Omega$ and $f_{CK} = 100kHz$. Unlike analog filters made with inductors, it is possible to see the absence of any resonant frequency that must be carefully taken into account. The R_{eq} of the filter strictly depends on the ESR of the capacitors and R_{SW}. Assuming negligible the ESR of the load capacitor, it can be estimated by means of:

$$R_{eq} = \frac{1}{2fC_1} + 2R_{SW} + 2ESR_{C1} \tag{A.21}$$

At the output, we always have the ripple due to the switching capacitors. With the test conditions above, the output ripple is about 1mV. This floor noise gives an asymptotic reachable value for the attenuation. If the input noise has an amplitude of $10V_{pp}$, like in the test conditions above, the maximum possible attenuation is -80dB. For this reason, to achieve good attenuation, the filter is required to have a very low ripple.

The switching frequency comes from a trade off between cut-off frequency and asymptotic attenuation. The higher is the switching frequency, the lower is the output ripple, and therefore the higher is the asymptotic attenuation,

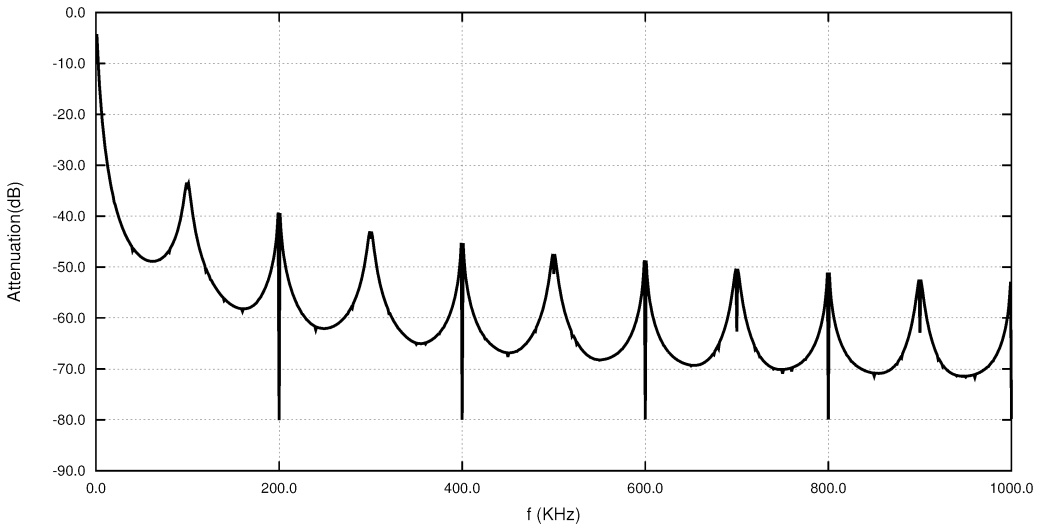


Figure A.6: Frequency response of the proposed active filter with $V_{IN} = 5$V, $R_{LOAD} = 50\Omega$, $C1 = C2 = 10\mu$F and $ESR = 2.8\Omega$, $C_{LOAD} = 10\mu$F and $ESR = 0.1\Omega$, $R_{SW} = 2.0\Omega$ and $f_{CK} = 100$kHz

at the same time, the lower is the switching frequency, the lower is the cut-off frequency. For this reason, the switching frequency should be set in order to achieve the minimum overall output noise, that is given by both intrinsic ripple and noise attenuation. If, for example, it is important to have a good attenuation at low frequencies, a low switching frequency should be considered. In any case, the output ripple can be reduced using bigger capacitors.

Another characteristic of the plot in Fig. A.6 is the presence of notches at multiples of $2f_{CK}$. This is an additional very interesting behavior, because, when the input noise is periodic and it is possible to synchronize the clock of the filter with the fundamental frequency of the noise source, very high attenuation is possible. This case is indeed very common, for example when the noise is due to a PWM switching converter. In this case, the output

voltage is affected by spikes due to the PWM itself. The proposed switched capacitor filter could be used to remove these spikes, in place of using bulk inductors, as depicted in Fig. A.7. In Fig. A.8 the results of a simulation, performed with LTspice IV circuit simulator when the noise in Fig. A.8(a) is added to the input voltage, are shown. In Fig. A.8(b) the output voltage is plotted. The fundamental frequency of the input noise was 100kHz, therefore the clock frequency of the filter was set to 50kHz in order to have fundamental frequency of the noise equal to $2f_{CK}$. With this frequency, the amplitude of the output ripple (floor noise) is about 3mV_{pp}, which means, with an input noise of 10V_{pp}, a maximum attenuation of -70dB is possible. The simulation shows that, at the output of the filter, only the switching ripple is present, and no evidence of the noise in input is present. Using a bigger load capacitor with lower ESR, the output ripple could be easily reduced.

Because of the low ripple and the ability to filter periodic noise, a prototype of the proposed regulator is intended to be used in the measurement instruments proposed in [Gaiotto et al., 2015] and [Gaiotto et al., 2016] where a measurement unit, based on a precise 24-bit ADC, is required to work in the same box with a powerful DC-DC converter.

A.4 Conclusion

In this paper a low output ripple switched-capacitor voltage regulator has been presented. It has also been shown that it has interesting decoupling capabilities and, therefore, how it can be used to supply low-power devices

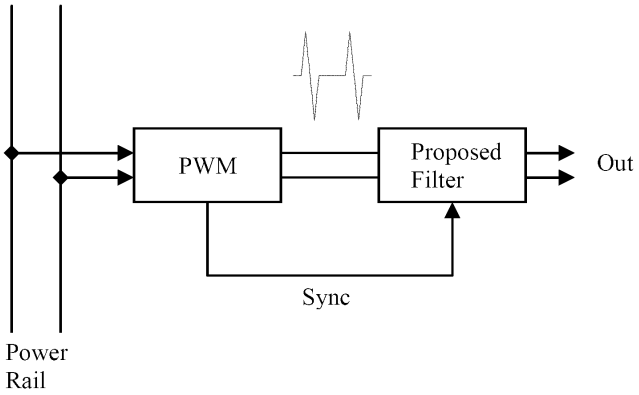


Figure A.7: The proposed active filter synchronized with the noise source.

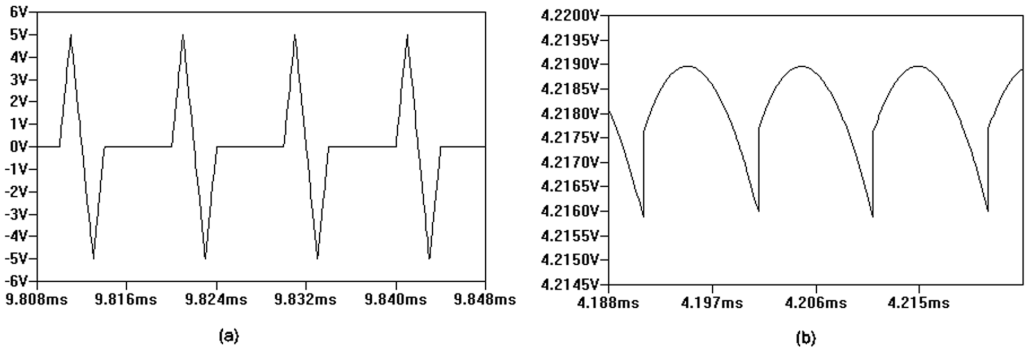


Figure A.8: Filter response with a typical PWM noise. Noise added in input (a), and output ripple (b).

when it is also required to filter heavy EMI noise annoying the system. The proposed solution is preferable over traditional passive decoupling filters because it does not require cumbersome inductors and capacitors, and, other issues related to them, for example resonant phenomena or parasitic effects of the components and PCB tracks, are not possible.

Appendix B

Some source code files

B.1 Hardware library

Listing B.1: ad7714-lib.c

```
#include <stdint.h>
#include <unistd.h>
#include <stdio.h>
#include <stdlib.h>
#include <getopt.h>
#include <fcntl.h>
#include <errno.h>
#include <string.h>
#include <sys/ioctl.h>
#include <sys/time.h>
#include <sys/types.h>
#include <dirent.h>
#include <linux/spi/spidev.h>


/*
CPHA=0: data are captured on the clock's rising edge (low-high transition)
        and data is output on a falling edge (high-low clock transition).
CPHA=1: data are captured on the clock's falling edge and data is output
        on a rising edge.
*/
//static uint8_t mode = 3; //SPI_CPHA | SPI_CPOL;
```

```
static uint8_t mode = SPI_CPHA; //SPI_CPHA=0, SPI_CPOL=0;
static uint8_t bits = 8;
static uint32_t speed = 100000;
static const char *device = "/dev/spidev2.0";
static int fd;
static unsigned long int zeroScaleCalReg[4];
static unsigned long int fullScaleCalReg[4];

static int  IN =0;
static int  OUT=1;

#define     YELLOW  0
#define     GREEN   1
#define     RED     2

#define     VREF    1.225   /* AD1580 */
#define     FCLKIN  2457600
#undef      USE_DRDY_PIN            // DRDY bit in COMM_REG is used


/* ---------------------------------------------------------------------- */
/*      GPIO pins                                                         */
/* ---------------------------------------------------------------------- */

#define     GPIO_RST        115     // P9_27
#define     GPIO_LED_RED    30      // P9_11
#define     GPIO_LED_YG     31      // P9_13
#define     GPIO_RHI        14      // P9_23
#define     GPIO_RLO        49      // P9_26
#define     GPIO_FAN        48      // P9_15
#define     GPIO_OFF        60      // P9_12

/* ---------------------------------------------------------------------- */
/*      Filter High Register                                              */
/* ---------------------------------------------------------------------- */

#define     BIPOLAR         0x00
#define     UNIPOLAR        0x80
#define     WL24            0x40
#define     WL16            0x00
#define     BOOSTON         0x20
#define     BOOSTOFF        0x00


/* ---------------------------------------------------------------------- */
/*      Communications Register                                           */
/* ---------------------------------------------------------------------- */

#define     CH0         0x01
#define     CH1         0x02
#define     CH2         0x04
#define     RNOTW       0x08
```

```
#define     RS0             0x10
#define     RS1             0x20
#define     RS2             0x40
#define     NOTREADY        0x80

#define     COMM_REG        0x00
#define     MODE_REG        RS0
#define     FILTERH_REG     RS1
#define     FILTERL_REG     RS1|RS0
#define     TEST_REG        RS2
#define     DATA_REG        RS2|RS0
#define     ZEROSCALE_REG   RS2|RS1
#define     FULLSCALE_REG   RS2|RS1|RS0

#define     WRITE_OP        0x00
#define     READ_OP         0x08


/* ------------------------------------------------------------------------ */
/*      Mode Register                                                       */
/* ------------------------------------------------------------------------ */

#define     FSYNC           0x01
#define     BO              0x02
#define     G0              0x04
#define     G1              0x08
#define     G2              0x10
#define     MD0             0x20
#define     MD1             0x40
#define     MD2             0x80

#define     NORMAL_MODE             0x00
#define     SELF_CAL                MD0
#define     ZERO_SCALE_SYSTEM_CAL   MD1
#define     FULL_SCALE_SYSTEM_CAL   MD1|MD0
#define     SYSTEM_OFFSET_CAL       MD2
#define     BACKGROUND_CAL          MD2|MD0
#define     ZERO_SCALE_SELF_CAL     MD2|MD1
#define     FULL_SCALE_SELF_CAL     MD2|MD1|MD0



/* ------------------------------------------------------------------------ */
/*      delayms                                                             */
/* ------------------------------------------------------------------------ */

void delayms(double ms)
{
    usleep(ms*1000);
}
```

```
void pabort(const char *s)
{
    perror(s);
    abort();
}


/* ------------------------------------------------------------------------ */
/*      openGPIOpin                                                         */
/* ------------------------------------------------------------------------ */

int openGPIOpin(int gpiopin, int dir)
{
    char str[256];
    FILE *wf;

    wf = fopen("/sys/class/gpio/export", "w");
    if (wf == NULL) {
        printf("Error opening /sys/class/gpio/export\n\r");
        return errno;
    }

    fprintf(wf,"%d", gpiopin);
    fclose(wf);

    sprintf(str, "/sys/class/gpio/gpio%d/direction", gpiopin);
    wf = fopen(str, "w");
    if (wf == NULL) {
        printf("Error opening %s\n\r", str);
        return errno;
    }

    if (dir==IN)
        fprintf(wf,"in");
    else
        fprintf(wf,"out");
    fclose(wf);
    return 0;
}


/* ------------------------------------------------------------------------ */
/*      writeGPIOpin                                                        */
/* ------------------------------------------------------------------------ */

int  writeGPIOpin(int gpiopin, int value)
{
    char str[256];
    FILE *wf;

    sprintf(str, "/sys/class/gpio/gpio%d/value", gpiopin);
```

```
    wf = fopen(str, "w");
    if (wf == NULL) {
        printf("Error opening %s\n\r", str);
        return errno;
    }

    fprintf(wf,"%d", value);
    fclose(wf);
    return 0;
}


/* ------------------------------------------------------------------------ */
/*      readGPIOpin                                                         */
/* ------------------------------------------------------------------------ */

int  readGPIOpin(int gpiopin)
{
    char str[256];
    FILE *wf;
    char value;

    sprintf(str, "/sys/class/gpio/gpio%d/value", gpiopin);

    wf = fopen(str, "r");
    if (wf == NULL) {
        printf("Error opening %s\n\r", str);
        return errno;
    }

    fscanf(wf,"%c", &value);
    fclose(wf);

    return (value-48);
}

/* ------------------------------------------------------------------------ */
/*      initGPIOpins                                                        */
/* ------------------------------------------------------------------------ */

 /************************/
 /* P9 Header            */
 /************************/

 /* P9_01                GND     */
 /* P9_02                GND     */
 /* P9_03                3.3V    */
 /* P9_04                3.3V    */
 /* P9_05                VDD_5V  */
 /* P9_06                VDD_5V  */
 /* P9_07                SYS_5V  */
 /* P9_08                SYS_5V  */
```

```
 /* P9_09                PWR_BUT */
 /* P9_10 (ZCZ ball A10) RESETn  */

 /* P9_11   OUT     LED RED          */
 /* P9_12   IN      OFF              */
 /* P9_13   OUT     LED GREEN        */
 /* P9_14   -       EXP. HEADER      */
 /* P9_15   OUT     FAN              */
 /* P9_16   -       EXP. HEADER      */

 /* P9_20   -       EXP. HEADER      */
 /* P9_22   PWM     OUT              */
 /* P9_23   OUT     BANK_1 (HI)      */
 /* P9_24   -       EXP. HEADER      */
 /* P9_26   OUT     BANK_0 (LOW)     */
 /* P9_27   OUT     AD7714 - RESET   */
 /* P9_29   IN      AD7714 - MISO    */
 /* P9_30   OUT     AD7714 - MOSI    */
 /* P9_31   OUT     AD7714 - CLK     */

int initGPIOpins(void)
{
    int rc=0;

    rc  = openGPIOpin(GPIO_LED_RED, OUT);
    rc  = writeGPIOpin(GPIO_LED_RED, 0);     // OFF;

    rc |= openGPIOpin(GPIO_OFF, IN);

    rc |= openGPIOpin(GPIO_LED_YG, OUT);
    rc |= writeGPIOpin(GPIO_LED_YG, 1); // GREEN

    rc |= openGPIOpin(GPIO_FAN, OUT);
    rc |= writeGPIOpin(GPIO_FAN, 1);     // OFF;

    rc |= openGPIOpin(GPIO_RST, OUT);
    rc |= writeGPIOpin(GPIO_RST, 1);     // RESET = 0;

    rc |= openGPIOpin(GPIO_RLO, OUT);
    writeGPIOpin(GPIO_RLO, 1);       // Disconnected

    rc |= openGPIOpin(GPIO_RHI, OUT);
    writeGPIOpin(GPIO_RLO, 0);       // Connected

    return rc;
}

/* ------------------------------------------------------------------------ */
/*      setYellowGreenLED                                                   */
/* ------------------------------------------------------------------------ */

void setYellowGreenLED(int color)
```

```
{
    if (color == YELLOW)
        writeGPIOpin(GPIO_LED_YG, 0);
    else
        writeGPIOpin(GPIO_LED_YG, 1);
}


/* ------------------------------------------------------------------------ */
/*      setRedLED                                                           */
/* ------------------------------------------------------------------------ */

void setRedLED(int status)
{
    writeGPIOpin(GPIO_LED_RED, status);
}


/* ------------------------------------------------------------------------ */
/*              getTimeStamp                                                */
/* ------------------------------------------------------------------------ */

int getTimeStamp(volatile unsigned long long *time)
{
    struct timeval tv;
    int rc;

    rc = gettimeofday(&tv, (struct timezone *) NULL);
    *time = (1000000 * tv.tv_sec + tv.tv_usec);
    return rc;
}


/* ------------------------------------------------------------------------ */
/*  findPWM                 */
/* ------------------------------------------------------------------------ */

/*
int findPWMchip(void)
{
    DIR *wd;
    char str[256];
    int i=-1;
    struct dirent *dir;

    wd = opendir("/sys/class/pwm");
    if (wd==NULL) {
        printf("ERROR: Unable to open /sys/class/pwm, errno =%d\n", errno);
        return -1;
    }

    dir = readdir(wd);
```

```
    printf("str = %s\n", dir->d_name);
        dir = readdir(wd);
    printf("str = %s\n", dir->d_name);
        dir = readdir(wd);
    printf("str = %s\n", dir->d_name);
//  while (fgets(str, strlen(str), wf) != NULL) {
//  struct dirent *readdir(DIR *dirp);
//      i++;
//      printf("str = %s\n", str);
//      if (strstr(str, "pwmchip0") != NULL) break;
//  }

    closedir(wd);
    return i;
}
*/

/* ------------------------------------------------------------------------ */
/*      set_pwm                                                             */
/* ------------------------------------------------------------------------ */

/*
*** Linux Kernel 4.22 ****
*/

int set_pwm(int pwmchip, float f, float d, int enable) {
    unsigned long int period, duty_cycle;
    int polarity=0;
    FILE *wf;
    char str[256];

    period=(1E6/f); // period in ns
    duty_cycle=period*(d);

    // esportiamo nel sys il primo ePWM */
    sprintf(str, "/sys/class/pwm/pwmchip%d/export", pwmchip);
    wf = fopen(str, "w");
    if (wf == NULL) {
        printf("Error opening export\n\r");
        return errno;
    }

    fprintf(wf,"0");
    fclose(wf);

    // run=0 (spegniamo il PWM prima di modificarne i parametri
    sprintf(str, "/sys/class/pwm/pwmchip%d/pwm0/enable", pwmchip);
    wf = fopen(str, "w");
    if (wf == NULL) {
        printf("Error opening enable\n\r");
        return errno;
    }
```

```
    fprintf(wf,"0");
    fclose(wf);

    // period
    sprintf(str, "/sys/class/pwm/pwmchip%d/pwm0/period", pwmchip);
    wf = fopen(str, "w");
    if (wf == NULL) {
        printf("Error opening period\n\r");
        return errno;
    }

    fprintf(wf,"%lu", period);
    fclose(wf);

    // duty cicle
    sprintf(str, "/sys/class/pwm/pwmchip%d/pwm0/duty_cycle", pwmchip);
    wf = fopen(str, "w");
    if (wf == NULL) {
        printf("Error opening duty_cycle\n\r");
        return errno;
    }

    fprintf(wf,"%lu", duty_cycle);
    fclose(wf);

    // polarity
    sprintf(str, "/sys/class/pwm/pwmchip%d/pwm0/polarity", pwmchip);
    wf = fopen(str, "w");
    if (wf == NULL) {
        printf("Error opening polarity\n\r");
        return errno;
    }

    fprintf(wf,"%d", polarity);
    fclose(wf);

    // run
    sprintf(str, "/sys/class/pwm/pwmchip%d/pwm0/enable", pwmchip);
    wf = fopen(str, "w");
    if (wf == NULL) {
        printf("Error opening enable\n\r");
        return errno;
    }

    fprintf(wf,"%d", enable);
    fclose(wf);

    return 0;
}
```

```
/* ------------------------------------------------------------------------ */
/*      initSPIport                                                         */
/* ------------------------------------------------------------------------ */

static unsigned int initSPIport(const char *device )
{
    int ret;

    fd = open(device, O_RDWR);
    if (fd<=0) {
        printf("Device %s not found\n", device);

        exit(1);
    }

    /*
     * spi mode
     */
    ret = ioctl(fd, SPI_IOC_WR_MODE, &mode);
    if (ret == -1)
        pabort("can't set spi mode");

    ret = ioctl(fd, SPI_IOC_RD_MODE, &mode);
    if (ret == -1)
        pabort("can't get spi mode");

    /*
     * bits per word
     */
    ret = ioctl(fd, SPI_IOC_WR_BITS_PER_WORD, &bits);
    if (ret == -1)
        pabort("can't set bits per word");

    ret = ioctl(fd, SPI_IOC_RD_BITS_PER_WORD, &bits);
    if (ret == -1)
        pabort("can't get bits per word");

    /*
     * max speed hz
     */
    ret = ioctl(fd, SPI_IOC_WR_MAX_SPEED_HZ, &speed);
    if (ret == -1)
        pabort("can't set max speed hz");

    ret = ioctl(fd, SPI_IOC_RD_MAX_SPEED_HZ, &speed);
    if (ret == -1)
        pabort("can't get max speed hz");

    printf("spi mode: %d\n", mode);
    printf("bits per word: %d\n", bits);
    printf("max speed: %d Hz (%d KHz)\n", speed, speed/1000);
```

```
    return 0;
}


/* ------------------------------------------------------------------------ */
/*      trasfer                                                             */
/* ------------------------------------------------------------------------ */

static int transfer(int fd, unsigned char send[], unsigned char receive[], \
                    int length)
{
    struct spi_ioc_transfer transfer;
    transfer.tx_buf = (unsigned long) send;
    transfer.rx_buf = (unsigned long) receive;
    transfer.len = length;
    transfer.speed_hz = speed;
    transfer.bits_per_word = 8;
    transfer.delay_usecs = 0;
    transfer.pad=0;

    int status = ioctl(fd, SPI_IOC_MESSAGE(1), &transfer);
    if (status<0) {
        perror("WSPI_IOC_MESSAGE Failed");
        return -1;
    }
    return 0;
}


/* ------------------------------------------------------------------------ */
/*      readSPI8                                                            */
/* ------------------------------------------------------------------------ */

static unsigned char readSPI8(void)
{
    unsigned char uc=0;
    transfer(fd, NULL, &uc,1);
    return uc;
}


/* ------------------------------------------------------------------------ */
/*      readSPI16                                                           */
/* ------------------------------------------------------------------------ */
/*
static unsigned short int readSPI16(void)
{
    unsigned char uc[2];
    transfer(fd, NULL, uc, 2);
    return uc[1]+((unsigned short int)(uc[0])*256);
}
*/
```

```
/* ------------------------------------------------------------------------ */
/*      readSPI24                                                           */
/* ------------------------------------------------------------------------ */

static unsigned long int readSPI24(void)
{
    unsigned char uc[3];
    unsigned long int val;

    //waitForReady();
    transfer(fd, NULL, uc, 3);
    val = uc[2] + ((unsigned long int)(uc[1])<<8) + \
                  ((unsigned long int)(uc[0])<<16);
    return val;
}


/* ------------------------------------------------------------------------ */
/*      writeSPI8                                                           */
/* ------------------------------------------------------------------------ */

static int writeSPI8(unsigned char b)
{
    if (write(fd, &b, 1) != 1) {
        perror("Write Error");
        return -1;
    }

    fsync(fd);

//  transfer(fd, &b, NULL, 1);
    return 0;
}


/* ------------------------------------------------------------------------ */
/*      writeSPI24                                                          */
/* ------------------------------------------------------------------------ */

static int writeSPI24(unsigned long int val)
{
    unsigned char uc[3];

    uc[2] = (val & 0x0000FF);
    uc[1] = (val & 0x00FF00) >> 8;
    uc[0] = (val & 0xFF0000) >> 16;

    //printf("uc[0]=0x%x, uc[1]=0x%x, uc[2]=0x%x\n", uc[0], uc[1], uc[2]);

    transfer(fd, uc, NULL, 3);
    return 0;
```

```
}


/* ------------------------------------------------------------------------ */
/*      waitForReady                                                        */
/* ------------------------------------------------------------------------ */

/* #DRDY: A logic low on this output indicates that a new output word is
available from the AD7714 data register. The DRDY pin will return high upon
completion of a read operation of a full output word. If no data read has
taken place, after an output update, the DRDY line will return high for
500 x tCLK_IN cycles prior to the next output update. This gives an
indication of when a read operation should not be attempted to avoid reading
from the data register as it is being updated.
DRDY is also used to indicate when the AD7714 has completed its on-chip
calibration sequence.
*/

void waitForReady(unsigned char ch)
{
    unsigned char uc=0;

#ifdef USE_DRDY_PIN
    do {
        uc=readGPIOpin(GPIO_DRDY);
    } while (uc!=0);
#else

    do {
        writeSPI8(COMM_REG | READ_OP | ch);
        transfer(fd, NULL, &uc, 1);
        //uc=readSPI8();
    } while ((uc & NOTREADY)==0x80);
#endif
}


/* ------------------------------------------------------------------------ */
/*      encodeChannel                                                       */
/* ------------------------------------------------------------------------ */

static unsigned char encodeChannel(unsigned char channel)
{
    return ((channel-1)&0x03);
}


/* ------------------------------------------------------------------------ */
/*      setupAD7714ch                                                       */
/* ------------------------------------------------------------------------ */

int setupAD7714ch(unsigned char channel, unsigned char gain, \
```

```
                    unsigned short fnotch)
{
    unsigned char val;
    unsigned short int code;
    unsigned long int val24;
    unsigned char ch, val8, mode_reg;

    printf("chnnel =%d, gain = %d, fnotch=%d\n", channel, gain,fnotch);

    ch = encodeChannel(channel);


    // --- FILTER HIGH ------------------------------------------------------
    /* filter first notch frequency = (FCLKIN/128)/code */
    code = (FCLKIN/128)/fnotch;
    printf("fnotch = %d, code = %d\n", fnotch, code);

    if ((code<19) || (code>4000)) {
        printf("Error: invalid first notch frequency, code = %d\n", code);
        return -1;
    }

    /* set next operation as write to the filter high register (0x27)*/
    writeSPI8(FILTERH_REG | WRITE_OP | ch);

    /* set Bipolar mode, 24 bits, boost off,
       all 4 MSBs of filterword to 1(0x4f) */
    if ((gain>8) && (FCLKIN==2457600))
        val8 = UNIPOLAR | WL24 | BOOSTON | (code >> 8);
    else
        val8 = UNIPOLAR | WL24 | BOOSTOFF | (code >> 8);

    writeSPI8(val8);


    // --- FILTER LOW -------------------------------------------------------
    /* set the next operation as a write to the filter low register */
    writeSPI8(FILTERL_REG | WRITE_OP | ch);

    /* max filter word allowed for low part of the filterword */
    val8 = code & 0x00FF;
    writeSPI8(val8);
    //printf("Filter low register = 0x%x\n", val8);


    /* set gain, burnout current off, no filter sync, and do
       a self calibration (0x20) */
    switch (gain)
    {
        case 1:
            val=0;
            break;
```

```
        case 2:
            val=1;
            break;
        case 4:
            val=2;
            break;
        case 8:
            val=3;
            break;
        case 16:
            val=4;
            break;
        case 32:
            val=5;
            break;
        case 64:
            val=6;
            break;
        case 128:
            val=7;
            break;
        default: {
            printf("Invalid GAIN %d\n\r", gain);
            return 0;
        }
    }

    // --- MODE REG -----------------------------------------------------
    /* set the operation as a write to the mode register (0x17) */
    val8 = MODE_REG | WRITE_OP | ch;
    writeSPI8(val8);
    //printf("Select write filter mode register = 0x%x\n", val8);

    mode_reg = SELF_CAL | (val<<2);
    //printf("mode register = 0x%x\n", mode_reg);
    writeSPI8(mode_reg);

    // wait for self calibration end
    waitForReady(ch);

    // stampa registri zero-scale and full-scale calibration registers
    writeSPI8(ZEROSCALE_REG | READ_OP | ch);
    val24 = readSPI24();
    printf("Zero-scale calibration register = 0x%lx\n", val24);
    zeroScaleCalReg[channel-1]=val24;

    writeSPI8(FULLSCALE_REG | READ_OP | ch);
    val24 = readSPI24();
    printf("Full-scale calibration register = 0x%lx\n", val24);
    fullScaleCalReg[channel-1]=val24;

    // Stop data acquisition
```

```
    writeSPI8(MODE_REG | WRITE_OP | ch);
    //writeSPI8(mode_reg | FSYNC);
    writeSPI8(NORMAL_MODE | FSYNC);

    printf("AD7714 channel %d ready!\n\r", channel);
    return 0;
}


/* ------------------------------------------------------------------------ */
/*      startConvertion                                                     */
/* ------------------------------------------------------------------------ */

int startConvertion(unsigned char channel)
{
    unsigned char ch;
    int err=0;

    ch = encodeChannel(channel);
    writeSPI8(MODE_REG | WRITE_OP | ch);
    writeSPI8(NORMAL_MODE); // Clear FSYNC bit
    return err;
}


/* ------------------------------------------------------------------------ */
/*      stopConvertion                                                      */
/* ------------------------------------------------------------------------ */

int stopConvertion(unsigned char channel)
{
    unsigned char ch;
    int err=0;

    ch = encodeChannel(channel);
    writeSPI8(MODE_REG | WRITE_OP | ch);
    writeSPI8(NORMAL_MODE | FSYNC); // Set FSYNC bit
    return err;
}


/* ------------------------------------------------------------------------ */
/*      getAD7714sample                                                     */
/* ------------------------------------------------------------------------ */

unsigned long int getAD7714sample(unsigned char channel)
{
    unsigned long int sample;
    unsigned char ch,uc;

    ch = encodeChannel(channel);
```

```
    // load zero-scale and full-scale cal. reg. when channel=3 or 4
    if ((channel==3) || (channel==4)) {
        writeSPI8(FULLSCALE_REG | WRITE_OP | ch);
        writeSPI24(fullScaleCalReg[channel-1]);

        writeSPI8(ZEROSCALE_REG | WRITE_OP | ch);
        writeSPI24(zeroScaleCalReg[channel-1]);
    }

    startConvertion(channel);

    // Data sample reading (discarding the first two samples)
    waitForReady(ch);
    writeSPI8(DATA_REG | READ_OP | ch);
    sample = readSPI24();
    waitForReady(ch);
    writeSPI8(DATA_REG | READ_OP | ch);
    sample = readSPI24();
    waitForReady(ch);
    writeSPI8(DATA_REG | READ_OP | ch);
    sample = readSPI24();

    // stop ADC operations
    stopConvertion(channel);

    /* Flush DATA reg */
    do {
        writeSPI8(COMM_REG | READ_OP | ch);
        transfer(fd, NULL, &uc, 1);
        // flush
        if ((uc & NOTREADY)==0x00) readSPI24();
    } while ((uc & NOTREADY)==0x00);

    return sample;
}


/* ------------------------------------------------------------------------ */
/*      initAD7714                                                          */
/* ------------------------------------------------------------------------ */

unsigned int initAD7714(unsigned char gain, unsigned short fnotch)
{
    unsigned long int val24;
    unsigned char ch, channel;
    int rc;

    printf("\n\rSetup GPIOs...\n\r");

    // RESET
    rc = openGPIOpin(GPIO_RST, OUT);
    rc |= writeGPIOpin(GPIO_RST, 0);     // RESET = 1;
```

```
#ifdef USE_DRDY_PIN
    // #DRDY
    rc |= openGPIOpin(GPIO_DRDY, IN);
#endif

    if (rc!=0) {
        printf("Error 0x%x while initializing GPIO pins...\n", rc);
        return rc;
    }

    printf("\n\rSetup SPI port...\n\r");
    rc = initSPIport(device);
    if (rc!=0) {
        printf("Error 0x%x while initializing SPI port...\n", rc);
        return rc;
    }

    // RESET AD7714
    printf("Resetting AD7714...\n\r");
    writeGPIOpin(GPIO_RST, 1);
    delayms(100);
    writeGPIOpin(GPIO_RST, 0);
    delayms(100);
    writeGPIOpin(GPIO_RST, 1);
    delayms(100);

    // Zero-scale and Full-scale calibration registers
    for (channel=1; channel<=4; channel++) {
        ch = encodeChannel(channel);

        writeSPI8(ZEROSCALE_REG | READ_OP | ch);
        val24 = readSPI24();
        if (val24 != 0x1F4000) {
            printf("ERROR: Zero-scale calibration register = 0x%lx\n", val24);
            printf("expected 0x1F4000\n");
            return -1;
        }

        printf("Zero-scale cal. reg. of ch%d OK!\n", channel);

        writeSPI8(FULLSCALE_REG | READ_OP | ch);
        val24 = readSPI24();
        if (val24 != 0x5761AB)  {
            printf("ERROR: Full-scale calibration register = 0x%lx\n", val24);
            printf("expected 0x5761AB\n");
            return -1;
        }

        printf("Full-scale calibration of ch%d register OK!\n", channel);
    }
```

```
    printf("\n\rSetup AD7714 channels...\n\r");
    setupAD7714ch(1, gain, fnotch);
    setupAD7714ch(2, gain, fnotch);
    setupAD7714ch(3, gain, fnotch);
    setupAD7714ch(4, gain, fnotch);

    return rc;
}
```

¡!—-¿

B.2 Characteristic curves tracer

Listing B.2: tracer.c

```
#include <stdint.h>
#include <unistd.h>
#include <stdio.h>
#include <stdlib.h>
#include <getopt.h>
#include <fcntl.h>
#include <errno.h>
#include <string.h>
#include <sys/ioctl.h>
#include <linux/spi/spidev.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <unistd.h>
#include <fcntl.h>
#include <sys/mman.h>
#include <sys/time.h>
#include <math.h>
//#include <neon.h>

#define     MAX_VOUT    150     /* Vds_max di IPP320N20N3 = 200V */
#define     IMAX        7.0     // RMBF: Parametro da cmd line?
#define     VREF        1.225   /* AD1580 */

#define     GPIO_RHI    14      // P9_26
#define     GPIO_RLO    49      // P9_23

#define     RLOAD_ON    0
#define     RLOAD_OFF   1

#define     V_IN        0
#define     I_IN        1

#define     IN          0
#define     OUT         1

#define     YELLOW  0
#define     GREEN   1
#define     RED     2

#define     RLOAD_HI    6*47
#define     RLOAD_LOW   6*4.7

#define     YELLOW  0
#define     GREEN   1
```

```
#define     RED     2

float f_pwm; // frequenza PWM in KHz
int pwmchip=0;
int gain;
int Np=1, Ns;

extern unsigned int initAD7714(unsigned char gain, unsigned short fnotch);
extern unsigned long int getAD7714sample(unsigned char channel);
extern int openGPIOpin(int gpiopin, int dir);
extern int writeGPIOpin(int gpiopin, int value);
extern int writeGPIOpin(int gpiopin, int value);
extern int set_pwm(int pwmchip, float f, float d, int enable);
extern void delayms(double ms);
extern void setRedLED(int status);
extern void setYellowGreenLED(int color);
extern int findPWMchip(void);


/* ------------------------------------------------------------------------ */
/*      getSample                                                           */
/* ------------------------------------------------------------------------ */

float getSample(int ch, float gain) {
    unsigned long int val24;
    float value;


    if (ch == V_IN) {
        val24 = getAD7714sample(1);
        value = 51*((float)val24/0x01000000)*(VREF/gain);
    } else {
        val24 = getAD7714sample(2);
        // i = Vsense / Rsense = (V/GainAmp)/Rsense = (V/11)/(0.01) = V*20
        value = (1.0/0.11)*((float)val24/0x01000000)*(VREF/gain);
    }

//  printf("Value = 0x%x ", val24);

    return value;
}


/* ------------------------------------------------------------------------ */
/*      main                                                                */
/* ------------------------------------------------------------------------ */

int main (int argc, char *argv[]) {
    float Voc, voltage, current, Vout, D;
    int i,rc,rl,nsample,delay;
    float rload;
    FILE *wf;
```

```
unsigned int fnotch;

if (argc != 6) {
     printf("Usage:\n\r");
     printf("tracer Rload[0|1] nsample delay(ms) fnotch pwmchip\n\r");
     printf("\n\r");
     return 0;
}

rl = atoi(argv[1]);
nsample = atoi(argv[2]);
delay = atoi(argv[3]);
fnotch = atoi(argv[4]);
pwmchip=atoi(argv[5]);

if ((rl!=0) && (rl!=1)) {
     printf("Invalid Rload %d, should be 0 (low) or 1 (high)\n\r", rl);
     return 0;
}

gain = 1;
f_pwm = 100;

/* --- init GPIO pins --- */
openGPIOpin();

/* --- no error condition detected --- */
setRedLED(0);

/* ---system is BUSY --- */
setYellowGreenLED(YELLOW);


/* --- PWM off ---------------------- */
//pwmchip = getCurrentPWMchip();
printf("PWM chip found %d\n", pwmchip);
set_pwm(pwmchip, f_pwm, 0.1, 0);

printf("\n\rSetup AD7714...\n\r");
rc = initAD7714(gain, fnotch);
if (rc==-1) return 0;

/* --- Voc --------------------- */
writeGPIOpin(GPIO_RLO, RLOAD_OFF);
writeGPIOpin(GPIO_RHI, RLOAD_OFF);
Voc = getSample(V_IN, gain);
current = getSample(I_IN, gain);
printf("Voc = %f, Ioc = %f\n", Voc, current);

// Open-circuit entry in data file
wf = fopen("misure.txt", "w");
D=0;
```

```
voltage=Voc;
//current=0;
i=0;
printf("i=%d, D=%f, V = %f, I = %f, P = %f\n", \
        i, D, voltage, current, voltage*current);
fprintf(wf, "%d %2.4f %2.4f %2.4f %2.4f\n", \
        i, D, voltage, current, voltage*current);

/* --- setup Rload */
printf("\n\rSetup Rload...\n\r");

if (rl==0) {
    writeGPIOpin(GPIO_RLO, RLOAD_ON);
    writeGPIOpin(GPIO_RHI, RLOAD_OFF);
    rload=RLOAD_LOW;
    printf("Rload = LOW (%f ohm)...\n\r", rload);
} else {
    writeGPIOpin(GPIO_RHI, RLOAD_ON);
    writeGPIOpin(GPIO_RLO, RLOAD_OFF);
    rload=RLOAD_HI;
    printf("Rload = HIGH (%f ohm)...\n\r", rload);
}

/* --- nsample acquisition ---------------- */
for(i=1; i<nsample; i++) {
    // duty cycle
    D = 2.0*(nsample-1)/(i+nsample-1);       // D in [0-,..,1]
    D = (float)i/nsample;                    // D unif in [0-,..,1]

    // Vout must be lower than MAX_VOUT
    Vout=sqrt(voltage*current*rload); // Vout estimation
    if (Vout>MAX_VOUT) {
        printf("ERROR! Estimated Vout = %f\n", Vout);
        setRedLED(1);
        break;
    }

    // set PWM
    set_pwm(pwmchip, f_pwm, D, 1);

    // wait for stable condition
    delayms(delay);

    // get samples
    voltage = getSample(V_IN, gain);
    current = getSample(I_IN, gain);

    // print
    printf("i=%d, D=%f, V = %f, I = %f, P = %f, Vout(est.)=%f\n",
            i, D, voltage, current, voltage*current, Vout);
    fprintf(wf, "%d %2.4f %2.4f %2.4f %2.4f\n", \
            i, D, voltage, current, voltage*current);
```

```
            if (current>IMAX) {
                printf("ERROR! I= %f\n", current);
                setRedLED(1);
                break;
            }
        }

        // turn off PWM
        set_pwm(pwmchip, f_pwm, 0, 0);

        // disable internal load
        writeGPIOpin(GPIO_RHI, RLOAD_OFF);
        writeGPIOpin(GPIO_RLO, RLOAD_OFF);

        // system is READY
        setYellowGreenLED(GREEN);

        fclose(wf);
        return 0;
}
```

¡!—-¿

B.3 PV panels identification routine library

Listing B.3: ident5lib.c

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <string.h>
#include <levmar.h>

#define VT   1.3806503E-23*(273.15+Tref)/(1.602E-19)

static  double Isc_Ref;
static  double Voc_Ref;
static  double Imp_Ref;
static  double Vmp_Ref;
static  double Tref;        // Temperature (Celsius)
static  int Np;             // number of cells connected in parallel
static  int Ns;             // number of cells connected in series
static  int nsample;
static  double *dataV, *dataI;
static  int tipofr;



/* ------------------------------------------------------------------------- */
/*       Reduced Form #1                                                     */
/* ------------------------------------------------------------------------- */

double Vpanel3Eq(double *v5)
{
    double EXPoc,EXPmpp,D,I0,Gp,Rp,Iirr;
    double n,Rs;

    n=v5[0];
    Rs=v5[1];

    EXPoc=exp(Voc_Ref/((n*VT)*Ns));
    EXPmpp=exp((Vmp_Ref+Rs*Imp_Ref)/((n*VT)*Ns));

    D=(Rs*Imp_Ref-Vmp_Ref)*\
        (EXPmpp*(Imp_Ref*Rs+Vmp_Ref-Voc_Ref-Ns*n*VT)+EXPoc*Ns*n*VT);

    I0 = Imp_Ref*(Voc_Ref-2*Vmp_Ref)*Ns*n*VT;
    I0 = I0/D;
    Iirr = Imp_Ref*(Voc_Ref*(Vmp_Ref-Imp_Ref*Rs+Ns*n*VT)*EXPmpp+Ns*n*VT*\
            (2*Vmp_Ref*(1-EXPoc)-Voc_Ref));
    Iirr = Iirr/D;
```

```
    Gp = Imp_Ref*((Vmp_Ref+Ns*n*VT-Imp_Ref*Rs)*EXPmpp-Ns*n*VT*EXPoc);
    Gp = Gp/D;
    Rp=1.0/Gp;

    v5[2]=I0;
    v5[3]=Iirr;
    v5[4]=Rp;

    return 0;
}



/* ------------------------------------------------------------------------ */
/*       Reduced Form #2                                                    */
/* ------------------------------------------------------------------------ */

static double Vpanel3Eq_2(double *v5)
{
    double A1,A2,A3,E1,E2,E3,P1,P2,P3,P4,Io,Gp,Rp,Iirr_ref;
    double n,Rs;

    n=v5[0];
    Rs=v5[1];

    A1 = (Rs*Imp_Ref)+Vmp_Ref-Voc_Ref;
    A2 = Voc_Ref-(Rs*Isc_Ref);
    A3 = (Rs*Isc_Ref)-(Rs*Imp_Ref)-Vmp_Ref;

    E1=exp(Voc_Ref/((n*VT)*Ns));
    E2=exp((Rs*Isc_Ref)/(n*VT*Ns));
    E3=exp((Vmp_Ref+Rs*Imp_Ref)/((n*VT)*Ns));

    P1 = Imp_Ref*Vmp_Ref;
    P2 = Imp_Ref*(Voc_Ref-Vmp_Ref);
    P3 = (Isc_Ref-Imp_Ref)*(Voc_Ref-Vmp_Ref);
    P4 = (Isc_Ref-Imp_Ref)*Vmp_Ref;

    Io = ((-Isc_Ref*(Vmp_Ref-Voc_Ref))-\
          (Imp_Ref*Voc_Ref))/((A1*E2)+(A2*E3)+(A3*E1));
    Gp = (((Imp_Ref-Isc_Ref)*E1)+(Isc_Ref*E3)-\
          (Imp_Ref*E2))/((A1*E2)+(A2*E3)+(A3*E1));
    Rp=1.0/Gp;
    Iirr_ref = Io*(E1 - 1.0) + (Voc_Ref*Gp);

    v5[2]=Io;
    v5[3]=Iirr_ref;
    v5[4]=Rp;

    return 0;
}
```

```
/* ------------------------------------------------------------------------ */
/*       lambertW0 function, main branch                                    */
/* ------------------------------------------------------------------------ */

double LambertW(const double z) {
    int i;
    const double eps=4.0e-16, em1=0.3678794411714423215955237701614608;
    double p,e,t,w;
    /*
    if (dbgW) fprintf(stderr,"LambertW: z=%g\n",z);
    if (z<-em1 || isinf(z) || isnan(z)) {
        fprintf(stderr,"LambertW: bad argument %g, exiting.\n",z); exit(1);
    }
    */
    i=0;
    if (0.0==z) return 0.0;
    if (z<-em1+1e-4) { // series near -em1 in sqrt(q)
        double q=z+em1,r=sqrt(q),q2=q*q,q3=q2*q;
        return
                -1.0
                +2.331643981597124203363536062168*r
                -1.812187885639363490240191647568*q
                +1.936631114492359755363277457668*r*q
                -2.353551201881614516821543561516*q2
                +3.066858901050631912893148922704*r*q2
                -4.175335600258177138854984177460*q3
                +5.858023729874774148815053846119*r*q3
                -8.401032217523977370984161688514*q3*q;  // error approx 1e-16
    }
    /* initial approx for iteration... */
    if (z<1.0) { /* series near 0 */
        p=sqrt(2.0*(2.7182818284590452353602874713526625*z+1.0));
        w=-1.0+p*(1.0+p*(-0.333333333333333333333+p*0.152777777777777777777777));
    } else
        w=log(z); /* asymptotic */
    if (z>3.0) w-=log(w); /* useful? */
    for (i=0; i<10; i++) { /* Halley iteration */
        e=exp(w);
        t=w*e-z;
        p=w+1.0;
        t/=e*p-0.5*(p+1.0)*t/p;
        w-=t;
        if (fabs(t)<eps*(1.0+fabs(w))) return w; /* rel-abs error */
    }
    /* should never get here
  fprintf(stderr,"LambertW: No convergence at z=%g, exiting.\n",z);
  exit(1);*/
    return 0;
}
```

```
/* ------------------------------------------------------------------------ */
/*      lambertW1 function, secondary branch                                */
/* ------------------------------------------------------------------------ */

double LambertW1(const double z) {
    int i;
    const double eps=4.0e-16, em1=0.3678794411714423215955237701614608;
    double p=1.0,e,t,w,l1,l2;
    /*
    if (z<-em1 || z>=0.0 || isinf(z) || isnan(z)) {
    fprintf(stderr,"LambertW1: bad argument %g, exiting.\n",z); exit(1);
    }
    */
    /* initial approx for iteration... */
    if (z<-1e-6) { /* series about -1/e */
        p=-sqrt(2.0*(2.7182818284590452353602874713526625*z+1.0));
        w=-1.0+p*(1.0+p*(-0.333333333333333333+p*0.152777777777777777777777));
    } else { /* asymptotic near zero */
        l1=log(-z);
        l2=log(-l1);
        w=l1-l2+l2/l1;
    }
    if (fabs(p)<1e-4) return w;
    for (i=0; i<10; i++) { /* Halley iteration */
        e=exp(w);
        t=w*e-z;
        p=w+1.0;
        t/=e*p-0.5*(p+1.0)*t/p;
        w-=t;
        if (fabs(t)<eps*(1.0+fabs(w))) return w; /* rel-abs error */
    }
    /* should never get here
    fprintf(stderr,"LambertW1: No convergence at z=%g, exiting.\n",z);
    exit(1);*/
    return 0;
}


/* ------------------------------------------------------------------------ */
/*      panelIfromV                                                         */
/* ------------------------------------------------------------------------ */

void panelIfromV(double *p, double *x, int m, int n_x, void *data)
{
    register int i;
    double ui;
    double n, Rs, I0, Iirr, Rsh;
    double I,V;
    double v5[5];

    n = v5[0] = p[0];
    Rs = v5[1] = p[1];
```

```
    if (m==2) {
        if (tipofr==1) Vpanel3Eq(v5); else Vpanel3Eq_2(v5);
        I0 = v5[2];
        Irr = v5[3];
        Rsh = v5[4];
    } else {
        I0 = p[2];
        Irr = p[3];
        Rsh = p[4];
    }

    for(i=0; i<n_x; ++i){
        /* I=I(V, n, Rs) */
        V=dataV[i];
        I = (Rsh*(I0+Irr)-V)/(Rsh+Rs)-((Ns*n*VT)/Rs)*LambertW(\
            ((Rs*Rsh*I0)/(Ns*n*VT*(Rs+Rsh)))*\
            exp(Rsh*(Rs*(I0+Irr)+V)/(Ns*n*VT*(Rs+Rsh)))\
            );
        x[i] = I;
    }

}


/* ------------------------------------------------------------------------ */
/*      ident5pv                                                            */
/* ------------------------------------------------------------------------ */

int ident5pv(double *v5, double Vmpp, double Impp, double *V, double *I, \
                int nsam, double T, int ns, int rf_type)
{
    double opts[LM_OPTS_SZ], info[LM_INFO_SZ];
    double *work, *covar;
    double Rsmax, Rsguess, nguess;
    double p[5]; // n, Rs
    int m,n;
    int i,j,ret;
    double lb[5], ub[5];


    Voc_Ref = V[0];          // first entry is Voc
    Isc_Ref = I[nsam+1];     // last entry is Isc

    Imp_Ref = Impp;
    Vmp_Ref = Vmpp;
    Tref = T;
    Np = 1;
    Ns = ns;
    nsample = nsam;
    tipofr = rf_type;
```

```
/* experimetal values */
dataV = malloc(nsam*sizeof(double));
dataI = malloc(nsam*sizeof(double));
memcpy(dataV, &V[1], nsam*sizeof(double) );
memcpy(dataI, &I[1], nsam*sizeof(double) );

m=5; n=nsam;
opts[0]=LM_INIT_MU; opts[1]=1E-15; opts[2]=1E-15; opts[3]=1E-20;

/*
    relevant only if the Jacobian is approximated using finite differences;
    specifies forward differencing
*/
opts[4]= LM_DIFF_DELTA;

/*
    specifies central differencing to approximate Jacobian;
    more accurate but more expensive to compute!
*/
//opts[4]=-LM_DIFF_DELTA;

/* Rsmax estimation */
nguess=1;
Rsmax=((Np*Vmp_Ref)/(Imp_Ref)+((Np*Ns*nguess*VT)/Imp_Ref)*\
 (1+LambertW1(-exp((Voc_Ref-(nguess*Ns*VT)-(2*Vmp_Ref))/(Ns*nguess*VT))))\
 );
printf("Rsmax=%f\n", Rsmax);
Rsguess = 0.9*Rsmax;
p[0]=nguess; p[1]=Rsguess;

work=malloc((LM_DIF_WORKSZ(m, n)+m*m)*sizeof(double));
if(!work){
    fprintf(stderr, "memory allocation request failed in main()\n");
    exit(1);
}
covar=work+LM_DIF_WORKSZ(m, n);

/* box constraints */
lb[0]=0.5;   ub[0]=5;          // n
lb[1]=0;     ub[1]=100;        // Rs

/*
     Levenberg-Marquardt Library function
     no Jacobian, caller allocates work memory, covariance estimated,
     box constraints
*/
ret=dlevmar_bc_dif(panelIfromV, p, dataI, 2, n, lb, ub, NULL, 2000, opts, \
                    info, work, covar, NULL);

if (p[4]<0) return -1;  // Rs<0: error

printf("Levenberg-Marquardt returned %d in %g iter, reason %g\n", \
```

```
            ret, info[5], info[6]);
    printf("\n");

    // Rs, Iirr, I0
    v5[0] = p[0];
    v5[1] = p[1];
    if (rf_type==1) Vpanel3Eq(v5); else Vpanel3Eq_2(v5);

    return 0;
}
```

¡!—-¿

Bibliography

[3C90, 2008] 3C90 (2008). *3C90 material specification*. Ferroxcube.

[Accarino et al., 2013] Accarino, J., Petrone, G., Ramos-Paja, C., and Spagnuolo, G. (2013). Symbolic algebra for the calculation of the series and parallel resistances in pv module model. In *Clean Electrical Power (ICCEP), 2013 International Conference on*, pages 62–66. IEEE.

[AD623, 2016] AD623 (2016). *Single and Dual-Supply, Rail-to-Rail, Low Cost Instrumentation Amplifier. AD623*. Analog Device.

[AD7714, 1998] AD7714 (1998). *3 V/5 V, CMOS, 500 uA Signal Conditioning ADC. AD7714*. Analog Device.

[AlHajri et al., 2012] AlHajri, M., El-Naggar, K., AlRashidi, M., and Al-Othman, A. (2012). Optimal extraction of solar cell parameters using pattern search. *Renewable Energy*, 44:238–245.

[Askarzadeh and Rezazadeh, 2012] Askarzadeh, A. and Rezazadeh, A. (2012). Parameter identification for solar cell models using harmony search-based algorithms. *Solar Energy*, 86(11):3241–3249.

[Askarzadeh and Rezazadeh, 2013] Askarzadeh, A. and Rezazadeh, A. (2013). Artificial bee swarm optimization algorithm for parameters identification of solar cell models. *Applied Energy*, 102:943–949.

[Bang et al., 2016] Bang, S., Seo, J.-s., Chang, L., Blaauw, D., and Sylvester, D. (2016). A low ripple switched-capacitor voltage regulator using flying capacitance dithering. *IEEE Journal of Solid-State Circuits*, 51(4):919–929.

[Bastidas-Rodriguez et al., 2017] Bastidas-Rodriguez, J. D., Franco, E., Petrone, G., Ramos-Paja, C. A., and Spagnuolo, G. (2017). Quantification of photovoltaic module degradation using model based indicators. *Mathematics and Computers in Simulation*, 131:101–113.

[Batzelis et al., 2015] Batzelis, E. I., Kampitsis, G. E., Papathanassiou, S. A., and Manias, S. N. (2015). Direct mpp calculation in terms of the single-diode pv model parameters. *IEEE Transactions on Energy Conversion*, 30(1):226–236.

[Benghanem, 2009] Benghanem, M. (2009). Low cost management for photovoltaic systems in isolated site with new iv characterization model proposed. *Energy Conversion and Management*, 50(3):748–755.

[Billings and Morey, 2011] Billings, K. H. and Morey, T. (2011). *Switchmode power supply handbook*. McGraw-Hill.

[Brano et al., 2010] Brano, V. L., Orioli, A., Ciulla, G., and Di Gangi, A. (2010). An improved five-parameter model for photovoltaic modules. *Solar Energy Materials and Solar Cells*, 94(8):1358–1370.

[Carrasco et al., 2017] Carrasco, M., Laudani, A., Lozito, G. M., Mancilla-David, F., Riganti Fulginei, F., and Salvini, A. (2017). Low-cost solar irradiance sensing for pv systems. *Energies*, 10(7):998.

[Carrasco et al., 2013] Carrasco, M., Mancilla-David, F., Fulginei, F. R., Laudani, A., and Salvini, A. (2013). A neural networks-based maximum power point tracker with improved dynamics for variable dc-link grid-connected photovoltaic power plants. *International Journal of Applied Electromagnetics and Mechanics*, 43(1, 2):127–135.

[Chen et al., 2006] Chen, W., Yang, X., and Wang, Z. (2006). An active emi filtering technique for improving passive filter low-frequency performance. *IEEE Transactions on Electromagnetic compatibility*, 48(1):172–177.

[Chen et al., 2011] Chen, Y., Wang, X., Li, D., Hong, R., and Shen, H. (2011). Parameters extraction from commercial solar cells i–v characteristics and shunt analysis. *Applied Energy*, 88(6):2239–2244.

[Chine et al., 2016] Chine, W., Mellit, A., Lughi, V., Malek, A., Sulligoi, G., and Pavan, A. M. (2016). A novel fault diagnosis technique for photovoltaic systems based on artificial neural networks. *Renewable Energy*, 90:501–512.

[Chouder et al., 2012] Chouder, A., Silvestre, S., Sadaoui, N., and Rahmani, L. (2012). Modeling and simulation of a grid connected pv system based on the evaluation of main pv module parameters. *Simulation Modelling Practice and Theory*, 20(1):46–58.

[Corless et al., 1996] Corless, R. M., Gonnet, G. H., Hare, D. E., Jeffrey, D. J., and Knuth, D. E. (1996). On the lambertw function. *Advances in Computational mathematics*, 5(1):329–359.

[Cubas et al., 2014] Cubas, J., Pindado, S., and Victoria, M. (2014). On the analytical approach for modeling photovoltaic systems behavior. *Journal of power sources*, 247:467–474.

[De Soto et al., 2006] De Soto, W., Klein, S., and Beckman, W. (2006). Improvement and validation of a model for photovoltaic array performance. *Solar energy*, 80(1):78–88.

[Dobos, 2012] Dobos, A. P. (2012). An improved coefficient calculator for the california energy commission 6 parameter photovoltaic module model. *Journal of solar energy engineering*, 134(2):021011.

[Durán et al., 2012] Durán, E., Andújar, J., Enrique, J., and Pérez-Oria, J. (2012). Determination of pv generator iv/pv characteristic curves using a dc-dc converter controlled by a virtual instrument. *International Journal of Photoenergy*, 2012.

[Duran et al., 2008] Duran, E., Piliougine, M., Sidrach-de Cardona, M., Galán, J., and Andujar, J. (2008). Different methods to obtain the i–v curve of pv modules: A review. In *Photovoltaic Specialists Conference, 2008. PVSC'08. 33rd IEEE*, pages 1–6. IEEE.

[El-Naggar et al., 2012] El-Naggar, K., AlRashidi, M., AlHajri, M., and Al-Othman, A. (2012). Simulated annealing algorithm for photovoltaic parameters identification. *Solar Energy*, 86(1):266–274.

[Erkaya et al., 2014] Erkaya, Y., Flory, I., and Marsillac, S. X. (2014). Development of a string level i–v curve tracer. In *Photovoltaic Specialist Conference (PVSC), 2014 IEEE 40th*, pages 3104–3107. IEEE.

[ETD59-31-22, 2008] ETD59-31-22 (2008). *ETD59/31/22. ETD core and accessories*. Ferroxcube.

[Faba et al., 2017] Faba, A., Gaiotto, S., and Lozito, G. M. (2017). A novel technique for online monitoring of photovoltaic devices degradation. *Solar Energy*, 158:520–527.

[Farivar et al., 2013] Farivar, G., Asaei, B., and Mehrnami, S. (2013). An analytical solution for tracking photovoltaic module mpp. *IEEE Journal of Photovoltaics*, 3(3):1053–1061.

[Fulginei and Salvini, 2010] Fulginei, F. R. and Salvini, A. (2010). The flock of starlings optimization: Influence of topological rules on the collective behavior of swarm intelligence. In *Computational Methods for the Innovative Design of Electrical Devices*, pages 129–145. Springer.

[Fulginei et al., 2012] Fulginei, F. R., Salvini, A., and Pulcini, G. (2012). Metric-topological–evolutionary optimization. *Inverse Problems in Science and Engineering*, 20(1):41–58.

[Gaiotto et al., 2015] Gaiotto, S., Laudani, A., Fulginei, F. R., and Salvini, A. (2015). An advanced measurement equipment for the tracing of photovoltaic panel i–v curves. In *Renewable Energy Research and Applications (ICRERA), 2015 International Conference on*, pages 1010–1014. IEEE.

[Gaiotto et al., 2016] Gaiotto, S., Laudani, A., Fulginei, F. R., Salvini, A., Cardelli, E., and Faba, A. (2016). An equipment for photovoltaic panels characterization based on a fully programmable dc-dc converter. In *Environment and Electrical Engineering (EEEIC), 2016 IEEE 16th International Conference on*, pages 1–6. IEEE.

[Gaiotto et al.,] Gaiotto, S., Riganti Fulginei, F., Lozito, G. M., and Salvini, A. A low-ripple switched-capacitor voltage regulator with decoupling capabilities.

International Journal of Numerical Modelling: Electronic Networks, Devices and Fields.

[Hamza and Qiu, 2013] Hamza, D. and Qiu, M. (2013). Digital active emi control technique for switch mode power converters. *IEEE Transactions on Electromagnetic Compatibility*, 55(1):81–88.

[Ibirriaga et al., 2010] Ibirriaga, J. J. M., de Mendiluce Pena, X. M., Opritescu, A., Sera, D., and Teodorescu, R. (2010). Low-cost, high flexibility i–v curve tracer for photovoltaic modules. In *Optimization of Electrical and Electronic Equipment (OPTIM), 2010 12th International Conference on*, pages 1210–1215. IEEE.

[Ishaque et al., 2012] Ishaque, K., Salam, Z., Mekhilef, S., and Shamsudin, A. (2012). Parameter extraction of solar photovoltaic modules using penalty-based differential evolution. *Applied Energy*, 99:297–308.

[Ji et al., 2016] Ji, J., Chen, W., and Yang, X. (2016). Arc welding inverter with embedded digital active emi controller. In *Applied Power Electronics Conference and Exposition (APEC), 2016 IEEE*, pages 493–498. IEEE.

[Jiang et al., 2013] Jiang, L. L., Maskell, D. L., and Patra, J. C. (2013). Parameter estimation of solar cells and modules using an improved adaptive differential evolution algorithm. *Applied Energy*, 112:185–193.

[Johnson et al., 1993] Johnson, H. W., Graham, M., et al. (1993). *High-speed digital design: a handbook of black magic*, volume 1. Prentice Hall Upper Saddle River, NJ.

[Kanzow et al., 2002] Kanzow, C., Fukushima, M., and Yamashita, N. (2002). *Levenberg-Marquardt methods for constrained nonlinear equations with strong local convergence properties*. Inst. of Applied Math. and Statistics.

[Kaplanis and Kaplani, 2011] Kaplanis, S. and Kaplani, E. (2011). Energy performance and degradation over 20years performance of bp c-si pv modules. *Simulation Modelling Practice and Theory*, 19(4):1201–1211.

[Kato et al., 1998] Kato, K., Murata, A., and Sakuta, K. (1998). Energy pay-back time and life-cycle co2 emission of residential pv power system with silicon pv module. *Progress in Photovoltaics: Research and Applications*, 6(2):105–115.

[Kazmi et al., 2009] Kazmi, S., Goto, H., Ichinokura, O., and Guo, H. J. (2009). An improved and very efficient mppt controller for pv systems subjected to rapidly varying atmospheric conditions and partial shading. In *Power Engineering Conference, 2009. AUPEC 2009. Australasian Universities*, pages 1–6. IEEE.

[Kuai and Yuvarajan, 2006] Kuai, Y. and Yuvarajan, S. (2006). An electronic load for testing photovoltaic panels. *Journal of Power Sources*, 154(1):308–313.

[Kundert, 2004] Kundert, K. (ver 4, Jan 2004). *Power Supply Noise Reduction*. Designer's Guide Consulting, Inc., Available: http://www.designers-guide.org/.

[Laudani et al., 2013a] Laudani, A., Fulginei, F. R., and Salvini, A. (2013a). Closed forms for the fully-connected continuous flock of starlings optimization algorithm. In *Computer Modelling and Simulation (UKSim), 2013 UKSim 15th International Conference on*, pages 45–50. IEEE.

[Laudani et al., 2014a] Laudani, A., Fulginei, F. R., and Salvini, A. (2014a). High performing extraction procedure for the one-diode model of a photovoltaic panel

from experimental i–v curves by using reduced forms. *Solar Energy*, 103:316–326.

[Laudani et al., 2014b] Laudani, A., Fulginei, F. R., and Salvini, A. (2014b). High performing extraction procedure for the one-diode model of a photovoltaic panel from experimental I–V curves by using reduced forms. *Solar Energy*, 103:316 – 326.

[Laudani et al., 2014c] Laudani, A., Fulginei, F. R., and Salvini, A. (2014c). Identification of the one-diode model for photovoltaic modules from datasheet values. *Solar Energy*, 108:432–446.

[Laudani et al., 2014d] Laudani, A., Fulginei, F. R., Salvini, A., Lozito, G. M., and Mancilla-David, F. (2014d). Implementation of a neural mppt algorithm on a low-cost 8-bit microcontroller. In *Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), 2014 International Symposium on*, pages 977–981. IEEE.

[Laudani et al., 2014e] Laudani, A., Lozito, G. M., Fulginei, F. R., and Salvini, A. (2014e). An efficient architecture for floating point based miso neural neworks on fpga. In *Computer Modelling and Simulation (UKSim), 2014 UKSim-AMSS 16th International Conference on*, pages 12–17. IEEE.

[Laudani et al., 2015] Laudani, A., Lozito, G. M., Fulginei, F. R., and Salvini, A. (2015). On training efficiency and computational costs of a feed forward neural network: a review. *Computational intelligence and neuroscience*, 2015:83.

[Laudani et al., 2013b] Laudani, A., Mancilla-David, F., Riganti-Fulginei, F., and Salvini, A. (2013b). Reduced-form of the photovoltaic five-parameter model for efficient computation of parameters. *Solar Energy*, 97:122–127.

[Laudani et al., 2014f] Laudani, A., Riganti Fulginei, F., Salvini, A., Lozito, G. M., and Coco, S. (2014f). Very fast and accurate procedure for the characterization of photovoltaic panels from datasheet information. *International Journal of Photoenergy*, 2014.

[Laudani et al., 2013c] Laudani, A., Riganti Fulginei, F., Salvini, A., Schmid, M., and Conforto, S. (2013c). Cfso3: A new supervised swarm-based optimization algorithm. *Mathematical Problems in Engineering*, 2013.

[Lei et al., 2011] Lei, P., Li, Y., and Seem, J. E. (2011). Sequential esc-based global mppt control for photovoltaic array with variable shading. *Sustainable Energy, IEEE Transactions on*, 2(3):348–358.

[Leite et al., 2012] Leite, V., Batista, J., Chenlo, F., and Afonso, J. (2012). Low-cost instrument for tracing current-voltage characteristics of photovoltaic modules. In *Proc. of the International Conference on Renewable Energies and Power Quality (ICREPQ'12)*, pages 28–30.

[Leite and Chenlo, 2010] Leite, V. and Chenlo, F. (2010). An improved electronic circuit for tracing the iv characteristics of photovoltaic modules and strings. In *Proceedings of the International Conference on Renewable Energies and Power Quality (ICREPQ'10)*. European Association for the Development of Renewable Energies, Environment and Power Quality.

[Li, 2014] Li, S. (2014). A mppt control strategy with variable weather parameter and no dc/dc converter for photovoltaic systems. *Solar Energy*, 108:117–125.

[Li et al., 2012] Li, Z., Wang, Y., Zhou, D., and Wu, C. (2012). An intelligent method for fault diagnosis in photovoltaic array. *System Simulation and Scientific Computing*, pages 10–16.

[Lima et al., 2013] Lima, N. N., de Freitas, L. C., Buiatti, G. M., Vieira, J. B., Freitas, L. C., and Coelho, E. A. (2013). Low complexity system for real-time determination of current-voltage characteristic of pv modules and strings. In *Applied Power Electronics Conference and Exposition (APEC), 2013 Twenty-Eighth Annual IEEE*, pages 2817–2823. IEEE.

[Lorente et al., 2014] Lorente, D. G., Pedrazzi, S., Zini, G., Dalla Rosa, A., and Tartarini, P. (2014). Mismatch losses in pv power plants. *Solar Energy*, 100:42–49.

[Lourakis, 2004] Lourakis, M. (Jul. 2004). levmar: Levenberg-marquardt nonlinear least squares algorithms in C/C++. [web page] `http://www.ics.forth.gr/~lourakis/levmar/`. [Accessed on 31 Jan. 2005.].

[Lozito et al., 2014] Lozito, G. M., Bozzoli, L., and Salvini, A. (2014). Micro-controller based maximum power point tracking through fcc and mlp neural networks. In *Education and Research Conference (EDERC), 2014 6th European Embedded Design in*, pages 207–211. IEEE.

[Luo et al., 2008] Luo, Y.-K., Chen, K.-H., and Hsu, W.-C. (2008). A dual-phase charge pump regulator with nano-ampere switched-capacitor cmos voltage reference for achieving low output ripples. In *Electronics, Circuits and Systems, 2008. ICECS 2008. 15th IEEE International Conference on*, pages 446–449. IEEE.

[Madsen et al., 2004] Madsen, K., Nielsen, H. B., and Tingleff, O. (2004). Methods for non-linear least squares problems.

[Mahamad and Saon, 2014] Mahamad, A. K. and Saon, S. (2014). Development of artificial neural network based mppt for photovoltaic system during shading condition. *Applied Mechanics and Materials*, 448:1573–1578.

[Mahrane et al., 2010] Mahrane, A., Guenounou, A., Smara, Z., Chikh, M., and Lakehal, M. (2010). Test bench for photovoltaic modules. In *International Symposium on Environment Friendly Energies in Electrical Applications*.

[Mancilla-David et al., 2014] Mancilla-David, F., Riganti-Fulginei, F., Laudani, A., and Salvini, A. (2014). A neural network-based low-cost solar irradiance sensor. *IEEE Transactions on Instrumentation and Measurement*, 63(3):583–591.

[Masters, 2013] Masters, G. M. (2013). *Renewable and efficient electric power systems*. John Wiley & Sons.

[MT-101, 2009] MT-101 (March 2009). *MT-101 Tutorial: Decoupling Techniques*. Analog Devices, Inc., Norwood, MA. [Online]. Available: http://www.analog.com/.

[Munoz et al., 2011] Munoz, M., Alonso-Garcia, M., Vela, N., and Chenlo, F. (2011). Early degradation of silicon pv modules and guaranty conditions. *Solar energy*, 85(9):2264–2274.

[Oliveri et al., 2017] Oliveri, A., Cassottana, L., Laudani, A., Fulginei, F. R., Lozito, G. M., Salvini, A., and Storace, M. (2017). Two fpga-oriented high-speed irradiance virtual sensors for photovoltaic plants. *IEEE Transactions on Industrial Informatics*, 13(1):157–165.

[Omata and Shimizu, 2016] Omata, S. and Shimizu, T. (2016). Design method for emi filters connected to both dc and ac ports of a residential-use photovoltaic power conditioner. *Electrical Engineering in Japan*, 197(2):56–67.

[Orioli and Di Gangi, 2013] Orioli, A. and Di Gangi, A. (2013). A procedure to calculate the five-parameter model of crystalline silicon photovoltaic modules on the basis of the tabular performance data. *Applied energy*, 102:1160–1177.

[Osterwald, 2003] Osterwald, C. R. (2003). Standards, calibration and testing of pv modules and solar cells. *Practical Handbook of Photovoltaics: Fundamentals and Applications*, pages 791–816.

[Ott and Ott, 1988] Ott, H. W. and Ott, H. W. (1988). *Noise reduction techniques in electronic systems*, volume 442. Wiley New York.

[Papageorgas et al., 2015] Papageorgas, P., Piromalis, D., Valavanis, T., Kambasis, S., Iliopoulou, T., and Vokas, G. (2015). A low-cost and fast pv iv curve tracer based on an open source platform with m2m communication capabilities for preventive monitoring. *Energy Procedia*, 74:423–438.

[Peled and Appelbaum, 2016] Peled, A. and Appelbaum, J. (2016). Minimizing the current mismatch resulting from different locations of solar cells within a pv module by proposing new interconnections. *Solar Energy*, 135:840–847.

[Peng et al., 2013] Peng, L., Sun, Y., Meng, Z., Wang, Y., and Xu, Y. (2013). A new method for determining the characteristics of solar cells. *Journal of power sources*, 227:131–136.

[Phinikarides et al., 2014] Phinikarides, A., Kindyni, N., Makrides, G., and Georghiou, G. E. (2014). Review of photovoltaic degradation rate methodologies. *Renewable and Sustainable Energy Reviews*, 40:143–152.

[Picault et al., 2010] Picault, D., Raison, B., Bacha, S., De La Casa, J., and Aguilera, J. (2010). Forecasting photovoltaic array power production subject to mismatch losses. *Solar Energy*, 84(7):1301–1309.

[Radjai et al., 2014] Radjai, T., Rahmani, L., Mekhilef, S., and Gaubert, J. P. (2014). Implementation of a modified incremental conductance mppt algorithm with direct control based on a fuzzy duty cycle change estimator using dspace. *Solar Energy*, 110:325–337.

[Rajasekar et al., 2013] Rajasekar, N., Kumar, N. K., and Venugopalan, R. (2013). Bacterial foraging algorithm based solar pv parameter estimation. *Solar Energy*, 97:255–265.

[Ramaprabha et al., 2014] Ramaprabha, R., Jubair, S., Suhas, K., and Lokesh, A. (2014). Development of improved curve tracer for photovoltaic system under partial shaded conditions. In *Circuit, Power and Computing Technologies (ICCPCT), 2014 International Conference on*, pages 585–589. IEEE.

[Rivai and Rahim, 2014] Rivai, A. and Rahim, N. A. (2014). Binary-based tracer of photovoltaic array characteristics. *IET Renewable Power Generation*, 8(6):621–628.

[SLLA284A, 2014] SLLA284A (January 2009–Revised November 2014). *SLLA284A: Isolator Design Guide*. Texas Instruments, Inc., Dallas, TX. [Online]. Available: http://www.ti.com/.

[SLLS868S, 2016] SLLS868S (SEPTEMBER 2007–REVISED APRIL 2016). *ISO724x High-Speed, Quad-Channel Digital Isolators*. Texas Instruments.

[Spertino et al., 2015] Spertino, F., Ahmad, J., Ciocia, A., Di Leo, P., Murtaza, A. F., and Chiaberge, M. (2015). Capacitor charging method for i–v curve tracer and mppt in photovoltaic systems. *Solar Energy*, 119:461–473.

[SPRA906, 2003] SPRA906 (April 2003). *SPRA906: Decoupling/Filtering Techniques*. Texas Instruments, Inc., Dallas, TX. [Online]. Available: http://www.ti.com/.

[STP235-20/Wd, 2011] STP235-20/Wd (2011). *STP235-20/Wd - 235W Polycrystalline Solar Module*. Suntech.

[SX310, 2007] SX310 (2007). *SX310. 10 Watt Photovoltaic Module*. BP Solar.

[Tey and Mekhilef, 2014] Tey, K. S. and Mekhilef, S. (2014). Modified incremental conductance mppt algorithm to mitigate inaccurate responses under fast-changing solar irradiation level. *Solar Energy*, 101:333–342.

[Tian et al., 2012] Tian, H., Mancilla-David, F., Ellis, K., Muljadi, E., and Jenkins, P. (2012). A cell-to-module-to-array detailed model for photovoltaic panels. *Solar energy*, 86(9):2695–2706.

[Vincheh et al., 2014] Vincheh, M. R., Kargar, A., and Markadeh, G. A. (2014). A hybrid control method for maximum power point tracking (mppt) in photovoltaic systems. *Arabian Journal for Science and Engineering*, 39(6):4715–4725.

[Wang et al., 2005a] Wang, S., Chen, R., Lee, F. C., and Van Wyk, J. (2005a). Improved passive filter configurations for high-frequency conducted emi in power electronics. In *Power Electronics and Applications, 2005 European Conference on*, pages 16–pp. IEEE.

[Wang et al., 2005b] Wang, S., Chen, R., Van Wyk, J. D., Lee, F. C., and Odendaal, W. G. (2005b). Developing parasitic cancellation technologies to improve emi filter performance for switching mode power supplies. *IEEE Transactions on Electromagnetic Compatibility*, 47(4):921–929.

[Wright and Nocedal, 1999] Wright, S. J. and Nocedal, J. (1999). Numerical optimization. *Springer Science*, 35(67-68):7.

[Yang et al., 2014] Yang, Y., Chang, X., Chen, W., and Yang, X. (2014). Design of a differential mode active emi filter based on conducted emission analysis used in dc/dc converter. In *General Assembly and Scientific Symposium (URSI GASS), 2014 XXXIth URSI*, pages 1–4. IEEE.

[Zhenyang et al., 2015] Zhenyang, Y., Shishan, W., Zheng, S., and Lee, B.-L. (2015). The reviews of integrated emi filters applied in power electronic system. In *Electromagnetic Compatibility (APEMC), 2015 Asia-Pacific Symposium on*, pages 227–230. IEEE.

List of Figures