Università di Pisa

Scuola di Dottorato in Ingegneria "Leonardo da Vinci"

Corso di Dottorato di Ricerca in
Ingegneria dell'Informazione

Tesi di Dottorato di Ricerca

Designing Innovative Electronic Systems to Face the Challenges of Feasibility and Performance in Sensing and Control Applications

Andrea Lazzeri

Università di Pisa

Scuola di Dottorato in Ingegneria "Leonardo da Vinci"

Corso di Dottorato di Ricerca in
Ingegneria dell'Informazione

Tesi di Dottorato di Ricerca

Designing Innovative Electronic Systems to Face the Challenges of Feasibility and Performance in Sensing and Control Applications

Autore:

Andrea Lazzeri   _____

Relatori:

Prof. R. Saletti   _____

Prof. R. Roncella   _____

Anno 2011
SSD ING-INF/01

Ai miei genitori,
e a Marta

Abstract

This thesis reports the results obtained during a three year research program on electronic systems design. Two advanced applications in the field of sensing and control electronics are analysed and discussed, with a particular emphasis on the design solutions which allowed to improve the performance provided by the state–of–the–art and, at the same time, to implement new functionalities previously considered unfeasible.

The first application is focused on the design, implementation and test of a long–range optical fibre Distributed Temperature Sensor (DTS). A new architecture, aiming to address the main design issues related to the sensing range extension and to the measurement time reduction, is presented. A low–noise APD–based optoelectronic front–end allows to optimize the input signal–to–noise ratio (SNR) of the system. A high level of configurability is also obtained by allowing the user to control the gain and offset of the amplification chain, as well as to fine tune the APD bias voltage and operating temperature. This way, the optimum multiplication factor of the APD, which ensures the best available input SNR, can easily be selected. Signals are acquired through a flexible ADC/FPGA–based platform. Here, a set of decimation and interleaved sampling algorithms allows to efficiently exploit the available memory resources and reach long measurement ranges. Finally, a patented SNR enhancing technique based on laser pulse coding allows to efficiently apply, for the first time, Simplex codes to long–range DTS systems, and therefore to significantly reduce the measurement time.

A prototype, which is able to cover distances up to $87.4\,\mathrm{km}$ with a spatial resolution of just $1.3\,\mathrm{m}$, has been implemented. Preliminary tests on a $20\,\mathrm{km}$

range, carried out without coding, show a performance comparable with the state–of–the–art. The benefits of the coding technique have been instead evaluated on a 10 km range. It has been proved that an outstanding measurement time reduction up to 95 % with respect to conventional long–range uncoded systems is achievable.

The second application is instead focused on the development of automotive embedded systems for Formula SAE vehicles. For the first time, linear Voice Coil Actuators (VCAs) have been used to implement a robotized shift–by–wire system, which has been applied to the gear and the clutch devices of the first student–build race car developed at the University of Pisa. A numerical model of the electromechanical system allows to properly size the actuators, so as to obtain a shifting performance better than any other known solution adopted for Formula SAE vehicles. A DSP–based Gear Control Unit (GCU), has been specifically designed to implement the upshift, downshift and car start procedures, and to direct drive the actuators. On–track tests show that the achieved upshift time is just 40 ms, i.e. less than half of the one provided by counterparts.

Moreover, a data logging system and a telemetry system have been also implemented. Both are based on a 16 MHz, 8 bit microcontroller and communicate with other on–board units through the CAN bus. The first is able to record on a removable SD card the information related to the speed of wheels, the suspensions stroke and the steering angle, as well as the data coming from a 3–axial accelerometer, a gyroscope and a GPS. Using a 2 GB card, signals can be logged for 97 h with a 40 Hz sampling frequency, which is a very good result if compared with commercially available products. Acquired data are also sent over the CAN bus and made available to other units. On the other hand, the telemetry system is composed by two twin units, one on–board and one connected to a PC via USB. The on–board unit listens to CAN activity and forwards messages to the PC unit over a 2.4 GHz wireless encrypted link. A custom developed LabVIEW application allows for the real–time monitoring of the vehicle status and for a rapid fault detection. The radio link is bidirectional, so that the PC is also able to send CAN messages back to on–board units, such as the GCU, and configure their parameters remotely. These capabilities, along with a maximum outdoor range of 2 km, make the system very interesting with respect to other Formula SAE products available on the market.

Sommario

In questa tesi sono riassunti i risultati conseguiti dall'attività di ricerca svolta durante l'intero triennio di dottorato nell'ambito della progettazione di sistemi elettronici. In particolare, sono discusse e analizzate due applicazioni d'avanguardia nel campo dell'elettronica di sensing e controllo, mettendo maggiormente in risalto le scelte progettuali che hanno consentito di migliorare le prestazioni dello stato dell'arte e, al tempo stesso, di implementare nuove funzionalità finora considerate infattibili.

La prima applicazione tratta la progettazione, l'implementazione e il collaudo di un sensore distribuito di temperatura in fibra ottica per misure long–range. In particolare, è stata sviluppata una nuova architettura che consente di affrontare le principali problematiche relative all'estensione del range di funzionamento e alla riduzione del tempo di misura. Tale architettura comprende un front–end optoelettronico a basso rumore, basato su APD, che consente di ottimizzare il rapporto segnale–rumore (SNR) di ingresso del sistema. Grazie ad un'elevata configurabilità, l'utente è in grado non solo di controllare il guadagno e l'offset della catena di amplificazione, ma anche di regolare la tensione di polarizzazione e la temperatura di funzionamento dell'APD. Così facendo è possibile portare con facilità l'APD a lavorare in condizioni ottime, per le quali l'SNR di ingresso è massimo. I segnali di ingresso sono poi acquisiti per mezzo di una piattaforma basata su ADC/FPGA. All'interno dell'FPGA sono implementati alcuni algoritmi di decimazione e campionamento interlacciato che consentono di sfruttare in modo efficiente le risorse di memoria disponibili e di raggiungere elevate distanze di misura. Infine, è descritta una tecnica brevettata che consente di migliorare ulteriormente l'SNR del sistema grazie alla codifica degli impulsi laser. In particolare, tale tecnica permette di utilizzare

per la prima volta i codici Simplex su sensori long–range, e quindi di ridurre significativamente il tempo di misura.

È stato realizzato un prototipo in grado di coprire distanze fino a 87.4 km con una risoluzione spaziale di soli 1.3 m. I test preliminari, svolti su un range di 20 km e senza l'utilizzo della tecnica di codifica, mostrano prestazioni confrontabili con lo stato dell'arte. Viceversa, i miglioramenti ottenibili con la tecnica di codifica sono stati valutati su una distanza di 10 km. In particolare, si è dimostrato che è possibile ottenere un'eccezionale riduzione del tempo di misura pari a circa il 95 %.

La seconda applicazione riguarda invece lo sviluppo di sistemi automotive embedded per veicoli di Formula SAE. In tale ambito, è stato progettato un sistema robotizzato di shift–by–wire utilizzando, per la prima volta, attuatori lineari di tipo Voice Coil. Tale sistema è stato poi applicato al cambio e alla frizione della prima auto da corsa costruita dagli studenti dell'Università di Pisa. Inizialmente, si descrive un modello numerico che consente di simulare il comportamento dell'intero sistema elettromeccanico, e in particolare di dimensionare opportunamente gli attuatori per ottenere tempi di cambiata migliori rispetto a quelli comunemente raggiunti su veicoli di Formula SAE. Successivamente si riportano i dettagli relativi alla progettazione di una centralina elettronica di controllo (Gear Control Unit, GCU) basata su DSP, che pilota direttamente gli attuatori e gestisce sia le procedure di cambiata, sia quelle di partenza da fermo. Il collaudo su strada mostra un tempo di cambiata di soli 40 ms, pari a circa la metà di quanto si riesce a ottenere con le altre soluzioni utilizzate in Formula SAE.

Sono stati inoltre sviluppati un sistema di data–logging e uno di telemetria, entrambi basati su un semplice microcontrollore a 8 bit, 16 MHz, e in grado di comunicare con le altre unità di bordo tramite bus CAN. Il primo di essi memorizza su scheda SD le informazioni relative alla velocità delle ruote, all'escursione delle sospensioni e all'angolo di sterzo, insieme ai dati provenienti da un accelerometro triassiale, un giroscopio e un GPS. In particolare, utilizzando una scheda da 2 GB, il collaudo finale dimostra che è possibile memorizzare tutte le informazioni per ben 97 h con una frequenza di campionamento di 40 Hz. Tali prestazioni sono estremamente significative se confrontate con quelle dei prodotti attualmente disponibili in commercio. I dati acquisiti sono inoltre inviati sul bus CAN e resi disponibili alle altre unità. Il sistema di telemetria è invece composto da due unità gemelle, una montata a bordo del

veicolo, l'altra connessa a un PC tramite link USB. L'unità di bordo ascolta l'attività del bus CAN e inoltra i messaggi ricevuti verso il PC tramite un link wireless crittografato a 2.4 GHz. Un'applicazione LabVIEW consente il monitoraggio in tempo reale dello stato del veicolo e una rapida individuazione di eventuali guasti. Inoltre, il collegamento radio è bidirezionale, e dunque il PC è in grado di inviare messaggi CAN alle unità di bordo e configurare i loro parametri di funzionamento da remoto. Tutte queste caratteristiche, insieme ad una portata in campo aperto di circa 2 km, rendono il sistema estremamente interessante rispetto ad altri prodotti commerciali dedicati alla Formula SAE.

Ringraziamenti

Desidero ringraziare sentitamente il prof. Roberto Saletti e il prof. Roberto Roncella, perché mi hanno dato l'opportunità di fare questa bellissima esperienza, incoraggiandomi nelle mie ricerche e sostenendomi nell'incertezza. Da loro ho imparato moltissimo, sia sul piano professionale che umano.

Grazie a Roberto Barsacchi per la disponibilità, la competenza e il prezioso contributo che ha dato al progetto DTS. Grazie anche agli altri tesisti che hanno collaborato allo stesso progetto, Antonio Vinci e Serena Cipriani, per l'ottimo lavoro svolto.

Un grazie a tutto il gruppo di lavoro della Scuola Superiore Sant'Anna con cui ho avuto l'onore di collaborare, a partire dal prof. Fabrizio Di Pasquale e da Gabriele Bolognini, per la loro estrema disponibilità e competenza, e per avermi dato la possibilità di ultimare la sperimentazione sui codici. Grazie a Alessandro Signorini, Marcelo Soto e Tiziano Nannipieri, per la grande pazienza dimostrata nello spiegarmi come "funziona" un DTS.

Desidero inoltre ringraziare il prof. Massimo Guiggiani, per aver creduto nella "folle" idea di creare una squadra corse all'interno dell'Università di Pisa, per la possibilità offertami di farne parte, e per aver insegnato a tutti i suoi membri a credere nelle rispettive capacità, e a non arrendersi di fronte ai continui ostacoli. Non dimenticherò mai l'entusiasmo con cui ha accompagnato ogni "vittoria" dell'E–Team.

Grazie a Gabriele Fantechi per la sua tenacia e determinazione, e perché mi ha fatto capire che "prima di dire che una cosa è infattibile, bisogna provare a costruirla". Grazie a Francesco Lenzi, perché mentre si rivelava un eccezionale compagno di lavoro, mi insegnava che il lavoro non è tutto nella vita.

Ringrazio i ragazzi che hanno svolto la loro tesi all'interno dell'E–Team, Alessandro Giovannini, Andrea Vitagliano, Antonino Riciputo e Giovanni Puccioni, perché hanno svolto ottimi lavori, sacrificando molto di loro stessi per la squadra. Grazie a tutto l'E–Team, e in particolare ai capitani Francesco Castellana, Lapo Mori e Alessio Simi, perché con tutti loro ho vissuto un'esperienza irripetibile e indimenticabile.

Grazie ai due Emanuele, Leonardi e Marraccini: anche se abbiamo trascorso poco tempo insieme, ho avuto modo di sperimentare la loro eccezionalità e di apprezzare la loro intelligenza.

Un grazie a Sergio Saponara, e ai "ragazzi della stanza di là", Esa Petri, Tony Bacchillone, Massimo Rovini, ai quali abbiamo dato spesso fastidio con i nostri "rumori molesti", ma con i quali abbiamo piacevolmente condiviso ogni qualsivoglia forma di banchetto, improvvisato o non. Un grazie particolare al prof. Luca Fanucci, per la sua autentica generosità e per avermi capito prima che io riuscissi a spiegarmi.

Last but not least, un grazie enorme a Federico Baronti, amico e maestro, insostituibile compagno di avventura e di sventura. Fin dall'inizio, i nostri pensieri si sono sempre agganciati sulla stessa lunghezza d'onda, e nelle rare volte in cui ciò non è successo. . . ci siamo sfidati a suon di tappini.

Voglio ringraziare anche tutti i miei amici: hanno sopportato le mie stanchezze e le serate a tema unico (il mio lavoro!), aspettando da circa un anno che io sia abbastanza riposato per venire a cena nella mia nuova casa.

Un grazie speciale al mio amico Mario, per la sua disponibilità illimitata nello spazio e nel tempo. Se oggi sono in grado di "camminare" da solo, lo devo alla forza del suo insegnamento.

Grazie a mio padre e a mia madre: in questi tre anni mi hanno aiutato a prendere decisioni fondamentali per la mia vita, anche quando non la pensavano come me. Nonostante in questo periodo abbiano dovuto affrontare grandi prove, non hanno mai smesso di pensarmi, nemmeno nei momenti più difficili.

E infine ringrazio Marta perché, capendo le mie difficoltà, mi ha amato di più proprio quando lo meritavo di meno.

Contents

Introduction

Nowadays, electronic system designers have at their disposal a large amount of high–performance devices with which they can develop innovative applications. Indeed, thanks to the progress of technology and to the advances in semiconductor research, a wide range of analog, mixed–signal and digital state–of–the–art devices featuring high processing speeds, low noise figures, reduced power consumptions and small sizes, is already available on the market at low cost.

In addition, the performance gap between Application Specific Integrated Circuits (ASICs) and general purpose programmable devices, such as Digital Signal Processors (DSPs) and microcontrollers (μCs), or configurable hardware resources, such as Field Programmable Gate Arrays (FPGAs), has been dramatically reduced. This means that high–performance and flexibility have become so close that innovative applications, previously considered unfeasible, are now really implementable.

As a consequence, system–level research focuses on the development of novel architectures to face feasibility challenges and, at the same time, to make the most of advanced devices. To this end, a transversal design approach, which addresses the issues related to hardware, firmware and software layers, is the key to increasing the value of the final product, as well as to contributing to the advancement of research.

In this context, the present work describes the main results obtained from the system–level research and design activities carried out during the three year PhD program. The work is divided into two different parts.

In the first part, the design, implementation and test of an optical fibre Distributed Temperature Sensor (DTS) is reported. Initially, in Chapter 1 the physical background, the working principles, and the main issues of DTS systems are analysed, the performance parameters of state–of–the–art sensors available on the market are reported, and the design specifications of the new prototype are determined. Chapter 2 describes the developed architecture and shows how the high–level specifications have been mapped to the different system blocks. Besides the required optical components, the architecture comprises an analog optoelectronic receiver based on Avalanche Photo Diodes (APDs), a digital conversion and acquisition stage based on a state–of–the–art ADC/FPGA platform, and a PC which implements the final processing algorithms and the user interface. In Chapter 3 the hardware implementation of the system is described, with a particular emphasis on the optoelectronic receiver, on the FPGA architecture, and on the design choices which allowed to optimize their performance and increase the system flexibility. On the other hand, Chapter 4 describes the firmware and software implementation of the system, showing how the adopted solutions, which have been obtained following a layered platform approach, can also be effectively used to develop other PC–based instruments with a minimum redesign effort. Chapter 5 reports the obtained experimental results. At first, the achieved performance is compared to the design specifications, both at block–level and at system–level. Then, the overall performance is compared to the state–of–the–art. Finally, in Chapter 6 a novel temperature measurement technique for DTS systems allowing to dramatically reduce the measurement time to values well lower than those provided by state–of–the–art commercial sensors is presented. This technique, which relies on cyclic Simplex codes, has been theoretically demonstrated and also validated by experimental results.

The second part deals with automotive embedded systems, and reports the research and design activities carried out in the last three years by the electronic engineering division of the Formula SAE student team of the University of Pisa. Chapter 7 introduces the international Formula SAE competitions and briefly resumes the history of the team. Chapter 8 presents the design and implementation of a robotized shift–by–wire system based on Voice Coil Actuators (VCAs), which has been applied to the first year race car, the ET1. After determining the system specifications, a numerical model, which allows to analyse both the dynamic and the thermal behaviour of VCAs, is presented. The model has been used to properly select the actuators required for the specific application, and also to determine their control strategy. Then,

the design and implementation of a DSP–based Gear Control Unit (GCU) is reported, with a particular emphasis on the power devices used to directly drive the VCAs and on the firmware control algorithms. Finally, the results obtained during test and official competitions are reported and compared with those provided by other solutions commonly adopted for Formula SAE vehicles. In Chapter 9, the shift–by–wire system developed for the second year car, the ET2ev, is described. The system is based on pneumatic actuators, which are controlled by means of solenoid valves. The design and implementation of a new µC–based GCU featuring an automatic upshift algorithm is presented, and the obtained results are compared with those of the previous year. Finally, Chapter 10 reports the new data processing units developed for the third year car, the ET3. A data logging system and a telemetry system, both based on microcontrollers, have been designed and implemented. Each unit is equipped with a configurable CAN interface, which allows to connect them to the GCU and the engine management system in a very efficient way and with a significant reduction of the wiring harness complexity and weight. The achieved flexibility and performance are described in detail and represent a very good result if compared with those provided by state–of–the–art counterparts available on the market.

Part I

Design, Implementation and Test of a Long–Range Optical Fibre Distributed Temperature Sensor based on Raman Scattering

1 Optical Fibre Distributed Temperature Sensors

1.1 Overview

An optical fibre Distributed Temperature Sensor (DTS) is an optoelectronic system which is able to detect the continuous temperature profile along the whole length of an optical fibre. First DTS prototypes were built about 30 years ago, but their real diffusion as measuring instruments is rather recent. Indeed, the large technological progress occurred in the last few years in the fields of optical and electronic devices has brought the performance of DTS systems to such a high level that they are used – or even required – in many demanding applications, both scientific and industrial [1].

The key aspect of these kind of sensors is that the optical fibre is the actual temperature probe. The fibre can be properly laid in contact with the object to be monitored, so that the thermal profile along the object length, surface or volume can be determined without the need for complex arrays of discrete sensors. Moreover, the fibre is immune to EM noise and made of dielectric material, so that it can be safely used in electrically noisy environments or in those applications in which spark hazard must be avoided. As an example, typical DTS systems available on the market are able to detect the temperature profile over a 20–30 km fibre with a temperature resolution of just 1 K and a spatial resolution down to 1 m. Thanks to these parameters, they are used for instance to monitor gas or oil pipelines [2], high–voltage underground [3,4] or submarine [5] power cables, nuclear [6] or chemical [7] plant processes, civil infrastructures such as railway or road tunnels, bridges or dikes [8], and also for aeronautic applications [9].

The operating principle of DTS systems is based on the Optical Time–Domain Reflectometry (OTDR) [10]: a laser pulse is launched into the probing fibre and the temperature–dependent backscattered lights are acquired as a function of time. These lights can be due to different physical phenomena, namely the *Raman scattering* and the *Brillouin scattering* [11]. The first produces two backscattered lights, which have symmetrical frequencies around the launched pulse one, and temperature–dependent intensities. The latter also produces two symmetrical frequency lights, however in this case the fibre temperature does not affect their intensities, but rather their frequency shift from the pulse light. Once Raman or Brillouin signals have been acquired, as their propagation speeds in the fibre are known, the evolution of signals along the fibre axis can be reconstructed, and hence the temperature profile can be extracted.

Raman and Brillouin approaches show different pros and cons. From a theoretical point of view, Brillouin signals have an intensity about a order of magnitude higher than Raman ones, and hence they feature a better Signal–to–Noise Ratio (SNR) which allows for a longer measuring range and for an improved spatial and temperature resolution [12]. On the other hand, Brillouin–based DTS systems show some implementation difficulties. Indeed, Brillouin signals are closer in frequency to the pulse light than Raman ones (about $\pm 13\,\mathrm{GHz}$ separation versus $\pm 13\,\mathrm{THz}$ in silica fibres) [11], and hence narrow bandwidth optical filters are required. Moreover, as the fibre temperature information is translated into a frequency shift, extremely linear and low–loss frequency–to–voltage converters should be used. Finally, it is important to point out that the Brillouin frequency shift not only depends on temperature but also on the fibre longitudinal strain (so that distributed strain sensors are based on it [13]), and hence advanced processing and calibration algorithms are required as well. For these reasons, Raman–based systems are more widespread, and there are great scientific and commercial interests in improving their performance by designing low noise receivers and developing SNR enhancing acquisition techniques.

1.2 The Raman effect

Backscatter phenomena occurring inside the fibre are due to the interaction between the photons of the launched laser pulse and the fibre itself. They are usually classified according to the relation between the frequencies of backscattered and incident photons. If these frequencies are equal, the phenomenon is referred to as an *elastic* process, whereas if they are different, as an *inelastic* process. Elastic processes are usually grouped under the name of *Rayleigh scattering*, whereas inelastic ones – such as Raman [14] or Brillouin scattering [15] – are indicated with the name of the particular physical effect on which they are based.

Elastic and inelastic processes occur simultaneously. In particular, Rayleigh and Raman scattering are based on the interaction between photons and fibre molecules [16]. An incident photon can be thought as an electromagnetic wave which is able to induce an oscillating electric dipole moment on a molecule according to its specific polarizability. In particular, if the polarizability does not depend on the molecule vibrations, the molecule is called *Raman–inactive*, otherwise *Raman–active*. When a photon interacts with a Raman–inactive molecule, Rayleigh backscatter occurs: the induced dipole oscillates at the same frequency of the incident photon, i.e. a photon with the same frequency is backscattered. On the other hand, when a photon interacts with a Raman–active molecule, Raman backscatter occurs instead: the frequency of the induced dipole is modulated by the molecule vibrations, and hence a photon with a different frequency is backscattered.

These different types of interactions can be summarized by the simplified energy transfer diagram reported in Figure 1.1 [17]. In this diagram, discrete quantum vibrational states of molecules are indicated with E_1 and E_2, and virtual prohibited states with E_3 and E_4. Rayleigh scattering occurs when an incident photon with frequency ν_0 interacts with a Raman–inactive molecule in the basic E_1 level. The molecule is excited to the $E_3 = E_1 + h\nu_0$ state (h is the Planck's constant), but as this state is prohibited, it immediately decays to the E_1 state, and a back–propagating photon with a frequency ν_0 is exited. It is important to point out that, being E_3 a virtual state, in this process no real absorption takes place, but only light scattering. Moreover, virtual energy levels are not fixed, but they float according to the energy of the incident photon.
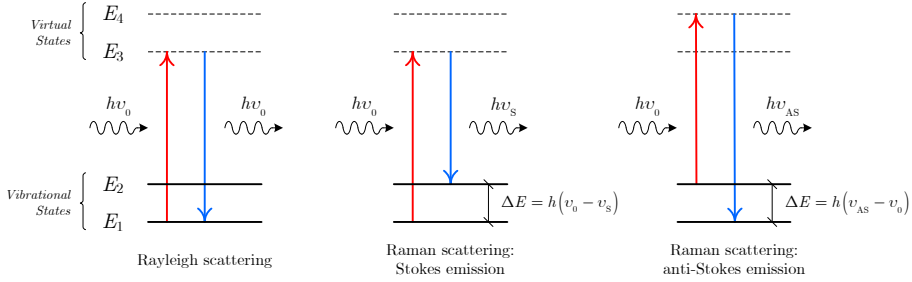
Figure 1.1: Energy transfer diagram for Rayleigh and Raman scattering

If instead the molecule is Raman–active, once excited to the virtual E_3 state, it decays to the E_2 level, and a photon with a frequency $\nu_S < \nu_0$ is exited. This is the Raman scattering, and the produced photon belongs to the so–called *Stokes emission*. The energy difference $\Delta E = E_2 - E_1 = h(\nu_0 - \nu_S)$ is converted into molecule vibration, which corresponds to a phonon emission.

Finally, if an incident photon interacts with a Raman–active molecule which already is in the excited E_2 vibrational state, the molecule is further excited to the $E_4 = E_2 + h\nu_0$ level, and then decays to E_1. In this case, a photon with a frequency $\nu_{AS} > \nu_0$ is exited. This is still the Raman scattering, but the produced photon now belongs to the so–called *anti–Stokes emission*. The energy difference can be written as $\Delta E = h(\nu_{AS} - \nu_0)$, and comparing this expression with the previous one, it can be verified that ν_S and ν_{AS} are symmetrical around ν_0.

The intensities of Stokes and anti–Stokes signals depend not only on the intensity of the incident laser pulse, but also on the occupancy of E_1 and E_2 vibrational energy levels, which is regulated by the Bose–Einstein phonon distribution, and hence by temperature. If we indicate with n_i the number of phonons in the i–th state with energy E_i, it follows that

$$n_i = \frac{g_i}{\exp\left[\dfrac{E_i}{kT}\right] - 1} \tag{1.1}$$

in which g_i is the degeneracy coefficient of E_i, k is the Boltzmann's constant, and T is the absolute temperature. As phonons are massless particles, no chemical potential is included in (1.1). In particular, as $E_1 < E_2$, for a given temperature the E_1 level is more populated than E_2, hence the intensity of

Stokes emission is always higher than the anti–Stokes one. Nevertheless, both signals produced by Raman scattering are very weak if compared to Rayleigh backscattering, and even extremely weak if compared to the launched laser pulse (about eight orders of magnitude [7]), so that high performance optical filters and optoelectronic receivers are required when implementing Raman–based DTS systems.

1.3 Optical Time–Domain Reflectometry (OTDR)

1.3.1 Basic principles

As mentioned before, the Optical Time–Domain Reflectometry (OTDR) is the fundamental operating principle of a Raman–based DTS system. To better understand this technique, it is worth studying the optical backscattered signal received by the system from an analytical point of view.

In general, as optical power propagates along the fibre, regardless of the direction, it decays with a factor $\exp\left[-\alpha D\right]$, in which D is the covered distance and α is the total attenuation coefficient, which takes into account both absorption and forward/side–scattering effects in the fibre at the considered wavelength. As a consequence, if z is the fibre axis, the decay of the forward–propagating laser pulse launched at $z = 0$ with initial power P_0 is described by the following relation

$$P(z) = P_0 \cdot \exp\left[-\alpha_\mathrm{P}\, z\right] \tag{1.2}$$

where α_P is the attenuation coefficient at the pulse wavelength. On the other hand, the backscattered power $\mathrm{d}P_\mathrm{BS}$ produced at a given position z in an infinitesimal fibre segment $\mathrm{d}z$ is given by

$$\mathrm{d}P_\mathrm{BS} = \beta \cdot P(z)\, \mathrm{d}z \tag{1.3}$$

where β is the fibre backscattering coefficient at the considered wavelength. In particular, this coefficient depends not only on some fibre properties, such as the numerical aperture and the refractive index profile, but – for Stokes and anti–Stokes signals – also on temperature.

7
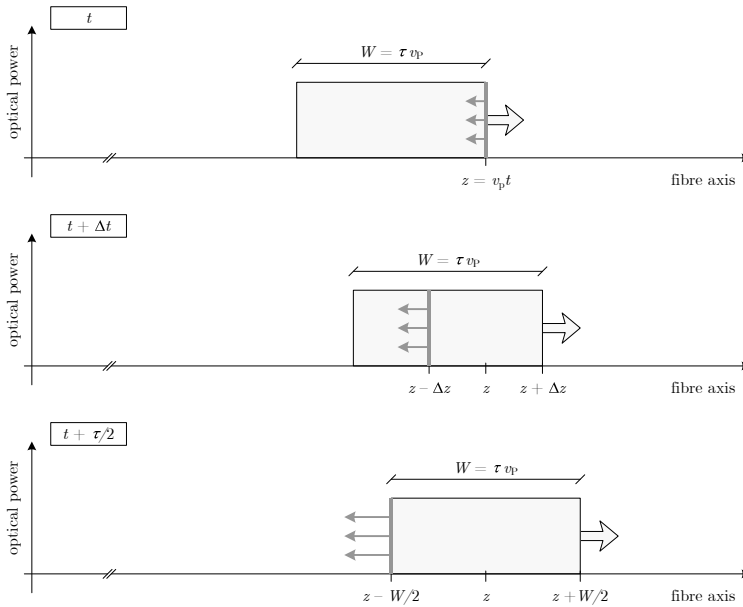
Figure 1.2: Analysis of the backscattered signal received with the OTDR technique

Now consider a rectangular laser pulse with an initial power P_0 and a duration τ. Its spatial width W is given by

$$W = \tau \, v_\mathrm{P} = \tau \frac{c}{n_\mathrm{P}} \tag{1.4}$$

where v_P is the pulse group velocity, c is the speed of light in vacuum, and n_P is the fibre group index.

1.3.2 Rayleigh backscatter

To determine the analytical expression of the Rayleigh backscatter signal, suppose that at a given time t the leading edge of the launched pulse is at location $z = v_\mathrm{P}t$, as represented in the first plot of Figure 1.2. For simplicity's sake, the exponential decay described by (1.2) is not shown in the plot, but it will be taken into account in the following.

Backscattered photons produced at the pulse leading edge will start travelling with velocity v_P back to the fibre input, where they will be detected by the

system at the time $2t$, i.e. the OTDR *round–trip* time. However, along their path, they will meet other new–generated photons, backscattered by other infinitesimal pulse segments, which will arrive at the fibre input at $2t$ as well.

An example of this process is reported in the second plot of Figure 1.2, which refers to the time $t + \Delta t$, with $\Delta t < \tau/2$. At this time, the pulse and the first backscattered photons have moved forwards and backwards, respectively, of the same quantity $\Delta z = \Delta t \cdot v_\mathrm{P}$. New backscattered photons generated by the infinitesimal pulse segment at $z - \Delta z$ will join the other ones travelling backwards, and all together will arrive at the fibre input at the time $2t$. This accumulation process continues till backscattered photons reach the pulse trailing edge, at the time $t + \tau/2$ and at location $z - W/2$ (see the third plot of Figure 1.2). This means that the total backscattered power P_P at the pulse wavelength, produced by the Rayleigh backscatter and received by the system at $2t$, is actually due to the integral of backscatter photons provided by (1.3) and generated between the locations $z - W/2$ and z. Hence it follows

$$P_\mathrm{P}(z) = \beta_\mathrm{P} P_0 \int_{z-W/2}^{z} \exp\left[-2\alpha_\mathrm{P}\,\xi\right] \mathrm{d}\xi \tag{1.5}$$

in which β_P is the Rayleigh backscattering coefficient for the pulse wavelength and $2\alpha_\mathrm{P}$ is the round–trip attenuation coefficient.

To reach this result, the implicit assumption that $z \geq W/2$ has been made. If instead we consider a distance $z < W/2$, i.e. we assume that the pulse trailing edge has not completely left the fibre input yet, the lower integration limit has to be coerced to 0. This aspect can be taken into account by simply setting the lower limit to $\max(0,\ z - W/2)$, so that

$$P_\mathrm{P}(z) = \beta_\mathrm{P} P_0 \int_{\max(0,\ z-W/2)}^{z} \exp\left[-2\alpha_\mathrm{P}\,\xi\right] \mathrm{d}\xi \tag{1.6}$$

Equation (1.6) is usually presented in a slightly different form [18], which can easily be obtained by applying the following variable change

$$\xi = z - \frac{\zeta}{2}, \quad \mathrm{d}\xi = -\frac{\mathrm{d}\zeta}{2} \tag{1.7}$$

thus resulting

$$P_{\mathrm{P}}(z) = \frac{\beta_{\mathrm{P}} P_0}{2} \int\limits_{0}^{\min(2z,\,W)} \exp\left[-2\alpha_{\mathrm{P}}\left(z - \frac{\zeta}{2}\right)\right] \mathrm{d}\zeta \qquad (1.8)$$

Then, performing the integration it follows that

$$P_{\mathrm{P}}(z) = \begin{cases} \dfrac{\beta_{\mathrm{P}} P_0}{2\alpha_{\mathrm{P}}}\left[1 - \exp\left(-2\alpha_{\mathrm{P}} z\right)\right] & \text{if } z < W/2 \\[2ex] \dfrac{\beta_{\mathrm{P}} P_0}{2\alpha_{\mathrm{P}}} \exp\left(-2\alpha_{\mathrm{P}} z\right)\left[\exp\left(\alpha_{\mathrm{P}} W\right) - 1\right] & \text{if } z \geq W/2 \end{cases} \qquad (1.9)$$

It is worth noting that for short pulses such that $\alpha_{\mathrm{P}} W \ll 1$, the exponential term $[\exp\left(\alpha_{\mathrm{P}} W\right) - 1]$ can be safely approximated to the Taylor's first–order by $\alpha_{\mathrm{P}} W$. This allows to simplify the second equation in (1.9) to

$$P_{\mathrm{P}}(z) = \frac{\beta_{\mathrm{P}} P_0}{2} W \exp\left(-2\alpha_{\mathrm{P}} z\right) \qquad (1.10)$$

This last equation shows that if $\alpha_{\mathrm{P}} W \ll 1$, the backscattered power received by the system is proportional to the pulse width W, and hence to the pulse duration τ [18]. However, as already deducible from (1.8), the greater W, the larger the integration interval, and the lower the precision of the collected information about a given fibre section. This is an important aspect of a fundamental trade–off between backscatter intensity and spatial resolution, which will be discussed with further details in Section 1.5.

Finally, the dependence of P_{P} on time can be expressed by replacing z in (1.9) with $t \cdot v_{\mathrm{P}}/2$, in which the factor 2 takes the OTDR optical round–trip path into account. It results

$$P_{\mathrm{P}}(t) = \begin{cases} \dfrac{\beta_{\mathrm{P}} P_0}{2\alpha_{\mathrm{P}}}\left[1 - \exp\left(-\alpha_{\mathrm{P}} v_{\mathrm{P}} t\right)\right] & \text{if } t < \tau \\[2ex] \dfrac{\beta_{\mathrm{P}} P_0}{2\alpha_{\mathrm{P}}} \exp\left(-\alpha_{\mathrm{P}} v_{\mathrm{P}} t\right)\left[\exp\left(\alpha_{\mathrm{P}} v_{\mathrm{P}} \tau\right) - 1\right] & \text{if } t \geq \tau \end{cases} \qquad (1.11)$$

Equation (1.11) proves the fibre low–pass behaviour. Indeed, it can be noted that (1.11) represents the response of a low–pass first–order filter, with a time constant $1/\alpha_{\mathrm{P}} v_{\mathrm{P}}$, to an incoming rectangular pulse of duration τ. In particular, the first equation describes the initial rising transient, which starts at $t = 0$, whereas the second equation describes the falling one, which starts at $t = \tau$.

1.3.3 Raman backscatter

The previous analysis can be repeated also for Stokes and anti–Stokes signals. This time, however, backscattered signals have different wavelengths with respect to the probing pulse, and hence different velocities, attenuation coefficients and backscatter coefficients. These latter also depend on temperature T, which in its turn is a function of z. In the following, the parameters related to Stokes and anti–Stokes signals will be indicated with S and AS subscripts, respectively. In particular, it results for β [19]

$$\beta_{\mathrm{S}}(T) = \frac{\beta_{\mathrm{S0}}}{\lambda_{\mathrm{S}}^4} \; \frac{\exp\left[\dfrac{h\Delta\nu}{kT}\right]}{\exp\left[\dfrac{h\Delta\nu}{kT}\right] - 1} \tag{1.12}$$

$$\beta_{\mathrm{AS}}(T) = \frac{\beta_{\mathrm{AS0}}}{\lambda_{\mathrm{AS}}^4} \; \frac{1}{\exp\left[\dfrac{h\Delta\nu}{kT}\right] - 1} \tag{1.13}$$

where β_{S0} and β_{AS0} do not depend on temperature, λ_{S}^4 and λ_{AS}^4 are the Stokes and anti–Stokes wavelengths, respectively, and $\Delta\nu = \nu_{\mathrm{AS}} - \nu_{\mathrm{S}}$. An equation similar to (1.5) can be written by observing that, this time, the photons backscattered at z will reach the pulse trailing edge at

$$z - \frac{W}{1 + n_\star/n_{\mathrm{P}}}$$

where $n_\star$, with $\star = \{\mathrm{S}, \mathrm{AS}\}$, is the refractive index at the considered Stokes or anti–Stokes wavelength. It follows that

$$P_\star(z) = P_0 \int\limits_{z - W/(1 + n_\star/n_{\mathrm{P}})}^{z} \beta_\star\left(T(\xi)\right) \exp\left[-(\alpha_{\mathrm{P}} + \alpha_\star)\xi\right] \mathrm{d}\xi \tag{1.14}$$

To simplify the analysis, we can suppose that T is constant within the integration interval, so that $\beta_\star$ can be pulled out of integral. In addition, the following variable change can be applied

$$\xi = z - \frac{\zeta}{1 + n_\star/n_{\mathrm{P}}}, \quad \mathrm{d}\xi = -\frac{\mathrm{d}\zeta}{1 + n_\star/n_{\mathrm{P}}} \tag{1.15}$$

thus resulting

$$P_\star(z) = \frac{\beta_\star\left(T(z)\right)P_0}{1 + n_\star/n_{\mathrm{P}}} \int\limits_0^W \exp\left[-(\alpha_{\mathrm{P}} + \alpha_\star)\left(z - \frac{\zeta}{1 + n_\star/n_{\mathrm{P}}}\right)\right]\mathrm{d}\zeta \qquad (1.16)$$

and hence

$$P_\star(z) = \frac{\beta_\star\left(T(z)\right)P_0}{\alpha_{\mathrm{P}} + \alpha_\star} \exp\left[-(\alpha_{\mathrm{P}} + \alpha_\star)z\right]\left[\exp\left[\frac{\alpha_{\mathrm{P}} + \alpha_\star}{1 + n_\star/n_{\mathrm{P}}}W\right] - 1\right] \qquad (1.17)$$

It can be verified that this last equation turns back into (1.9) if the Rayleigh backscattering wavelength is considered.

1.4 Measurement techniques

1.4.1 Single–ended configuration

The typical measurement configuration of DTS systems is the one reported in Figure 1.3, in which the near–end of the fibre is connected to the instrument, and the far–end of the fibre is terminated. To extract the temperature information from ODTR signals, a possible solution is to evaluate the ratio $R(z)$ between anti–Stokes and Stokes powers [20,21], so as to obtain a relation between the acquired data and temperature without any dependence on the pulse peak power and on second–order effects. By specifying (1.17) for S and AS, and substituting (1.13), it results

$$R(z) = \frac{P_{\mathrm{AS}}(z)}{P_{\mathrm{S}}(z)} = \kappa \cdot \exp\left[(\alpha_{\mathrm{S}} - \alpha_{\mathrm{AS}})z\right]\exp\left[-\frac{h\Delta\nu}{kT(z)}\right] \qquad (1.18)$$

where κ is given by

$$\kappa = \left[\frac{\lambda_{\mathrm{S}}}{\lambda_{\mathrm{AS}}}\right]^4 \frac{\beta_{\mathrm{AS0}}}{\beta_{\mathrm{S0}}} \frac{\alpha_{\mathrm{P}} + \alpha_{\mathrm{S}}}{\alpha_{\mathrm{P}} + \alpha_{\mathrm{AS}}} \frac{\exp\left[\dfrac{\alpha_{\mathrm{P}} + \alpha_{\mathrm{AS}}}{1 + n_{\mathrm{AS}}/n_{\mathrm{P}}}W\right] - 1}{\exp\left[\dfrac{\alpha_{\mathrm{P}} + \alpha_{\mathrm{S}}}{1 + n_{\mathrm{S}}/n_{\mathrm{P}}}W\right] - 1} \qquad (1.19)$$

Unfortunately, the direct use of (1.18) for the evaluation of the fibre absolute temperature is unfeasible in practice, as it is difficult to exactly know both κ
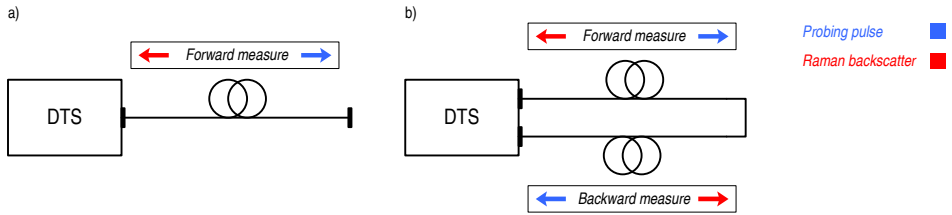
Figure 1.3: Typical measurement configurations for Raman–based DTS systems: a) single–ended, b) double–ended (loop)

and $(\alpha_{\mathrm{S}} - \alpha_{\mathrm{AS}})$ coefficients, which depend on the particular fibre used for the measurement. As a consequence, a calibration technique is usually adopted. At first, the fibre is set at a known reference temperature $T_{\mathrm{ref}}(z)$ and the reference ratio $R_{\mathrm{ref}}(z)$ is acquired. Then, the ratio $R(z)/R_{\mathrm{ref}}(z)$ is evaluated and $T(z)$ is extracted as a function of $T_{\mathrm{ref}}(z)$ [22]. It results

$$T(z) = \left\{ \frac{1}{T_{\mathrm{ref}}(z)} - \frac{k}{h\Delta\nu} \ln\left[\frac{R(z)}{R_{\mathrm{ref}}(z)} \right] \right\}^{-1} \tag{1.20}$$

1.4.2 Double–ended configuration

The previously described calibration technique is based on the assumption that κ and $(\alpha_{\mathrm{S}} - \alpha_{\mathrm{AS}})$ coefficients do not vary with time, so that only a single reference measurement is required to calibrate the instrument for all its lifetime. This unfortunately is not always true. Indeed, in some harsh environments such as nuclear plants or geothermal wells, it has been shown that these coefficients are affected by the fibre exposure to ionizing radiations [6], high temperatures, and damp or Hydrogen–rich atmospheres [23]. When this happens, the calibration is no more valid. In particular, as the $\exp\left[(\alpha_{\mathrm{S}} - \alpha_{\mathrm{AS}})z\right]$ term is not entirely cancelled when the ratio $R(z)/R_{\mathrm{ref}}(z)$ is evaluated, a measurement error which increases with z is introduced.

A typical solution to this problem is represented by the *double–ended* (or *loop*) configuration reported in Figure 1.3b. In this case, both ends of the fibre are connected to the instrument. Laser pulses are launched alternatively from each fibre end, so that Stokes and anti–Stokes signals in both forward and backward directions can be acquired. This leads to an interesting advantage.

13

If we indicate with $z = 0$ the fibre input when pulses are launched in the forward direction, the anti–Stokes to Stokes ratio in both forward and backward directions can be respectively written as

$$R^{\text{fw}}(z) = \kappa \cdot \exp\left[(\alpha_{\text{S}} - \alpha_{\text{AS}})z\right] \exp\left[-\frac{h\Delta\nu}{kT(z)}\right] \tag{1.21}$$

$$R^{\text{bw}}(z) = \kappa \cdot \exp\left[(\alpha_{\text{S}} - \alpha_{\text{AS}})(L - z)\right] \exp\left[-\frac{h\Delta\nu}{kT(z)}\right] \tag{1.22}$$

where L is the total fibre length. A new ratio $R^{\text{lp}}(z)$ for the loop configuration can now be defined as the geometric mean between $R^{\text{fw}}(z)$ and $R^{\text{bw}}(z)$. It follows that

$$R^{\text{lp}}(z) = \sqrt{R_{\text{fw}}(z) \cdot R_{\text{bw}}(z)} = \kappa \cdot \exp\left[(\alpha_{\text{S}} - \alpha_{\text{AS}})\frac{L}{2}\right] \exp\left[-\frac{h\Delta\nu}{kT(z)}\right] \tag{1.23}$$

in which the dependency of the $(\alpha_{\text{S}} - \alpha_{\text{AS}})$ exponential term on z is cancelled. As in the previous case, the temperature can be estimated using the reference loop ratio $R^{\text{lp}}_{\text{ref}}(z)$ measured at the temperature $T_{\text{ref}}(z)$, so as to obtain

$$T(z) = \left\{ \frac{1}{T_{\text{ref}}(z)} - \frac{k}{h\Delta\nu} \ln\left[\frac{R^{\text{lp}}(z)}{R^{\text{lp}}_{\text{ref}}(z)}\right] \right\}^{-1} \tag{1.24}$$

The important difference with respect to the single–ended configuration is that if κ or $(\alpha_{\text{S}} - \alpha_{\text{AS}})$ coefficients vary with time, the calibration mismatch does not produce measurement errors increasing with z, but just scales the $R^{\text{lp}}(z)/R^{\text{lp}}_{\text{ref}}(z)$ ratio of a constant factor. This would translate into a constant offset on $T(z)$, which can easily be compensated using, for instance, an external punctual temperature sensor at a known location.

1.5   Performance parameters

The two main performance parameters of a DTS systems are the *temperature resolution* and the *spatial resolution*, and they are bounded by some important trade–offs.

The temperature resolution is defined as the standard deviation of the temperature measurement, and it is evaluated over several consecutive temperature profiles extracted by the system, as a function of z. From a numerical

point of view, the temperature resolution is deeply related to the SNR of Stokes and anti-Stokes traces, i.e. to the intensity of backscattered signals and to the total noise of the optoelectronic receiver. In particular, as traces decay exponentially, their SNR, and hence the temperature resolution of the system, get worse with distance. This is the reason why the average of some million of consecutive traces is usually needed to achieve a sub–degree resolution beyond 20 km, with a non–negligible increment of the measurement time.

To enhance the SNR of backscattered traces, the receiver noise should be reduced and the backscatter power should be increased. According to (1.9) and (1.10), the latter can be accomplished by increasing the laser pulse P_0 and W parameters. Unfortunately, P_0 cannot be freely increased, but it must be kept lower than a given threshold which depends on the fibre type [12], so as to avoid the onset of non–linear effects, such as the *stimulated Raman scattering*, which make the temperature measurement unfeasible. On the other hand, equation (1.8) shows that a large W implies a wide integration interval, and hence a poor spatial resolution.

The spatial resolution of the system is defined as the distance, measured on the reconstructed temperature profile, required to detect a temperature step in the optical fibre. This distance is measured between the 10 % and the 90 % of the reconstructed step. It is clear that the spatial resolution is related to the response time of the optoelectronic receiver, and hence it can be improved by extending the system bandwidth. This however increases the noise level, thus affecting the SNR and the temperature resolution.

Finally, two other performance parameters are the *sampling resolution* and the *distance range*. The first is related to the sampling frequency of the analog–to–digital converter used in the system, and it is defined as the distance between two consecutive samples of the reconstructed temperature profile. The latter is instead related to the memory depth of the system, and it is defined as the maximum distance that can be acquired at a given sampling resolution.

1.6   State–of–the–art DTS systems

Using the parameters defined in the previous paragraph, it is interesting to compare the performance of some state–of–the–art Raman–based DTS systems available on the market. In particular, focusing on the so–called long–range systems, i.e. on those system with are able to reach a measuring range greater than 20 km, the Gemini™ system by SensorTran [24] and the DiTemp® system by Smartec [25] have been selected.

The Gemini™ system has a modular architecture, build up by two main parts. The first comprises all optical and optoelectronic devices, whereas the latter includes the acquisition and the PC–based processing electronics. The choice of a PC platform is commonly adopted by DTS manufacturers, as it provides advanced hardware and software resources, not only for processing the acquired data but also for storing, displaying and sharing the extracted temperature profiles, in a very cost–effective way. Each Gemini™ module can also be separately upgraded to improve the overall system performance. In particular, the acquisition module is rather flexible, as it allows the user to dynamically set the trade–off between the sampling resolution and the distance range.

On the other hand, the DiTemp® system, which is also based on a PC platform, features a compact fully–integrated architecture. The manufacturer provides four different models, which allows to reach a distance range of 5 km, 8 km, 10 km and 30 km, respectively.

In Table 1.1, the parameters of the Gemini™ system, featuring the *High Speed & High Intensity Upgrade*, and those of the DiTemp® system in its most performing *Extreme Range* version, are summarized and compared. In particular, for those parameters that are bounded each other, as described in Section 1.5, the best trade–off is reported. A dash is instead used whenever a value is unavailable or not specified my the manufacturer.

It can be observed that the first system shows a slightly shorter distance range. In addition, according to the range the user selects, different values of both the spatial and the temperature resolution are available. Within the 8–10 km range, the first system features a spatial resolution of 1.5 m, whereas the latter of just 1 m. As expected, this difference affects the temperature reso-

Table 1.1: Comparison of state–of–the–art DTS system parameters

		SensorTran Gemini™ with High Speed & High Intensity Upgrade			Smartec DiTemp® Extreme Range Version		
Distance range [km]		27			30		
Sampling resolution [m]		0.25			–		
Spatial resolution [m]	8 km	1.5			1		
	10 km	1.5			1		
	27 km	1			–		
	30 km	–			2		
		Acquisition time					
		10 s	60 s	300 s	10 s	60 s	300 s
Temperature resolution [K]	8 km	0.3	0.15	0.1	1	0.4	0.2
	10 km	0.5	0.25	0.15	2.25	1	0.5
	27 km	–	–	2	–	–	–
	30 km	–	–	–	2.7	1	0.7

lution. Indeed, for a given acquisition time, i.e. for a given number of averaged traces, the the first system always shows a better temperature resolution than the latter. On the other hand, within the 27–30 km range the opposite happens. The first system features a spatial resolution of 1 m, whereas the latter of 2 m. After 300 s of acquisition, the first system has reached a temperature resolution of 2 K, the latter of just 0.7 K.

1.7 Research challenges

The main research challenge in the design of Raman–based DTS systems is the improvement of the trade–off between the temperature resolution and the measuring time, for a given spatial resolution. Indeed, a faster and more accurate measurement of the temperature along the fibre may allow DTS systems to be

used in even wider applicative scenarios, including for instance those related to fire detection and prevention in civil or industrial infrastructures. To improve this trade–off, the SNR of the backscattered traces needs to be increased with neither reducing the receiver bandwidth nor enlarging the probing pulse width. As a consequence, once the laser peak power has been maximized, the main research goals are to reduce the total noise of the optoelectronic receiver and to develop advanced acquisition techniques, typically based on laser pulse coding [26] (see Chapter 6), which result more time–efficient than the simple averaging.

As the SNR of traces increases, it could also be interesting to extend the distance range. Indeed, previously unfeasible fibre lengths which required a too long acquisition time to obtain a reasonable temperature resolution, may eventually become easily achievable.

All these aspects have been faced within a research project carried out by the Information Engineering Department of the University of Pisa, in collaboration with Scuola Superiore Sant'Anna, Pisa (Italy). The aim of this project was to design and implement a Raman–based DTS system which showed a performance comparable with – and possibly better than – the state–of–the–art reported in Table 1.1. The following Chapters describe the main issues which have been addressed during the project, analyse the final hardware and software implementation of the system, and report the obtained results. Finally, a new pulse coding technique is presented, and its advantages in terms of SNR enhancing and measurement time reduction are not only theoretically demonstrated but also validated by experimental results.

2 Design Considerations

2.1 System architecture

The conventional PC–based architecture of a Raman DTS system is shown in Figure 2.1. The diagram is divided into three main sections. The first comprises the optical components, the second includes the optoelectronic receivers with the dedicated acquisition, averaging and triggering electronics, and the third is represented by the trace post–processing and temperature displaying algorithms, which are implemented in software on the external PC.

The main optical component is the pulsed laser. Whenever it receives a trigger signal, it produces a laser pulse, which passes through the following

Figure 2.1: Conventional PC–based DTS architecture

Raman filter and enters into the probing fibre. Then, as the pulse propagates along the fibre, backscattered light comes back to the filter. In the backward direction, the filter acts as a Wavelength Division Multiplexer (WDM). Indeed, it selects the Stokes and anti–Stokes wavelengths, and split them through two separate output ports. Stokes and anti–Stokes lights are then routed to the optoelectronic receivers. Here, two photodetectors initially convert optical signals to electric currents. Then, currents are converted to voltages, and hence amplified. These voltages are fed to an A/D acquisition system, which is synchronized to the trigger generator. In this way, each backscattered trace is separately acquired, stored, and provided to the averaging process. Finally, by means of a high–speed data link, the averaged traces are transferred to the PC, where a dedicated software extracts the temperature profile and displays it to the user. In addition, the same software usually allows the user to control and configure several systems parameters, such as the gain of the amplification chain, the samples to be acquired, the number of averages, and so on.

Starting from the example of Figure 2.1, the DTS architecture shown in Figure 2.2 has been designed. The leading idea is to partition the whole system in some functionally–independent hardware units. Each unit features a microcontroller, which is able to communicate with the PC, exchange data, and receive specific configuration settings. In particular, it can be observed that the connection between the system and the PC is actually implemented with by means of two different USB links. The first link is used only for the *Conversion and Acquisition* block, so as to provide all the bandwidth needed to transfer the acquired data, whereas the latter is used to control the other system blocks. Actually, these blocks are not directly reached by USB lines, but they are rather connected each other by means of a simple 2–wire Multiprocessor UART (M–UART) bus [27]. Messages coming from the USB control link are indeed received by the *Control* block, bridged to the M–UART bus, and finally delivered to the recipient block. Here the µCparses the message and accordingly configures the different devices through I2C, SPI or general–purpose I/O lines. In the following, the main aspects related to the design of each block are summarized.
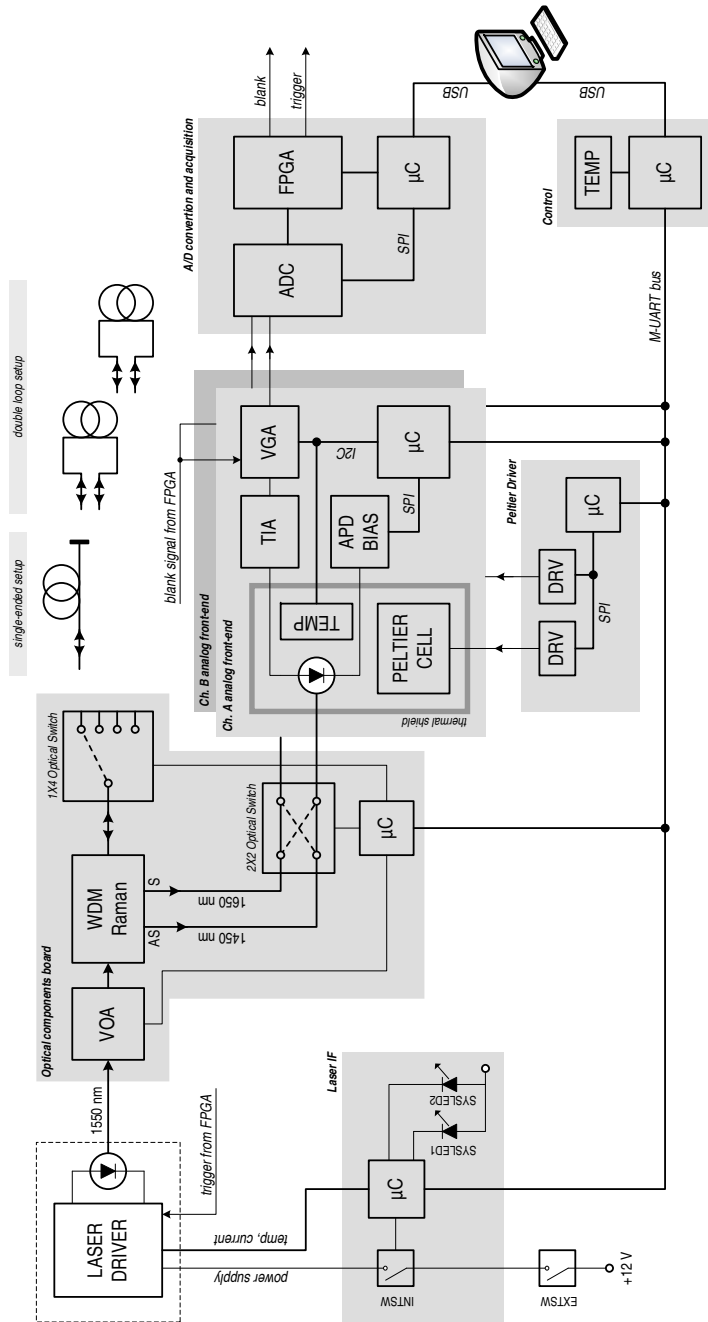
Figure 2.2: The final implemented DTS architecture

2.2 Optical components

The first task to carry out when designing the optical part of the system is the choice of the laser operating wavelength λ_P, which indirectly affects the total backscattered power, and hence the overall performance. As shown in Section 1.3, $\alpha_\star$ and $\beta_\star$ coefficients depend on their related wavelengths. However, as $\Delta\nu$ is fixed and symmetrical around λ_P, they actually depend only on λ_P. This means that, once P_0 is determined, if the fibre is at a constant temperature the backscattered power $P_\star(L)$ depends only on L and λ_P (see Eq. 1.17). With this assumption, it can be proved that, for a given distance L_0, there is an optimum pulse wavelength λ_0 which maximizes the backscattered power [28]. L_0 and λ_0 are bounded by the following equation

$$L_0 = \frac{\lambda_0^4}{\eta\left[(1 - \Delta w\,\lambda_0)^4 - \Delta w\,\lambda_0 + 1\right]} \tag{2.1}$$

where η [m³] is the constant of proportionality used to approximate the dependency of α_P on λ [29], i.e.

$$\alpha_P(\lambda) = \frac{\eta}{\lambda^4} \tag{2.2}$$

and Δw [m^{-1}] is the Raman shift expressed in wavenumbers, i.e.

$$\Delta w = \begin{cases} \dfrac{1}{\lambda_{AS}} - \dfrac{1}{\lambda_P} < 0 & \text{for anti–Stokes backscatter} \\[2mm] \dfrac{1}{\lambda_S} - \dfrac{1}{\lambda_P} > 0 & \text{for Stokes backscatter} \\[2mm] 0 & \text{for Rayleigh backscatter} \end{cases} \tag{2.3}$$

Typical silica fibres available on the market report $\alpha_P = 0.2\,\text{dB/km}$ at $\lambda = 1.55\,\mu\text{m}$ [18]. To convert [dB/km] attenuation values to [km^{-1}] values, it can be observed that Eq. (1.2) can be also written as

$$P(z) = P_0 \cdot e^{-\alpha z} = P_0 \cdot 10^{-(\alpha_{dB}/10)z} \tag{2.4}$$

so that $\alpha = \alpha_{dB}(\ln 10)/10$. This means that the previous value can be expressed also as $\alpha_P = 0.0461\,\text{km}^{-1}$, thus giving $\eta = 2.6581 \times 10^{-28}\,\text{m}^3$. Now, considering the weaker anti–Stokes backscatter, for silica fibres it is typically
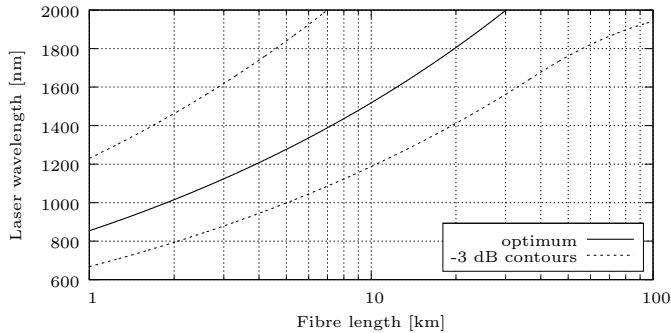
Figure 2.3: Optimum laser wavelength maximizing the anti–Stokes backscatter, and its $-3\,\mathrm{dB}$ contours, as a function of the fibre length ($\alpha = 0.2\,\mathrm{dB/km}$ at $\lambda = 1.55\,\mathrm{\mu m}$, $\Delta w = 440\,\mathrm{cm}^{-1}$)

$\Delta w = 440\,\mathrm{cm}^{-1}$ [28]. If we choose $L_0 = 30\,\mathrm{km}$, by inverting (2.1) it results $\lambda_0 = 1.92\,\mathrm{\mu m}$. This is the optimal laser wavelength that should be used. However, following the remarks reported in [28], it can be shown that if we set $\lambda_\mathrm{P} = 1.55\,\mathrm{\mu m}$, the penalty in terms of backscattered power reduction for using a non–optimal laser wavelength is less than $3\,\mathrm{dB}$ (see Figure 2.3). As a consequence, accepting this not too severe trade–off allows for the use of a wide family of optical components already available on the market, which represents a great advantage in terms of development time and costs.

The adopted laser is hence a $1.55\,\mathrm{\mu m}$ solid–state Q–switched Erbium–doped pulsed fibre laser [30,31], which is able to launch pulses with a $400\,\mathrm{nJ}$ energy. The nominal pulse width τ and peak level P_0 are $10\,\mathrm{ns}$ and $40\,\mathrm{W}$, respectively. This parameters are actually achievable only if the triggering frequency is kept lower than a hundred of kHz. Indeed, for technology–related issues, at higher frequencies τ increases and P_0 decreases, but the pulse energy, i.e. the τP_0 product, is preserved [32].

The laser receives the trigger signal directly from the *A/D Conversion and acquisition block*, and provides information on the temperature and the current of its internal pump diode to the *Laser IF block*. This block controls the $12\,\mathrm{V}$ power supply of the laser, which is enabled by means of two series switches. The first is controlled by the block µC, and hence by the software running on the PC, the latter is mounted externally to the system, so that it can easily be operated by the user if an emergency stop or an instantaneous laser shutdown is required. The µC also controls two leds which are used by the PC software

23

to the working status of the system.

Next to the laser, an electric–controlled Variable Optical Attenuator (VOA) is used to reduce the pulse peak power, so as to guarantee that the fibre non–linear threshold (see Section 1.5) is not exceeded. In particular, we used a 50/125 graded–index multimode fibre, which allows for a peak power of about 20 W. From an electrical point of view, the VOA features a miniaturized stepper–motor, which is controlled by the µC mounted on the *Optical components board* through a proper motor–driver, not shown in Figure 2.2.

The same µC also drives two optical switches. The first allows to select one of four available probing fibre outputs. This way, the system is able to alternately measure the temperature from up to four different fibres in the single–ended configuration, or from up to two fibres in the double–ended configuration. These fibres can of course be installed in a star–topology, so as to virtually extend the actual distance range in more than one direction.

The second switch allows to swap the two receivers. This allows to acquire consecutive Stokes and anti–Stokes traces from the same receiver for short periods in which the fibre temperature can be considered constant. This way, to estimate the temperature, the ratio between these traces can be used instead of the conventional one between simultaneously acquired traces. The benefit is that the calibration procedure can be simplified, because in this case the gain mismatch between receivers does not need to be compensated.

2.3 Optoelectronic receiver

2.3.1 Avalanche photodiodes

To receive the weak optical signals produced by the Raman backscatter, each receiver has been equipped with an Avalanche Photodiode (APD). This choice is due to the fact that APDs are able to provide a higher *responsivity*, i.e. a higher ratio between the produced photocurrent and the incident power, if compared to other solid–state detectors, such as p–n or p–i–n photodiodes [33].

The working principle of APDs relies on the conversion of incident photons to electron–hole pairs, and on their subsequent avalanche multiplication. Considering at first a simple reverse–biased p–n junction, incident photons with a greater energy than the semiconductor band–gap can be absorbed to produce electron–hole pairs. These new carriers may be localized inside or outside the depletion region. In the first case, they are immediately accelerated by the high electric field and drift to the p or n side, whereas in the latter case they have to diffuse to the boundary of the depletion region before being accelerated. Thanks to this two contributions, a photocurrent proportional to the incident optical power is produced. However, the contribution of the diffusion current is actually considered a drawback, as its inherently slow generation process can distort the temporal response of the p–n photodiode thus limiting its bandwidth.

To address this issue, it is necessary to decrease the width of p and n regions, and increase the one of the depletion region, so that most of incident optical power is absorbed inside it. This is achieved by p–i–n photodiodes, in which an intrinsic i layer is inserted between the p–n junction [34]. As in general the depletion region extends more to the less doped side of a junction, in p–i–n structures this region spreads throughout the i layer, so that its width can be controlled by changing the i layer thickness. As a result, most of the optical incident power is absorbed inside the i layer, and the drift component of the photocurrent dominates over the diffusion one.

APD structure further improves the performance of p–i–n photodiodes by including an additional multiplication layer, in which secondary electron–hole pairs are generated through *impact ionization* [35]. This new layer is of p type and it is inserted between the i and n regions to form the p^+–i–p–n^+ structure reported in Figure 2.4a. The i layer still acts as the depletion region, in which most of the incident photons are absorbed and primary electron–hole pairs are generated. However, as primary carriers reach the narrow p–n^+ multiplication region, they are accelerated by a very high electric field (usually 2×10^5–4×10^5 V/cm for a reverse bias voltage is in the order of a hundred of volt, see 2.4b), and acquire enough kinetic energy to extract new secondary carriers from the valence band by impacting on them. Secondary carriers can then be accelerated as well, and may produce in their turn even more carriers, in a sort of chain reaction process, that is the avalanche multiplication.

From an analytical point of view, it can be proved that the APD multipli-
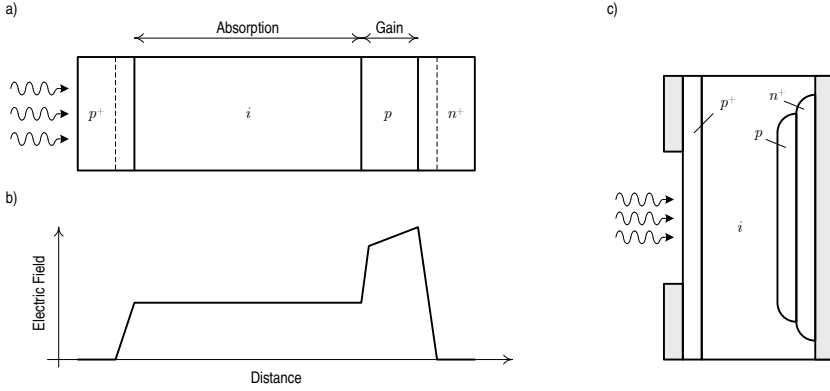
Figure 2.4: An example of reverse–biased APD structure (a) with the electric field distribution (b) and a typical layout (c)

cation factor M due to the avalanche process can be written as [33]

$$M = \frac{1 - k_A}{\exp\left[-(1 - k_A)\alpha_e d\right] - k_A} \tag{2.5}$$

where $k_A = \alpha_h/\alpha_e$ is the ratio between the the impact ionization coefficients of holes and electrons, respectively, and d is the width of the multiplication region. Taking M into account, the total APD photocurrent I_{p} is given by

$$I_{\mathrm{p}} = R_{\mathrm{APD}}P_{\mathrm{in}} = MRP_{\mathrm{in}} \tag{2.6}$$

where P_{in} is the optical incident power, and $R_{\mathrm{APD}} = MR$ is the total APD responsivity, being R the responsivity of the basic p–i–n structure.

In addition to the photocurrent I_{p}, APDs also produce the dark current I_{d}. This current is also present in p–i–n or p–n photodiodes, and it can be thought as the reverse saturation current of the junction, which is mainly due to thermally generated electron–hole pairs. Even if it is typically lower than $10\,\mathrm{nA}$, its contribution may be not neglected when noise aspects have to be analysed.

2.3.2 Photoreceiver noise

To evaluate the noise performance of APDs, two main contributions have to be taken into account when designing an APD–based receiver, i.e. shot noise and thermal noise. The first is due to the photocurrent generation and to the avalanche multiplication processes within the APD, the latter is related to the electronic front–end used to amplify the APD current. In particular, the first stage of the front–end is usually represented by a transimpedance amplifier, whose basic scheme comprising the equivalent noise generators is reported in Figure 2.5. From the diagram, the total current coming into the amplifier can be written as

$$I(t) = I_{\mathrm{p}} + i_{\mathrm{s}}(t) + i_{\mathrm{t}}(t) \tag{2.7}$$

where $i_{\mathrm{s}}(t)$ and $i_{\mathrm{t}}(t)$ represent the shot and thermal noise currents, respectively. For shot noise, it can be shown that the variance σ_{s}^2 of $i_{\mathrm{s}}(t)$ can be expressed as [36]

$$\sigma_{\mathrm{s}}^2 = 2qM^2 F_A \left(R P_{\mathrm{in}} + I_{\mathrm{d}} \right) B \tag{2.8}$$

in which the effect of both the photocurrent and the dark current are included. B is the receiver bandwidth, and F_A is the *excess noise factor* of the APD, defined as

$$F_A(M) = k_A M + (1 - k_A)(2 - 1/M) \tag{2.9}$$

In particular, as $0 < k_A < 1$, F_A increases with M. For this reason, the excess noise factor is often approximated with $F_A(M) = M^x$, where x is referred to as the *excess noise index*. Considering instead the thermal noise, the variance σ_{t}^2 of $i_{\mathrm{t}}(t)$ can be expressed as [33]

$$\sigma_{\mathrm{t}}^2 = \frac{4kT}{R_{\mathrm{L}}} F_{\mathrm{n}} B \tag{2.10}$$

where R_{L} is the equivalent load resistance of the transimpedance amplifier, F_{n} is the amplifier noise figure, and T the temperature.

To determine the SNR of the APD–based receiver, it can be observed that the $i_{\mathrm{s}}(t)$ and $i_{\mathrm{t}}(t)$ are independent random processes, and hence their variances can be added to determine the total noise power. It results

$$\mathrm{SNR_{APD}} = \frac{I_{\mathrm{p}}^2}{\sigma_{\mathrm{s}}^2 + \sigma_{\mathrm{t}}^2} = \frac{M^2 \left(R P_{\mathrm{in}} \right)^2}{2qM^2 F_A \left(R P_{\mathrm{in}} + I_{\mathrm{d}} \right) B + 4 \left(kT/R_{\mathrm{L}} \right) F_{\mathrm{n}} B} \tag{2.11}$$
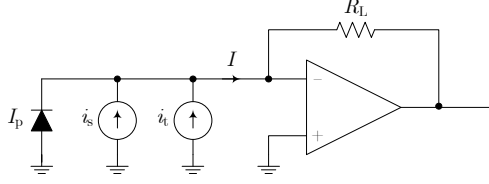
Figure 2.5: Basic scheme of an APD transimpedance front–end with equivalent noise sources

This equation shows the different ways in which shot and thermal noise affect the SNR. Indeed, if shot noise is greater than thermal noise, i.e. $\sigma_s^2 \gg \sigma_t^2$, the APD receiver, despite its multiplication factor M, is more noisy than a p–i–n photodiode, for which it is $M = 1$ and $F_A = 1$. Fortunately, in practical receiver implementations the opposite happens, i.e. thermal noise is much greater than shot noise ($\sigma_t^2 \gg \sigma_s^2$), so that (2.11) can be approximated by

$$\text{SNR}_{\text{APD}} = \frac{I_p^2}{\sigma_s^2 + \sigma_t^2} \approx \frac{M^2 \left(RP_{\text{in}}\right)^2}{4\left(kT/R_{\text{L}}\right) F_{\text{n}} B} \tag{2.12}$$

in which an improvement of a factor M^2 is achieved with respect to the SNR of p–i–n receivers.

Finally, it can be shown that, once P_{in} is given, SNR_{APD} has a maximum for an optimum value M_{opt} of the APD multiplication factor M. This value is given by the solution of the following equation [33]

$$k_A M_{\text{opt}}^3 + (1 - k_A)M_{\text{opt}} = \frac{4kTF_{\text{n}}}{qR_{\text{L}}\left(RP_{\text{in}} + I_{\text{d}}\right)} \tag{2.13}$$

in which the second term can be neglected whenever an APD with k_A in the range 0.7–1 is selected, so as to obtain

$$M_{\text{opt}} \approx \left[\frac{4kTF_{\text{n}}}{k_A\, qR_{\text{L}}\left(RP_{\text{in}} + I_{\text{d}}\right)}\right]^{1/3} \tag{2.14}$$

Usually, this last simplification is allowed for InGaAs APDs, for which it is $k_A \approx 0.7$. In particular, InGaAs APDs also have the proper band–gap which allows to receive wavelengths within the 1.3–1.6 μm range. For this reason, they have been selected for the development of the proposed DTS prototype [37].

By comparing (2.5) with (2.14) it can be noted that while M depends on the electric field, and hence on the APD reverse bias voltage, M_{opt} depends also on the receiver temperature T and on P_{in}. As a consequence, for a given P_{in}, it is important to control both the bias voltage and the temperature of the receiver to let M be as close as possible to M_{opt}. For this reason, a fine tunable APD biasing circuit has been implemented, along with a thermostat based on a Peltier cell and on an integrated temperature sensor (see Figure 2.2. In particular, the bias circuit and the temperature sensor are controlled by the front–end µC, whereas the Peltier cell is supplied by the dedicated *Peltier driver* block, in which another µC allows for the selection of both the amplitude and the direction of the cell current.

Equation (2.14) shows that M_{opt} depends on the reverse bias voltage (by means of k_A), on the optical input power P_{in}, and on the amplifier temperature. On the other hand, (2.5) shows that M depends on the reverse bias voltage as well. However, it also depends on d, which is a function of the APD breakdown voltage and hence of the APD temperature. As a consequence, for a given P_{in}, it is important to control both the bias voltage and the temperature of the APD to let M be as close as possible to M_{opt}. For this reason, a fine tunable APD biasing circuit has been implemented, along with a thermostat based on a Peltier cell and on an integrated temperature sensor (see Figure 2.2). In particular, the bias circuit and the temperature sensor are controlled by the front–end µC, whereas the Peltier cell is supplied by the dedicated *Peltier driver* block, in which another µC allows for the selection of both the amplitude and the direction of the cell current.

Figure 2.6 shows the result of a bench characterization for one of the selected APD. In the temperature–bias voltage plane, it reports the locus of points for which M is at its nominal maximum value, i.e. $M = 70$ [37]. It can be observed that voltages up to 85 V are required to cover the typical 0–70 °C operating range of commercial measuring instruments.

2.3.3 Amplification chain

To amplify the APD photocurrent, a two–stage amplification chain has been adopted. This chain consists of a fixed–gain Transimpedance Amplifier (TIA), and of a Variable–Gain voltage Amplifier (VGA) (see Figure 2.2). The total
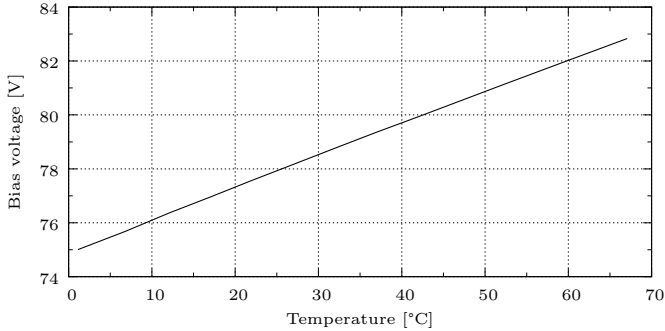
Figure 2.6: APD bias voltage vs. temperature at $M = 70$ (NEC NR8800)

gain G_T provided by the chain should allow the signal coming from the photo-diode to be normalized within the input range of the following A/D conversion block. The minimum G_T required can therefore be estimated with

$$G_T > \frac{V_{IS}}{I_p} = \frac{V_{IS}}{RMP_{in}} \tag{2.15}$$

where V_{IS} is the peak–to–peak input span of the A/D stage. Considering the typical APD parameters $R = 0.9\,\mathrm{A/W}$, $M = 70$, an expected maximum backscattered power $P_{in} = 10\,\mathrm{nW}$, and an input span $V_{IS} = 2\,\mathrm{V}$, it follows $G_T > 3.2 \times 10^6\,\Omega = 130\,\mathrm{dB}\,\Omega$.

The VGA can be entirely configured by the front–end µC, and hence by the PC. This way, it is possible to develop specific software algorithms which implement automatic gain control functions. This gives the system a good degree of flexibility, as it allows to rapidly configure the whole front–end according to the VOA and APD bias settings.

In addition, the VGA also receives a *blank* signal from the A/D stage. This signal is activated together with the laser trigger and keeps the VGA input disabled for all its duration. Indeed, it is possible that the typical 60 dB isolation provided by the Raman filter [38] between the laser port and the Stokes or anti–Stokes port is not enough to prevent even a small fraction of the pulse from reaching the APDs. If this happens, the residual pulse may dazzle the receiver and bring it into saturation. If eventually the receiver recovery time is too long (e.g. 1 µs), the first part of the backscattered traces result corrupted, and hence no information from the first part of the fibre (e.g.

100 m) can be retrieved. With the help of the blank signal, the VGA stage does not saturate, and its normal operation can be restored as soon as the signal turns off. This allow to reduce the receiver recovery time, and therefore to shorten the initial fibre dead zone. In particular, the blank duration can be fine tuned by software to match the pulse one.

As far as the bandwidth of the amplification chain is concerned, it depends on the spatial resolution σ_{S} to be achieved (see Section 1.5). To estimate its value, some simplifying assumptions can be made. First, we may neglect the inherent low–pass effect of both the OTDR and the APD, and assume the chain as a first–order system under the dominant–pole approximation. Then, we may consider the effect of a real temperature step on the backscattered traces instead of on the reconstructed profile. Hence, it follows that B has to satisfy the requirement

$$B > \frac{1}{2\pi\tau} = \frac{\ln 9}{2\pi t_{\mathrm{r}}} = \frac{v\ln 9}{2\pi\sigma_{\mathrm{S}}} \tag{2.16}$$

where τ is the time constant of the system, and $t_{\mathrm{r}} = \tau\ln 9$ is the required 10–90 % step–response rise time, given by the ratio between σ_{S} and the speed v of the Raman backscattered light in the fibre. In particular, for the worst–case estimation, the Stokes speed should be taken into account, as it is greater than the anti–Stokes one. However, it can be safely approximated to $v = 2 \times 10^8$ m/s. Thus, to achieve a metre–scale spatial resolution, i.e. $\sigma_{\mathrm{S}} = 1$ m, it results $B > 70$ MHz. It is worth noting that if the previous assumptions on the OTDR and the APD are not verified, the actual spatial resolution is determined by the subsystem with the narrowest bandwidth.

In general, if on the one hand a wide bandwidth allows for a good spatial resolution, on the other hand it increases the amplifier noise. As a consequence, B should be chosen not too larger than the value obtained from (2.16), so that the equivalent input rms noise current of the chain is kept lower than the minimum photocurrent. If we assume an average fibre attenuation of 0.2 dB/km and a backscattered power at the fibre input $P_{\mathrm{in}}(0) = 10$ nW, the power received from the distance $L = 30$ km is about -12 dB lower, i.e. $P_{\mathrm{in}}(L) = 0.6$ nW. With $R = 0.9$ A/W and $M = 70$, this power leads to a photocurrent $I_{\mathrm{p}}(L) = 40$ nA, which represent the upper limit of the chain rms noise current.

2.4 Digital conversion and acquisition

To convert and acquire the backscattered traces, an ADC/FPGA–based platform has been adopted. This approach allows to significantly reduce the overall development times and cost, and gives the system a great degree of flexibility.

In fact, as shown in Figure 2.2, data coming from the ADC are acquired by the FPGA, which handles the whole acquisition process by driving both the trigger and the blank signals. Thanks to the FPGA reconfigurability, it is possible to develop specific architectures which allow to easily test and debug the basic system functionalities, as well as to implement advanced algorithms, such as the interleaved sampling (see Section 3.4.1) or the pulse coding (see Chapter 6), which may remarkably increase the system performance.

The FPGA is configured through the block μC, which receives the binary configuration file from the PC via USB. The same link is used to set the ADC parameters, and of course to download the acquired traces.

2.4.1 Analog to digital converter

From an architectural point of view, a dual–channel ADC is needed to acquire Stokes and anti–Stokes traces simultaneously. The choice of a specific converter can be carried out by determining the required number n of resolution bits and the sampling frequency f_s.

The choice of n determines the quantization noise of the converter, as well as its SNR, which is given by the well–known relation [39]

$$\mathrm{SNR} = 6.02n\ \mathrm{dB} + 1.76\,\mathrm{dB} \tag{2.17}$$

when the full Nyquist bandwidth $f_\mathrm{s}/2$ is considered. To keep the quantization noise lower than the noise already affecting the signal, n should be chosen so that the SNR of the ADC is greater than the output SNR of the front–end. However, as (2.17) refers to an ideal ADC and does not include the noise due to distortions or to non–linearity effects, the following relation can be more appropriately taken into account

$$\mathrm{SINAD} = 6.02 \cdot \mathrm{ENOB}\ \mathrm{dB} + 1.76\,\mathrm{dB} \tag{2.18}$$

in which SINAD is the signal to noise–and–distortion ratio, and ENOB is the effective number of bits of the particular converter to be evaluated. Once ENOB is determined, an ADC with $n > $ ENOB can be safely selected. Preliminary bench testing of both the APD and the amplification chain showed that an ADC with $n > 10$ should well meet the noise requirements. Finally, the choice of f_S can be carried out by taking into account the receiver bandwidth given by (2.16).

2.4.2 Data acquisition and averaging

The main requirement for the FPGA is represented by the amount of available memory resources to store the acquired traces. The basic idea is to let the FPGA accumulate several consecutive Stokes and anti–Stokes traces into two different sum–traces, and transfer these accumulated data to the PC for averaging and further processing.

In general, the acquisition of a single trace requires a memory space of $n \times S$ bits, being n the number of bits per sample, and S the number of samples per trace. The value of S is given by

$$S = \left\lceil L \left(\frac{1}{v_\mathrm{P}} + \frac{1}{v_\mathrm{AS}} \right) f_\mathrm{S} \right\rceil \qquad (2.19)$$

in which L is the fibre length, v_P and v_AS are the speed of pulse and anti–Stokes wavelengths, respectively, and f_S is the sampling frequency. Here, anti–Stokes speed is considered because it is lower than Stokes one. This way, the result provided by (2.19) represents the worst–case, and it can be safely used also for Stokes traces. However, if the available memory resources do not allow to reach a given measurement range, a decimation process can be implemented, so that S can be divided by the decimation factor. The main drawback of this approach is the worsening of the effective sampling resolution and, likely, of the spatial resolution.

In addition, if several traces have to be accumulated, more than n bits per sample need to be reserved to avoid saturation or overflows. If indeed w is the actual memory width, i.e. the actual number of memory bits reserved for each n–bit sample, the FPGA can safely accumulate up to 2^{w-n} traces without any numerical representation error.

As the number of the accumulated traces increases, the SNR of the averaged data improves. This allows to reduce the error introduced on the temperature assessment, and therefore to increase the achieved temperature resolution. In particular, once the SNR of raw traces is given, it is interesting to evaluate the number N of the averages required to obtain a specific temperature resolution σ_T. To this end, the relationship between the traces SNR and σ_T has to be determined.

A detailed analysis leading to the formal definition of σ_T as a function of $P_\text{S}(z)$ and $P_\text{AS}(z)$ variances can be found in [17]. Here, a simplified but easy–to–use derivation of σ_T as a function of the anti–Stokes SNR detected at the receiver output is instead proposed. Despite several approximations, the obtained result is very close to real data, as confirmed by the experimental tests reported in Chapter 5. Starting from (1.18), the temperature profile as a function of z is given by

$$T(z) = \frac{h\Delta\nu/k}{\ln\left[\kappa'(z)\dfrac{P_\text{S}(z)}{P_\text{AS}(z)}\right]} \tag{2.20}$$

in which $\kappa'(z) = \kappa \cdot \exp\left[(\alpha_\text{S} - \alpha_\text{AS})z\right]$. To take the noise into account, the backscattered powers can be written as

$$P_\text{S}(z) = P_\text{S0}(z) + p_\text{Sn}(z) \tag{2.21}$$
$$P_\text{AS}(z) = P_\text{AS0}(z) + p_\text{ASn}(z) \tag{2.22}$$

where $p_\text{Sn}(z)$ and $p_\text{ASn}(z)$ represent the uncorrelated Gaussian noise terms affecting $P_\text{S0}(z)$ and $P_\text{AS0}(z)$, respectively. By substituting these relations in the previous equation, it follows

$$T(z) = \frac{h\Delta\nu/k}{\ln\left[\kappa'(z)\dfrac{P_\text{S0}(z)}{P_\text{AS0}(z)}\right] + \ln\left[\dfrac{1 + p_\text{Sn}(z)/P_\text{S0}(z)}{1 + p_\text{ASn}(z)/P_\text{AS0}(z)}\right]} \tag{2.23}$$

Here, the noiseless part $T_0(z)$ of $T(z)$ is identified by the first denominator term, that is

$$T_0(z) = \frac{h\Delta\nu/k}{\ln\left[\kappa'(z)\dfrac{P_\text{S0}(z)}{P_\text{AS0}(z)}\right]} \tag{2.24}$$

whereas the second denominator term is related to noise. Focusing on its logarithm, and assuming $P_\text{AS0} \gg p_\text{ASn}$ and $P_\text{S0} \gg p_\text{Sn}$, the approximation

$(1+x)^{-1} \approx 1 - x$ for $x \to 0$ can be applied to the argument, so as to obtain

$$\ln\left\{\left[1 + \frac{p_{\mathrm{Sn}}(z)}{P_{\mathrm{S0}}(z)}\right]\left[1 - \frac{p_{\mathrm{ASn}}(z)}{P_{\mathrm{AS0}}(z)}\right]\right\}$$

which can also be simplified by neglecting the second order cross products to

$$\ln\left\{1 + \frac{p_{\mathrm{Sn}}(z)}{P_{\mathrm{S0}}(z)} - \frac{p_{\mathrm{ASn}}(z)}{P_{\mathrm{AS0}}(z)}\right\}$$

Finally, by applying the further approximation $\ln(1+x) \approx x$ for $x \to 0$, and substituting the obtained result in (2.23) together with (2.24), it remains

$$
\begin{aligned}
T(z) &\approx \frac{1}{\dfrac{1}{T_0(z)} + \dfrac{k}{h\Delta\nu}\left[\dfrac{p_{\mathrm{Sn}}(z)}{P_{\mathrm{S0}}(z)} - \dfrac{p_{\mathrm{ASn}}(z)}{P_{\mathrm{AS0}}(z)}\right]} \\[2ex]
&\approx \frac{T_0(z)}{1 + \dfrac{T_0(z)}{T_{\mathrm{B}}}\left[\dfrac{p_{\mathrm{Sn}}(z)}{P_{\mathrm{S0}}(z)} - \dfrac{p_{\mathrm{ASn}}(z)}{P_{\mathrm{AS0}}(z)}\right]}
\end{aligned}
\tag{2.25}
$$

in which $T_{\mathrm{B}} = h\Delta\nu/k = 435\,\mathrm{K}$ has been defined. Again, the approximation $(1+x)^{-1} \approx 1 - x$ for $x \to 0$ can be applied to produce

$$T(z) \approx T_0(z)\left\{1 - \frac{T_0(z)}{T_{\mathrm{B}}}\left[\frac{p_{\mathrm{Sn}}(z)}{P_{\mathrm{S0}}(z)} - \frac{p_{\mathrm{ASn}}(z)}{P_{\mathrm{AS0}}(z)}\right]\right\} \tag{2.26}$$

which finally gives

$$T(z) - T_0(z) \approx -\frac{T_0^2(z)}{T_{\mathrm{B}}}\left[\frac{p_{\mathrm{Sn}}(z)}{P_{\mathrm{S0}}(z)} - \frac{p_{\mathrm{ASn}}(z)}{P_{\mathrm{AS0}}(z)}\right] \tag{2.27}$$

The left term of this last equation represent the temperature noise. Therefore, if we call with σ_{n} the standard deviation of p_{Sn} and p_{ASn}, it is possible to evaluate σ_{T} as follows

$$\sigma_{\mathrm{T}}^2(z) = \left[\frac{T_0^2(z)}{T_{\mathrm{B}}}\right]^2\left[\frac{1}{P_{\mathrm{S0}}^2(z)} + \frac{1}{P_{\mathrm{AS0}}^2(z)}\right] \cdot \sigma_{\mathrm{n}}^2 \tag{2.28}$$

and since in practice it is $P_{\mathrm{S0}}^2 \gg P_{\mathrm{AS0}}^2$, it results

$$\sigma_{\mathrm{T}}^2(z) \approx \left[\frac{T_0^2(z)}{T_{\mathrm{B}}}\right]^2 \cdot \frac{\sigma_{\mathrm{n}}^2}{P_{\mathrm{AS0}}^2(z)} \tag{2.29}$$

By defining the anti–Stokes signal–to–noise ratio $\mathrm{SNR_{AS}}$ as

$$\mathrm{SNR_{AS}}(z) = \frac{P_{\mathrm{AS0}}^2(z)}{\sigma_{\mathrm{n}}^2} = \frac{P_{\mathrm{AS0}}^2(0)}{\sigma_{\mathrm{n}}^2} \exp\left[-2(\alpha_{\mathrm{P}} + \alpha_{\mathrm{AS}})z\right] \qquad (2.30)$$

equation (2.29) turns into

$$\sigma_{\mathrm{T}}(z) = \frac{T_0^2(z)}{T_{\mathrm{B}}} \cdot \frac{\exp\left[(\alpha_{\mathrm{P}} + \alpha_{\mathrm{AS}})z\right]}{\sqrt{\mathrm{SNR_{AS}}(0)}} \qquad (2.31)$$

which definitely represents the temperature resolution at a given z.

Observing equation (2.31), two important aspects of σ_{T} can be pointed out. The first is related to its dependency on z. Indeed, as z increases, backscattered power decays exponentially, and the same happens to the SNR. As a consequence, σ_{T} get worse with distance, as confirmed by the positive exponential factor. The second aspect is instead related to the dependency on T_0. The higher the temperature, the higher the backscattered power, the better the SNR. However, the numerator term T_0^2 prevails over the SNR improvement, so that σ_{T} shows an almost linear worsening with the temperature, as confirmed in [17].

When N traces are averaged, the SNR improves of a factor N. Including this effect in the previous equation gives

$$\sigma_{\mathrm{T}}(z) = \frac{T_0^2(z)}{T_{\mathrm{B}}} \cdot \frac{\exp\left[(\alpha_{\mathrm{P}} + \alpha_{\mathrm{AS}})z\right]}{\sqrt{N \cdot \mathrm{SNR_{AS}}(0)}} \qquad (2.32)$$

and solving for N

$$N = \left[\frac{T_0^2(z)}{T_{\mathrm{B}}}\right]^2 \cdot \frac{\exp\left[2(\alpha_{\mathrm{P}} + \alpha_{\mathrm{AS}})z\right]}{\sigma_{\mathrm{T}}^2(z) \cdot \mathrm{SNR_{AS}}(0)} \qquad (2.33)$$

By applying this equation to the results collected in preliminary bench tests, it followed that to let $\sigma_{\mathrm{T}}(z = 30\,\mathrm{km}) = 1\,\mathrm{K}$ at room temperature, about $N = 2 \times 10^6$ averages are required. In practice, such a large number of traces cannot be entirely accumulated within the FPGA, as they would require an amount of memory resources which is typically unavailable even in high–end devices. As a consequence, the approach adopted is to accumulate as much traces as possible in the FPGA, transfer them to the PC, and repeat the process until N is reached. The PC in its turn further accumulates sum–traces coming from the FPGA, and when all the N traces are accumulated, it computes the average and extracts the temperature profile.

3

Hardware Implementation

3.1 APD biasing circuit

According to the remarks reported in Sections 2.3.1 and 2.3.2, a fine tunable APD biasing circuit with a wide output voltage range has been designed. The circuit is based on the Maxim MAX1932 device, which is considered the state–of–the–art for this kind of applications. Figure 3.1 shows the adopted schematic diagram, which slightly differs from the one proposed by the manufacturer [40] because of some modifications that have been implemented to extend the operating range.

Basically, the MAX1932 is a constant–frequency PWM step–up converter.
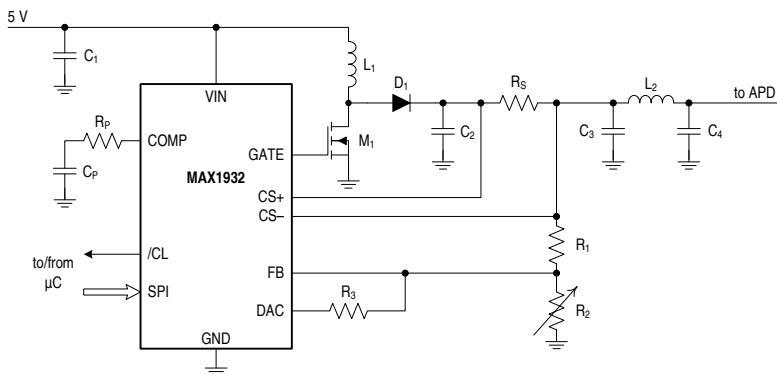
Figure 3.1: The implemented APD biasing circuit based on the MAX1932 device

It features a low output ripple ($< 1\,\mathrm{mV}$) and a current sensing/limiting architecture which allows to protect the APD if the avalanche breakdown occurs. It also provides an internal 8 bit SPI-controlled DAC, which can be used to digitally regulate the APD bias voltage.

The circuit is powered by a regulated $5\,\mathrm{V}$ supply. The PWM stage is implemented by the diode D_1, the inductor L_1 and the N–MOSFET M_1, whose gate is directly driven by the internal control logic. The steady–state output voltage V_{APD} can be determined by analysing the feedback network, in which the node FB can be considered as virtually shorted to the internal band–gap reference $V_{\mathrm{REF}} = 1.25\,\mathrm{V}$. It results

$$V_{\mathrm{APD}} = V_{\mathrm{FB}} + R_1 \left[\frac{V_{\mathrm{FB}}}{R_2} + \frac{V_{\mathrm{FB}} - V_{\mathrm{DAC}}}{R_3} \right] \tag{3.1}$$

in which the output DAC voltage V_{DAC} is given by

$$V_{\mathrm{DAC}} = \frac{\mu + 1}{256} V_{\mathrm{REF}} \tag{3.2}$$

The number $\mu = 0 \ldots 255$ is the value of the 8 bit codeword loaded in the DAC control register via SPI. In particular, for $\mu = 0$ the DAC output is disabled, and the step–up converter is actually shut down. By substituting (3.2) in (3.1) and considering $V_{\mathrm{FB}} = V_{\mathrm{REF}}$ it follows

$$V_{\mathrm{APD}} = V_{\mathrm{REF}} \left[1 + \frac{R_1}{R_3} \frac{\mu}{256} + \frac{R_1}{R_2} \right] \tag{3.3}$$

From this last equation, it can be observed that once R_2 is given, the values of R_1 and R_3 determine both the regulation step and the operating range width. This means that it is not possible to obtain a wide range with a small step if R_2 is fixed. As a consequence, to give the system a greater flexibility without affecting the regulation accuracy, R_2 has been implemented with the digitally controlled network reported in Figure 3.2. From the diagram it follows that the value of $R_2 = 1/G_2$ is given by

$$G_2 = \frac{1}{R_2} = G_{20} + \rho \Delta G_2 \tag{3.4}$$

where $\rho = 0 \ldots 63$ is the value of the binary codeword representing the status of the six digitally controlled switches, which have been implemented with the
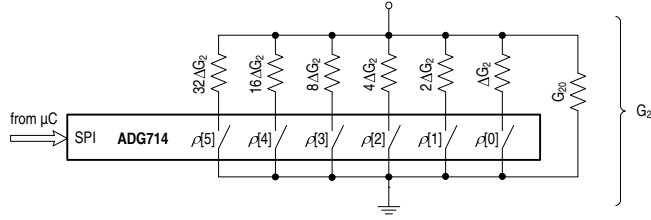
Figure 3.2: The R_2 regulation network based on the ADG714 device

Analog Devices ADG714 [41]. This device features an SPI interface, and it is controlled by the front–end µC along with the MAX1932. In particular, if we set $R_1 \Delta G_2 = 1$ and $R_1 = R_3$, when (3.4) is substituted in (3.3) it remains

$$V_{\text{APD}} = V_{\text{REF}} \left[1 + \frac{\mu}{256} + R_1 G_{20} + \rho \right] \tag{3.5}$$

This way, by varying ρ it is possible to scan a wide output range without affecting the fine regulation step controlled by μ. In particular, we set $R_1 G_{20} = 16$ so that the minimum ($\mu = 1$, $\rho = 0$) and maximum ($\mu = 255$, $\rho = 63$) output voltage result

$$V_{\text{APD}}^{\text{min}} = V_{\text{REF}} \left[1 + \frac{1}{256} + 16 \right] = 21.25 \, \text{V}$$

$$V_{\text{APD}}^{\text{max}} = V_{\text{REF}} \left[1 + \frac{255}{256} + 16 + 63 \right] = 102.5 \, \text{V}$$

whereas the regulation step is kept at $V_{\text{APD}}^{\text{step}} = V_{\text{REF}}/256 = 4.88 \, \text{mV}$. These values not only allow to fine tune the bias point of the adopted APDs and precisely track the optimum M, but also make the circuit well suited for a wide range of different photodetectors which may be tested in the future. The whole circuit can hence be controlled by a single 14 bit codeword given by the concatenation $\{\rho[5:0], \mu[7:0]\}$. However, it is important to point out that all the codewords with $\mu = 0$ are actually unavailable due to the MAX1932 shut down. This exception is handled via firmware, and all the invalid codewords – except the only all–zeros one – are masked.

From the implementation point of view, both R_1, R_3 and the R_2 network have been realized using integrated resistors arrays. For each array, single resistors have been properly connected in series or in parallel to achieve the values required by design. This way, a good precision and an high immunity

against possible operating temperature variations is obtained. Indeed, state–of–the–art resistor arrays available on the market feature absolute tolerances in the order of 0.1%, relative tolerances within resistors of the same array of just 0.05%, and a thermal drift down to $25\,\mathrm{ppm/^\circ C}$. In particular, we set $R_1 = 400\,\mathrm{k\Omega}$ to assure that each resistance connected to the ADG714 results so greater than the parasitic $4.5\,\Omega$ switch resistance, that it can be safely neglected.

The limitation of the bias current has been set according to the maximum $1\,\mathrm{mA}$ safety threshold declared by the APD manufacturer [37]. To this end, as the MAX1932 internal limiter turns on when a $2\,\mathrm{V}$ voltage drop across the sensing resistor R_S is detected, we accordingly set $R_\mathrm{S} = 2\,\mathrm{k\Omega}$. In particular, the limiter working status is signalled to the front–end µC by means of the /CL flag. This allows for the firmware/software implementation of advanced biasing algorithms which guarantee the safe operating conditions for APDs.

As reported in Section 2.3.1, these algorithms may involve the temperature regulation of the photodiode. For this reason, a Texas Instruments TMP275 digital I2C temperature sensor [42] has been included on the front–end board (see Figure 2.2), placed very close to the APD, and connected to the µC. This sensor features a temperature accuracy of $\pm 0.065\,^\circ\mathrm{C}$ within the -20–$100\,^\circ\mathrm{C}$ working range. Hence, it is well suited for this application, and it has been mounted also on other system blocks whenever the board temperature information was required.

Finally, all the remaining components of the biasing circuit, such as M_1, D_1, L_1, the R_P–C_P compensation network, and the C_3–L_2–C_4 output filter have been sized according to the guidelines provided by the manufacturer [40].

3.2 Analog front–end

3.2.1 Block diagram and gain analysis

As reported in Section 2.1, the front–end chain is made of a transimpedance amplification stage (TIA), followed by a variable voltage amplification stage (VGA). The related block diagram is shown in Figure 3.3.
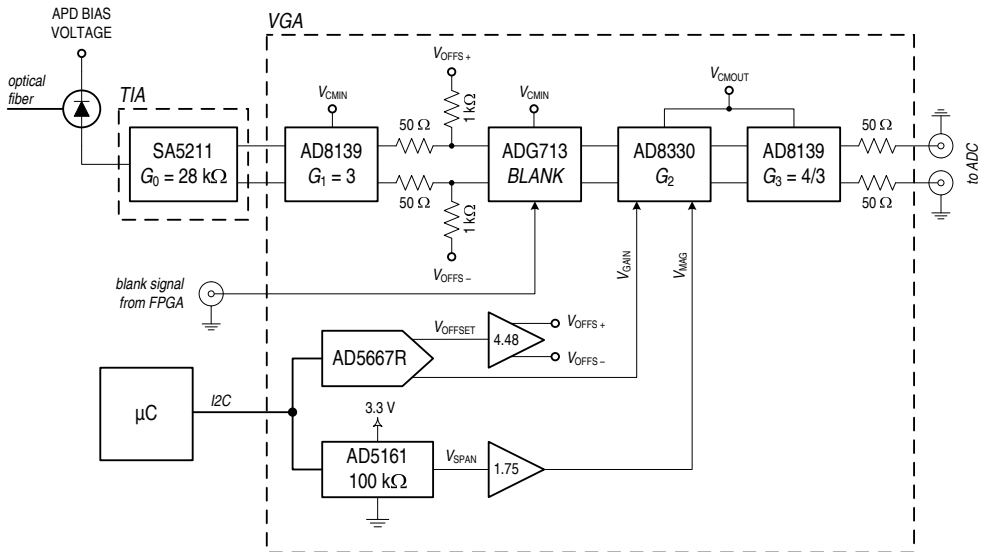
Figure 3.3: The analog front–end block diagram

The transimpedance stage is implemented with a stand–alone device, the SA5211 [43], which is specifically designed for the pre–amplification of photodiode current signals. It is a single–ended to differential amplifier which provides a gain $G_0 = 28\,\text{k}\Omega$, a nominal bandwidth of 180 MHz, and an equivalent input noise current of $1.8\,\text{nA}/\sqrt{\text{Hz}}$. Its main feature is an high immunity of noise parameters from the capacitance of the connected APD. This happens because the Miller effect arising across its internal gain stages makes the input capacitance $C_{\text{TIA}}^{\text{in}} = 4\,\text{pF}$ dominate over the APD one, which in our case is $C_{\text{APD}} = 0.5\,\text{pF}$. Under these conditions, the actual bandwidth can be evaluated using the model provided by the manufacturer [43], thus obtaining

$$B_0 = \frac{1}{2\pi R_{\text{TIA}}^{\text{in}} \left(C_{\text{TIA}}^{\text{in}} + C_{\text{APD}} \right)} = 177\,\text{MHz} \tag{3.6}$$

in which $R_{\text{TIA}}^{\text{in}} = 200\,\Omega$. With this result, it follows that the rms input noise current is about $24\,\mu\text{A}$, which is enough lower than the $40\,\mu\text{A}$ threshold determined in Section 2.3.3.

The VGA stage is instead implemented by three fully–differential substages. The first one is based on the AD8139 amplifier [44], and provides a gain $G_1 = 3$. By means of a dedicated input pin, the output common mode voltage is set

to $V_{\mathrm{CMIN}} = 2.37\,\mathrm{V}$. This value matches the output common mode of the TIA stage, and it is derived from a simple resistive voltage divider not shown in the diagram.

An additional voltage divider is instead used along the signal path to regulate the output offset of the first stage . This is achieved by adding to the signal the two symmetrical voltages $V_{\mathrm{OFFS+}}$ and $V_{\mathrm{OFFS-}}$, which are obtained from the dual–channel 16 bit AD5667R DAC [45]. This device features an integrated $1.25\,\mathrm{V}$ reference and two independent output channels. As shown in Figure 3.3, only the first one is used for the offset regulation, thanks to a simple op-amp circuit which provides the two symmetrical voltages. The DAC device is connected to the front–end µC through an I2C link, thus allowing the user to regulate the offset and keep the signal within the dynamic range of the chain once the total gain has been set.

The blank functionality (see Section 2.3.3) is implemented by means of the ADG713 [46] quad switch. When the blank signal coming from the FPGA is active, the switch configuration forces the blank stage outputs to the V_{CMIN} common mode voltage. When instead the blank signal is inactive, the block is transparent and the signal path remains unchanged.

The next block is the actual variable gain stage, and it is based on the stand–alone AD8330 [47] amplifier. This device provides very good noise parameters and shows a constant $150\,\mathrm{MHz}$ bandwidth which does not depend on the selected gain. As reported in Figure 3.3, it is controlled by means of three different signals, i.e. V_{CMOUT}, V_{GAIN} and V_{MAG}. The first is used to set the output common mode, and it is fixed at $2\,\mathrm{V}$ by a voltage divider not shown in the diagram. The other ones are instead used to set the gain, which is given by

$$G_2 = \frac{V_{\mathrm{MAG}}}{0.5\,\mathrm{V}} \cdot 10^{V_{\mathrm{GAIN}}/0.6\,\mathrm{V}} \qquad (3.7)$$

which holds for $V_{\mathrm{MAG}} \leq 5\,\mathrm{V}$ and $V_{\mathrm{GAIN}} \leq 1.5\,\mathrm{V}$. From this relation, it can be observed that the value of G_2 is proportional to V_{MAG} when a linear–scale is used, and to V_{GAIN} when a dB–scale is used. For this reason V_{MAG} and V_{GAIN} are called *linear–in–magnitude* and *linear–in–dB* gain controls, respectively. As shown in the diagram, V_{GAIN} is directly provided by the second channel of the AD5667R, whereas V_{MAG} is obtained through the $100\,\mathrm{k}\Omega$ digital potentiometer AD5161 [48]. In particular, this device is supplied by a 5–to–3.3 V linear regulator and then buffered by a simple op-amp. Despite the lower supply, it

is anyway able to safely receive the 0–5 V SPI control signals coming from the front–end µC.

Finally, the last stage, which is still based on the AD8139 amplifier, provides a gain $G_3 = 4/3$ and is used to drive the two 50 Ω lines towards the ADC board.

By recalling the contribution of each stage, the total gain G_T of the chain is hence given by

$$G_T = 28\,\text{k}\Omega \cdot 3 \cdot \frac{V_{\text{MAG}}}{0.5\,\text{V}} \cdot 10^{V_{\text{GAIN}}/0.6\,\text{V}} \cdot \frac{4}{3} \cdot \frac{1}{2} \tag{3.8}$$

where the last factor $1/2$ takes into account the inherent attenuation of the matched 50 Ω transmission line at the ADC board input. By simplifying (3.8) and substituting $V_{\text{MAG}} = 1.75 V_{\text{SPAN}}$ it follows

$$G_T = 19.6 \cdot \frac{V_{\text{MAG}}}{1\,\text{V}} \cdot 10^{V_{\text{GAIN}}/1\,\text{V}} \text{M}\Omega \tag{3.9}$$

which holds for $V_{\text{SPAN}} \leq 2.857\,\text{V}$ and $V_{\text{GAIN}} \leq 1.5\,\text{V}$. This means that the maximum G_T results

$$G_T^{\text{max}} = 1.77 \times 10^9\,\Omega = 185\,\text{dB}\,\Omega \tag{3.10}$$

which meets the requirements determined in Section 2.3.3.

3.2.2   Bandwidth analysis

The SA5211 and AD8330 amplifiers are stand–alone devices. This means that, besides some bypass capacitors on their supply pins, no additional external components are required to let them work properly. If on one hand this allows to significantly simplify the design and layout of the front–end PCB, on the other hand it makes the tuning of some parameters, and in particular of their bandwidth, a quite difficult task.

As shown in the previous section, the SA5211 and AD8330 devices provide a fixed bandwidth of 178 MHz and 150 MHz, respectively. Hence, to reduce the overall bandwidth of the chain, it is necessary to act on the AD8139–based stages, and in particular on the first one, so as to also reduce the noise propagating through the next amplification blocks.
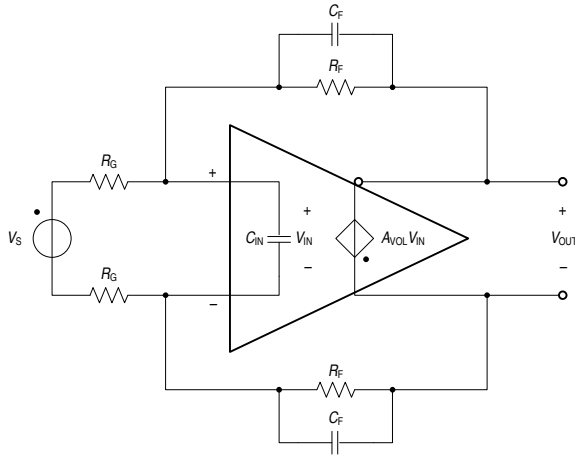
Figure 3.4: The first amplification stage based on the AD8139 (the control pin for the output common mode pin is omitted)

The schematic diagram of the first AD8139 stage is shown in Figure 3.4. For simplicity's sake, the control pin for the output common mode has been omitted. The circuit implements a differential first–order low–pass filter, with a DC gain and a bandwidth given by

$$G_1 = \frac{R_{\mathrm{F}}}{R_{\mathrm{G}}} \tag{3.11}$$

$$B_1 = \frac{1}{2\pi R_{\mathrm{F}} C_{\mathrm{F}}} \tag{3.12}$$

Following the remarks reported in Section 2.3.3, we set $B_1 = 100\,\mathrm{MHz}$ as a good trade–off between the minimum required bandwidth and the receiver noise. To obtain this value, along with $G_1 = 4$, and assure a good closed–loop stability, R_{F}, C_{F} and R_{G} have been determined by analysing the AD8139 feedback network, including the effect of the input capacitance C_{IN}.

By applying the cut–insertion theorem [49] at the amplifier input, the circuit of Figure 3.4 turns into the one reported in Figure 3.5, with $Z_{\mathrm{P}} = 1/sC_{\mathrm{IN}}$. For this new circuit, the following transfer functions can be defined

$$\rho = \left.\frac{I_{\mathrm{P}}}{V_{\mathrm{S}}}\right|_{V_{\mathrm{P}}=0} = 0 \tag{3.13}$$
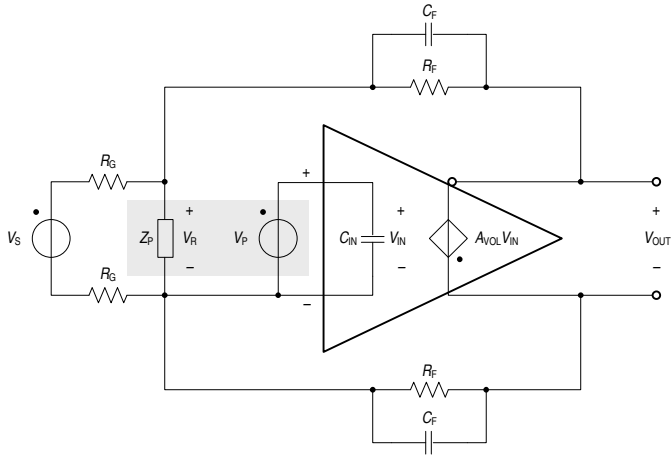
Figure 3.5: Application of the cut–insertion theorem to the AD8139 stage

$$\gamma = \left.\frac{V_{\text{OUT}}}{V_{\text{S}}}\right|_{V_{\text{P}}=0} = 0 \tag{3.14}$$

$$\alpha = \left.\frac{V_{\text{R}}}{V_{\text{S}}}\right|_{V_{\text{P}}=0} = \frac{\alpha_0}{1 + \dfrac{s}{\omega_{\text{P}\alpha}}} \tag{3.15}$$

$$A = \left.\frac{V_{\text{OUT}}}{V_{\text{P}}}\right|_{V_{\text{S}}=0} = -A_{\text{VOL}} = -\frac{A_{\text{VOL0}}}{\left(1 + \dfrac{s}{\omega_{\text{P1}}}\right)\left(1 + \dfrac{s}{\omega_{\text{P2}}}\right)} \tag{3.16}$$

$$\beta = \left.\frac{V_{\text{R}}}{V_{\text{OUT}}}\right|_{V_{\text{S}}=0} = \beta_0 \frac{1 + \dfrac{s}{\omega_{\text{Z}\beta}}}{1 + \dfrac{s}{\omega_{\text{P}\beta}}} \tag{3.17}$$

where $\alpha_0 = \dfrac{R_{\text{F}}}{R_{\text{G}} + R_{\text{F}}}$, $\beta_0 = \dfrac{R_{\text{G}}}{R_{\text{G}} + R_{\text{F}}}$, $\omega_{\text{P}\alpha} = \omega_{\text{P}\beta} = \dfrac{1}{(2C_{\text{IN}} + C_{\text{F}})\dfrac{R_{\text{G}}R_{\text{F}}}{R_{\text{G}} + R_{\text{F}}}}$,

$\omega_{\text{Z}\beta} = \dfrac{1}{R_{\text{F}}C_{\text{F}}}$, $A_{\text{VOL0}} = 1.122 \times 10^6$, $\omega_{\text{P1}} = 3.079\,\text{krad/s}$, $\omega_{\text{P2}} = 3.299\,\text{Grad/s}$.

In particular, the values of A_{VOL0}, ω_{P1} and ω_{P2} have been derived from the AD8139 Spice model provided by the manufacturer.

It is worth noting that both α and β have only a single pole because of the presence of a capacitor–only degenerate circuit in the network, and of the ideal matching between R_{F}, C_{F} and R_{G} in the upper and lower branches. The amplifier loop–gain βA is hence given by

$$\beta A = \beta_0 A_{\mathrm{VOL0}} \, \frac{1 + \dfrac{s}{\omega_{\mathrm{Z}\beta}}}{\left(1 + \dfrac{s}{\omega_{\mathrm{P1}}}\right)\left(1 + \dfrac{s}{\omega_{\mathrm{P2}}}\right)\left(1 + \dfrac{s}{\omega_{\mathrm{P}\beta}}\right)} \tag{3.18}$$

As βA has three poles, the closed–loop system may result unstable. To guarantee stability, two different approaches may be adopted. The first is the zero–pole cancellation, which is obtained by forcing $\omega_{\mathrm{Z}\beta} = \omega_{\mathrm{P}\beta}$. This can be achieved if R_{F}, C_{F} and R_{G} are fixed at precise values and perfectly matched each other. Unfortunately, it is not easy to obtain this condition in practice, and hence this method, even if theoretically applicable, results unfeasible.

The second approach relies on choosing proper component values so that not only the gain and bandwidth requirements are met, but also a good phase margin, and hence stability, is guaranteed. This task is usually carried out with the help of CAD tools, restricting the scope of investigation to those set of commercially available values which satisfy (3.12). In particular, resistance values should be chosen neither so large as to require the use of small and practically unfeasible feedback capacitors, nor so small as to excessively load the SA5211 output stage. A good trade–off is represented by resistances of a few hundred ohms, and capacitances of some picofarads. If $R_{\mathrm{F}} = 604\,\Omega$, $R_{\mathrm{G}} = 200\,\Omega$ and $C_{\mathrm{F}} = 2.7\,\mathrm{pF}$ are chosen, the Bode plot of βA obtained by means of numerical simulations is the one reported in Figure 3.6. It can be noted that with this set of values, the open–loop gain shows a phase margin $\varphi_{\mathrm{M}} = 87.7°$ which guarantees the stability. Hence this set was adopted.

Once βA has been determined, the closed–loop response of the AD8139, and hence the response of the whole chain, have been evaluated. Unfortunately, the SA5211 Spice model was unavailable, so that the TIA stage could not be included in the numerical analysis. This is not actually a problem, as the SA5211 provides a bandwidth of about $178\,\mathrm{MHz}$, which is wider than the one provided by the VGA block, and hence it does not affect the dynamic behaviour of the system. As a consequence, focusing on the VGA block only,
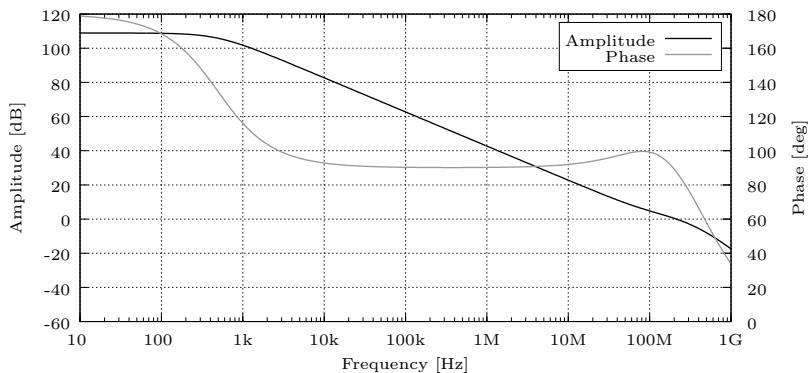
Figure 3.6: Bode plot of the AD8139 open–loop gain ($R_F = 604\,\Omega$, $R_G = 200\,\Omega$, $C_F = 2.7\,\text{pF}$)
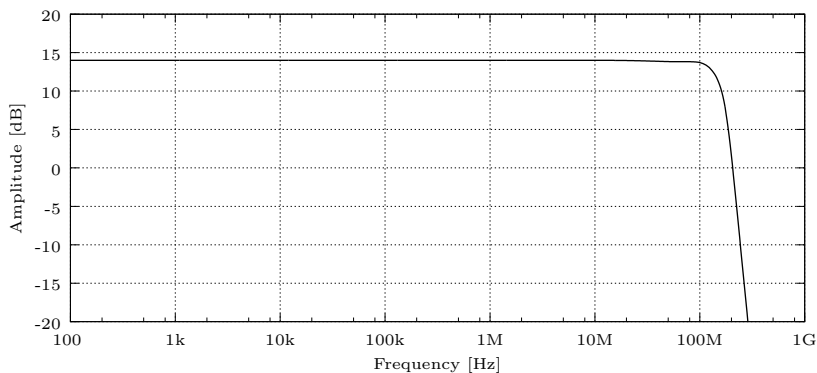
Figure 3.7: Simulated frequency response of the VGA block

the obtained frequency response and step response of the voltage amplification chain are reported in Figure 3.7 and Figure 3.8, respectively.

In the first plot, the total gain G_T has been set to $14\,\text{dB}$ by properly configuring V_{MAG} and V_{GAIN}. It can be noted that the VGA provides a flat response with a bandwidth of about $105\,\text{MHz}$.

The second plot shows instead the VGA response to an input voltage step of $0.1\,\text{V}$, for the same configured gain. Some ringing effects can be noted. In particular, the overshoot is $8.4\,\%$ and the settling time is $5.7\,\text{ns}$. These values, however, are not worrying and do not compromise the system performance,
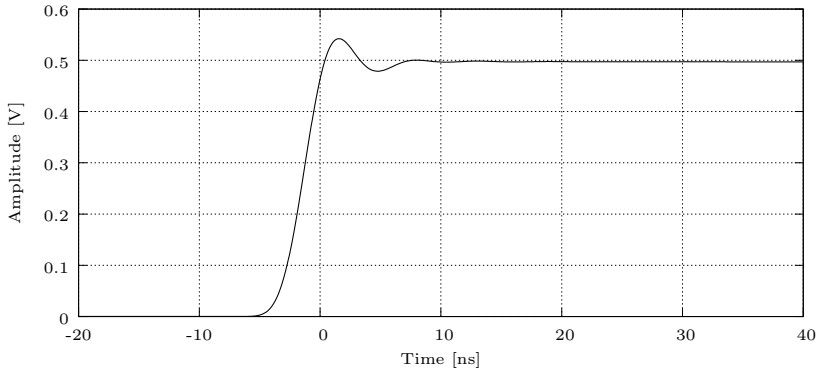
Figure 3.8: Simulated step response of the VGA block

as it will be shown in the following. The rise time is instead 3.2 ns. According to the remarks reported in Section 2.3.3, this is a very good value which well meets the design requirements.

3.3 A/D conversion and acquisition platform

To reduce the overall times and costs of the design, a commercially–available ADC/FPGA–based development platform has been adopted to implement the A/D conversion and acquisition block. Indeed, modern FPGAs easily support clock frequencies greater than 100 MHz, and provide a so large number of logic resources that can even compete with ASIC performance. In addition, by implementing the instrument core with a configurable device, an extreme flexibility is achieved. In fact, new functions and algorithms can be added by just developing new FPGA architectures, without the need of any time–consuming hardware redesign.

The selected solution is made of two interconnected boards, namely the AD9640–150EBZ conversion board and the HSC–EVAL–CZ capture board, both by Analog Devices. Their simplified diagram is shown in Figure 3.9.
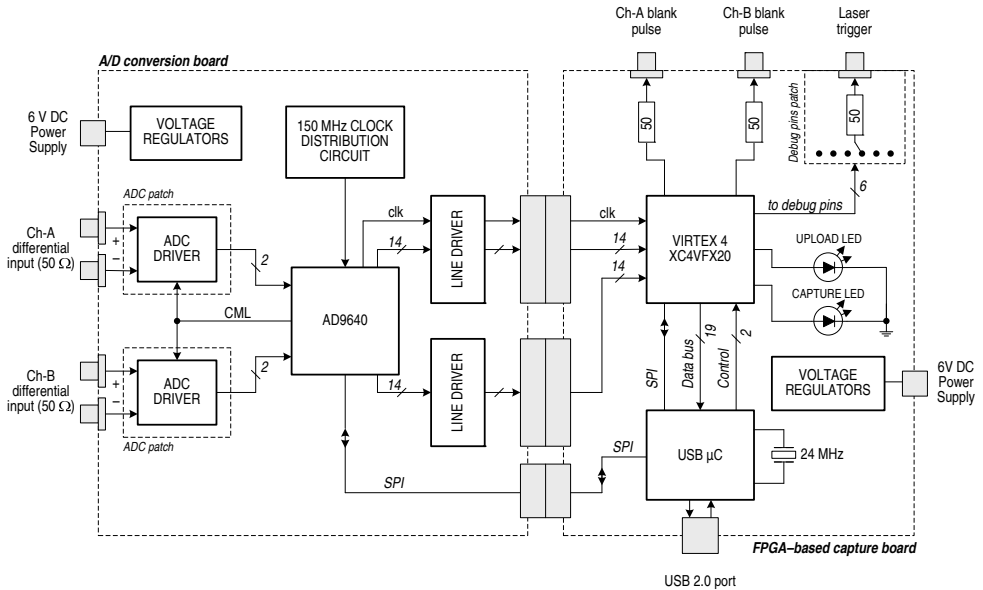
Figure 3.9: A simplified diagram of the A/D conversion and acquisition platform

3.3.1 The AD9640–150EBZ board

The conversion board is based on the AD9640 converter [50], which represents the state–of–the–art of multi–channel high–resolution ADCs. Indeed, it features two completely independent differential input channels, with a sampling frequency f_S of 150 MHz and a resolution of 14 bit, which well meets the system requirements. With these values, a sub–metre sampling resolution σ_L can be easily achieved, as it is

$$\sigma_L = \frac{1}{2} \frac{v}{f_S} = 0.67 \, \text{m} \tag{3.19}$$

where $v = 2 \times 10^8$ m/s is the average light propagation speed in the fibre, and the factor $1/2$ takes the OTDR round–trip time into account. The peak–to–peak input range is fixed on–board at 2 V, and this justifies the choice of the front–end gain G_T which has been determined in Section 3.2.1.

As shown in the diagram, signals coming from the front–end are received by two differential ADC drivers, whose main task is to match the 50 Ω impedance of the incoming lines, and to properly drive the switched–capacitors input

49

stage of the ADC. The manufacturer provides two selectable ADC drivers for each channel. The first is a passive network based on transformers, whereas the latter is an active network based on instrumentation op–amps. Unfortunately, none of them resulted suitable for our application, as their bandwidth did not extend to DC.

As a consequence, new customized ADC drivers were designed using the AD8139 op–amp and then implemented on two small PCBs, which have been mounted as a patch over the acquisition board. However, to make this operation possible, the components of the original unused drivers have been removed. In particular, the new drivers, besides providing the required bandwidth, also allow to match their output common mode to the ADC input one. This has been achieved by routing the converter CML output to the AD8139 OCM input (see Figure 3.9).

The 150 MHz sampling clock required for the ADC operation is generated on–board by a crystal oscillator and directly routed to the converter, which forwards it to the acquisition board together with the two synchronous 14 bit parallel output buses. This clock signal therefore represents also the FPGA master clock.

3.3.2   The HSC–EVAL–CZ board

The acquisition board is based on a Xilinx Virtex®–4 XC4VFX20 FPGA [51], which is a high–performance SRAM–based device implemented with the 90 nm copper CMOS technology. Besides the standard logic and I/O blocks, it embeds some specific resources to extend the conventional data processing and storing capabilities, as well as to improve the clock management. The main features of these resources, which have been used within the DTS application, are reported in the following list.

- *DSP48 Slices.* These logic blocks are based on the PowerPC architecture, and allow to perform advanced digital signal processing at firmware level, without the need of any additional hardware resource. Each slice features a 18×18bit multiplier with 2's complement arithmetic, a 48 bit full–adder, which can be configured also as an accumulator, and some

pipeline registers. The XC4VFX20 device embeds 32 DSP48 slices, which can also be cascade–connected to implement complex functions.

- *Block RAMs.* In addition to the few distributed RAM cells available within each standard logic block, the adopted FPGA features 64 dual–port synchronous memory blocks storing 18 kbit of data. These blocks can be arranged in several aspect ratios, i.e. 1k × 18 bit, 2k × 9 bit, 512 × 36 bit, or 4k × 4 bit even if dropping some memory cells. Both ports can have mutually asynchronous clock signals, and are able to perform read, write or read–during–write operations independently, in a single clock cycle.

- *Digital Clock Managers (DCMs).* A DCM is a special device which allow to fine control the duty cycle, frequency and phase of a clock signal. Its working principle is based on a digital Delay Locked Loop (DDL), which allows to reach a few picoseconds resolution. Once the input signal has been locked, it is also able to provide 90°, 180° and 270° phase–shifted replicas of the main output clock. The adopted device features 8 DCM units equally spaced within the chip area to ensure a good coverage of the logic blocks and an efficient clock distribution.

The main task performed by the architecture implemented on the FPGA is to accumulate Stokes and anti–Stokes samples coming from the ADC, as well as to handle of the trigger and blank signals. These signals are routed to the front–ends over 50 Ω matched lines. In particular, the trigger output has been obtained from a general purpose test pin header by means of a custom designed adapter. On the other hand, the accumulated data are instead transferred to the on–board μC through a dedicated 19 bit parallel bus, and then forwarded to the PC over the USB link for the temperature profile extraction.

In addition, the FPGA also implements a register bank containing all the acquisition parameters. These registers can be accessed at any time by the PC through the μC and the SPI link. Indeed, the firmware of the μC implements an USB–to–SPI bridge which allows the PC to directly issue SPI data transfers to and from the FPGA registers. By the way, the same mechanism is also used to set the ADC parameters, as SPI lines reach the conversion board as well. The PC is also able to drive two other μC pins which are used to start and stop the FPGA acquisition, as well as to request the download of the accumulated

data. Two leds mounted on–board indicate if either the acquisition or the data upload process is running.

Finally, the firmware of the µC provides a set of functions which allows for the configuration of the FPGA architecture during the system start–up. The binary programming file produced by the CAD tools can be uploaded via USB or retrieved by a preloaded on–board EEPROM, which is not shown in Figure 3.9.

3.4 FPGA architecture

3.4.1 Acquisition modes

All the 64 available Block RAMs have been used to collect incoming data. In particular, as the ADC samples are represented with 14 bit words in offset binary arithmetic, all blocks have been arranged in the $1k \times 18$ bit aspect ratio, and then organized in two $32k \times 18$ bit arrays. This means that at most 32k samples per channel can be collected for each acquisition run. Therefore, the maximum measurement range is given by

$$L_0 = \frac{32k}{f_s} \frac{v}{2} = 21.85 \, \text{km} \tag{3.20}$$

It follows that, to extend the measurement range and reach the 30 km target, some alternative solutions have to be adopted. Since the available memory resources are fixed by the chosen FPGA device and cannot be expanded, two *sample decimation* algorithms have been implemented.

The first algorithm is the conventional *one–every–n* decimation. Every $n \geq 2$ received samples, only the q–th one, with $1 \leq q \leq n$, is actually stored in memory. The n and q parameters can be selected by the user. If we indicate with S the total samples to be acquired, the covered range is now given by

$$L = \frac{S \cdot n}{f_s} \frac{v}{2} \tag{3.21}$$

and, for $S = 32k$, it results $L = nL_0$, i.e. it is extended by a factor n with respect to the *normal* acquisition mode. The diagram in Figure 3.10 shows the
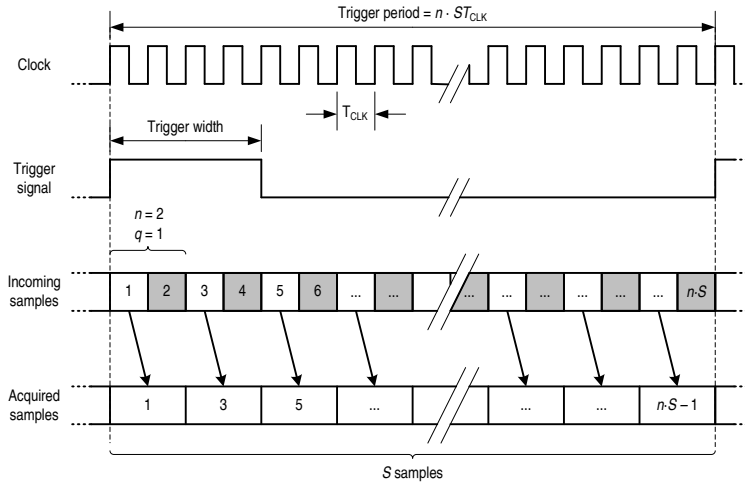
Figure 3.10: The one–every–n decimation algorithm, example for $n = 2$ and $q = 1$

one–every–n decimation example for $n = 2$ and $q = 1$. It can be observed that the length of the single acquisition, that is the trigger period, is accordingly increased to $n \cdot ST_{\mathrm{CLK}}$, where $T_{\mathrm{CLK}} = 1/f_{\mathrm{s}}$ is the system clock period. This means that, once the number of traces to be acquired and averaged is given, the total measurement time increases of a factor n. In addition, as in general happens for any decimation algorithm, also the sampling resolution gets worse of the same factor n. However, assuming that the temperature profile $T(z)$ is constant during the total acquisition time, n consecutive traces can be acquired, each of them with a different q value, and properly interlaced at the PC–side, so as to recover the original sampling resolution σ_L.

The second algorithm is instead the *sum–n* decimation. Every $n \geq 2$ received samples, their sum is stored in memory. This technique, which is represented in Figure 3.11 for $n = 2$, does not allow to interlace consecutive traces, but it rather allows to reduce the number of averages. Indeed, as the noise affecting each sample can be considered a Gaussian ergodic process, by adding n consecutive–in–space samples in each available memory slot, the number of traces to be accumulated and then averaged for the noise reduction can be reduced by a factor n. This means that the total measurement time can be kept equal to the one required for the normal acquisition mode.

From (3.21) it follows that $n = 2$ is enough to reach the $30\,\mathrm{km}$ target.
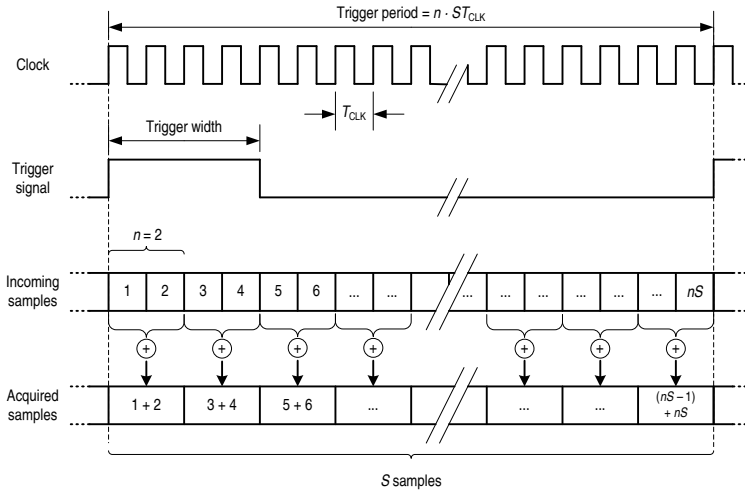
Figure 3.11: The sum–n decimation algorithm, example for $n = 2$

Despite this, to give the system a good flexibility, the developed architecture is able to handle decimation factors up to $n = 4$, so as to virtually extend the instrument measurement range up to $84\,\mathrm{km}$.

Besides the decimation algorithms, two other acquisition modes have been implemented in the FPGA, i.e. the *delayed acquisition* and the *interleaved sampling*. The first allows to introduce a delay between the rising edge of the trigger signal and the actual acquisition process. This can be useful, for instance, if only a specific segment of the fibre has to be investigated, as it allows to use the normal mode also at those distances which are typically unreachable due to memory limitations. The initial delay is expressed in number of clock cycles.

On the other hand, the interleaved sampling allows to increase the nominal sampling resolution of the system. In this case, the leading idea is to interlace a given number r of traces, each of them delayed of T_{CLK}/r with respect to the previous one. To this end, a fine control of the timing relationship between the clock signal and the laser trigger is required. Thanks to the available DCM blocks, the developed architecture allows to set $r = 2, 4$ (see Section 3.4.6). This means that the sampling resolution can be pushed down to $\sigma_L/4 = 0.17\,\mathrm{m}$.
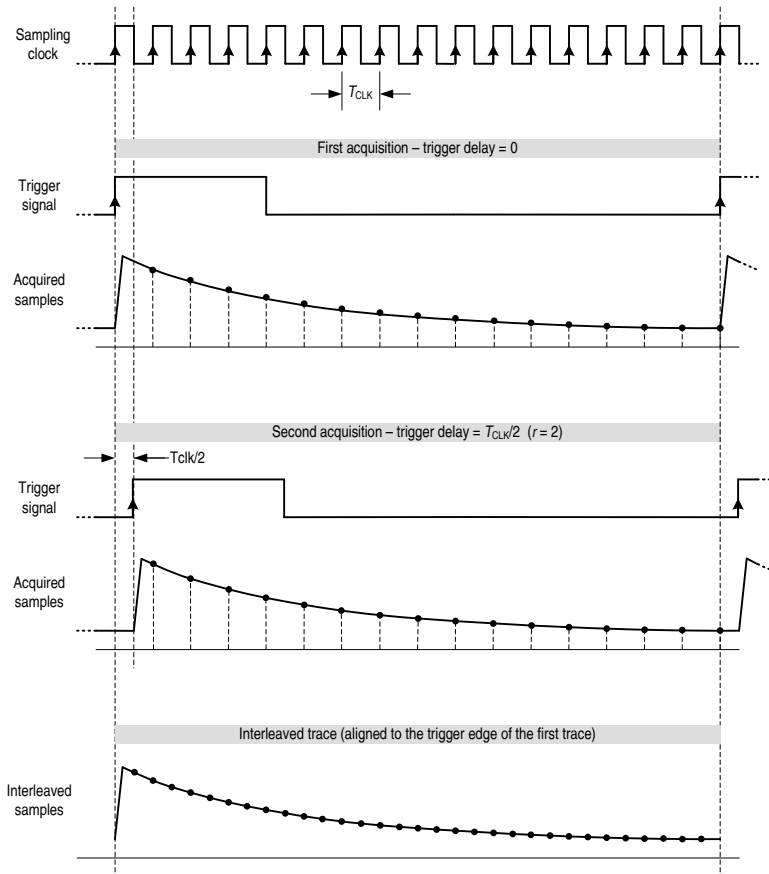
Figure 3.12: The interleaved sampling, example for $r = 2$

In Figure 3.12, an example of interleaved sampling for $r = 2$ is reported. It is worth noting that the sampling clock is taken as the reference signal, whereas the trigger is delayed. This means that the samples collected during the second acquisition will be shifted $T_{\mathrm{CLK}}/2$ ahead at PC–side, and then combined with those of the first acquisition to obtain the interleaved trace with a sampling resolution $\sigma_L/2$. As happens for decimation, it is clear that once the number of traces to be collected and averaged is given, the total measurement time increases of a factor r, because r acquisitions are needed to build a single trace.

3.4.2   The top–level

At the top hierarchy level, the FPGA architecture is represented by the block diagram of Figure 3.13.

The core of the system is the *DTS Module*, whose task is to properly generate the trigger and blank signals, acquire the Stokes and anti–Stokes samples according to the particular acquisition mode selected, build the accumulated traces and finally make them available to the µC. The latter is indeed able to directly access the samples memory of the DTS Module and download the accumulated data by means of the Data bus signal group. In addition, the µC drives the two lines _MR and _WEN which control the global reset and the acquisition start, respectively. The DTS Module also controls the Capture and Upload leds, which are turned on when the samples acquisition or the data transfer are in progress.

The *SPI Interface* block handles the communication between the FPGA and the SPI bus, which is used by the µC to set and retrieve the configuration
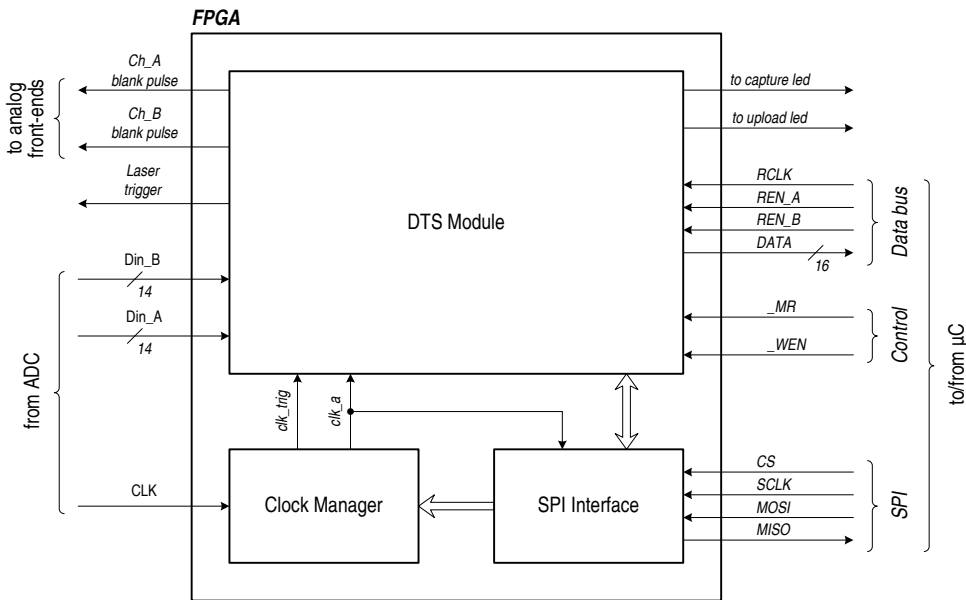
Figure 3.13: The FPGA top–level architecture

parameters. In particular, their values are mapped in a 32×8 bit register bank, which also includes a special read–only Status register containing the current status information of the DTS Module.

The task of the *Clock Manager* module is instead to lock the 150 MHz clock signal CLK coming from the ADC and produce the two internal clock signals clk_a and clk_trig, at the same frequency. The first is routed to both the DTS Module and the SPI Interface, and it used as the master clock signal. The latter is instead used only for the trigger generation, as its phase can be fine controlled to implement the interleaved sampling technique.

3.4.3 The DTS Module

The architecture of the DTS Module is reported in the simplified diagram of Figure 3.14. To handle the samples coming from the ADC, two identical *Channel Modules* (ChMs) have been implemented. Their task is to collect the samples coming from the ADC output channels, build the sum traces, and store them in the available Block RAMs. The two ChMs work in parallel, and their synchronous operation is obtained by means of the control signals coming from the *ChM Controller*.

The collected data can be accessed by the µC through the *Read Interface* block. To start the download, the µC selects which ChM wants to read by activating REN_A or REN_B signals, and then provides a 48 MHz read clock on the RCLK line. The Read Interface starts reading from the first memory location of the selected channel and outputs data on the 16 bit parallel bus. In particular, as the ChM memory is 18 bit wide, two RCLK cycles are actually needed to transfer each sample. This issue, which avoids to halve the time required for the upload, could not be solved because the width of the parallel bus is imposed by the hardware platform. Nevertheless, the first tests carried out on the prototype showed that this overhead on the total upload time does not significantly afflict the overall system performance. In addition, when the upload is in progress, the Read Interface turns the upload led on.

The *System Controller* is the DTS Module supervisor. It receives the asynchronous start/stop and reset signals from the µC, scans the working cycles of the ChM Controller, handles the capture led, and keeps the system Status
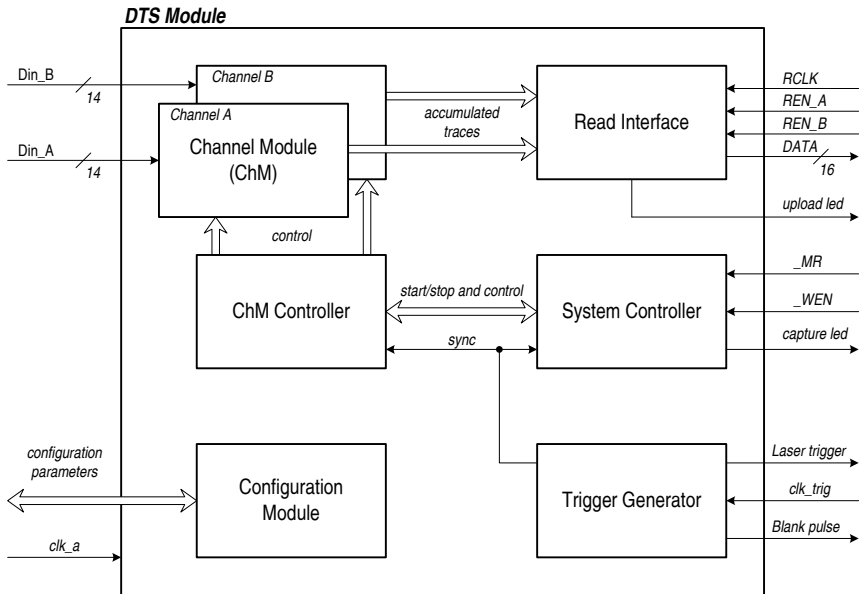
Figure 3.14: The DTS Module architecture

register updated, so as to let the user know if the acquisition is completed and data are ready to be uploaded. This latter task is carried out by means of the *Configuration Module*, which routes the input and output signals of the register bank to each block of the DTS Module. In particular, the Configuration Module is also able to compute some second–level parameters which are a function of the main parameters stored in the register bank. This is achieved by means of the DSP48 Slices, so that both the register bank and the Configuration Module itself are kept rather small and simple.

Finally, the *Trigger Generator* continuously produces the trigger and blank signals, as well as a dedicate sync signal which is used by both the System and the ChM controllers to synchronize the acquisition start to the launch of the trigger pulse. It is worth noting that, in addition to the master clock clk_a, this block also receives the clk_trig signal from the Clock Manager, which is used to produce the delayed trigger pulses whenever an interleaved acquisition is requested.

3.4.4 The Channel Module

The architecture of the Channel Module (ChM) is reported in Figure 3.15. Following the data path, the samples coming from the ADC (IN signal) are initially routed to the *Shifter* block. This block is a sort of requantizer, as it allows to right–shift the incoming 14 bit samples of SHIFT positions, i.e. to cut out the first SHIFT LSBs. Thanks to this block, the number of traces N_0 which can be safely accumulated by the FPGA without any numerical representation error can be significantly extended. In fact, if the right–shift option is considered, N_0 can be written as (see Section 2.4.2)

$$N_0 = 2^{18-14+\text{SHIFT}} \tag{3.22}$$

in which the FPGA memory width has been fixed at 18 bit. It can be seen that for (SHIFT = 0), no more than $N_0 = 16$ traces can be accumulated. This means that if a large number N of traces has to be averaged, the required number N/N_0 of transfers from the FPGA to the PC may become large as well, and the time needed for the upload may even dominate over the one spent just for the acquisition. On the other hand, by setting for instance (SHIFT = 6), i.e. by representing samples with 8 bit, as it is commonly done in practice, it results $N_0 = 1024$, and the time required for the data upload can be significantly reduced. However, it is important to point out that the effective quantization noise increases with SHIFT. For this reason, SHIFT should be kept lower than the value for which the quantization noise equals the noise of the input signal. From bench tests, we found that values of SHIFT up to 8 can be applied.

The Shifter block is implemented by means of a shift register, and provides the shifted data on its Q output after one clock cycle. The shifted–out bits are replaced with zeros. In addition, the last bit which has been removed is presented on the C output, and it is used as the carry input for the following adder. This allows to compensate the offset introduced by the shift operation. Indeed, it can be verified that after the shift operation, the equivalent requantized ADC transfer characteristic has an offset which increases with SHIFT. As an example, Figure 3.16 shows the case for a 4 bit ideal ADC with (SHIFT = 2). It can be seen that the dashed line connecting the central–code points crosses the input voltage axis at

$$V_{\text{offset}} = \frac{\text{LSB}}{2} \sum_{j=0}^{\text{SHIFT}} j \tag{3.23}$$
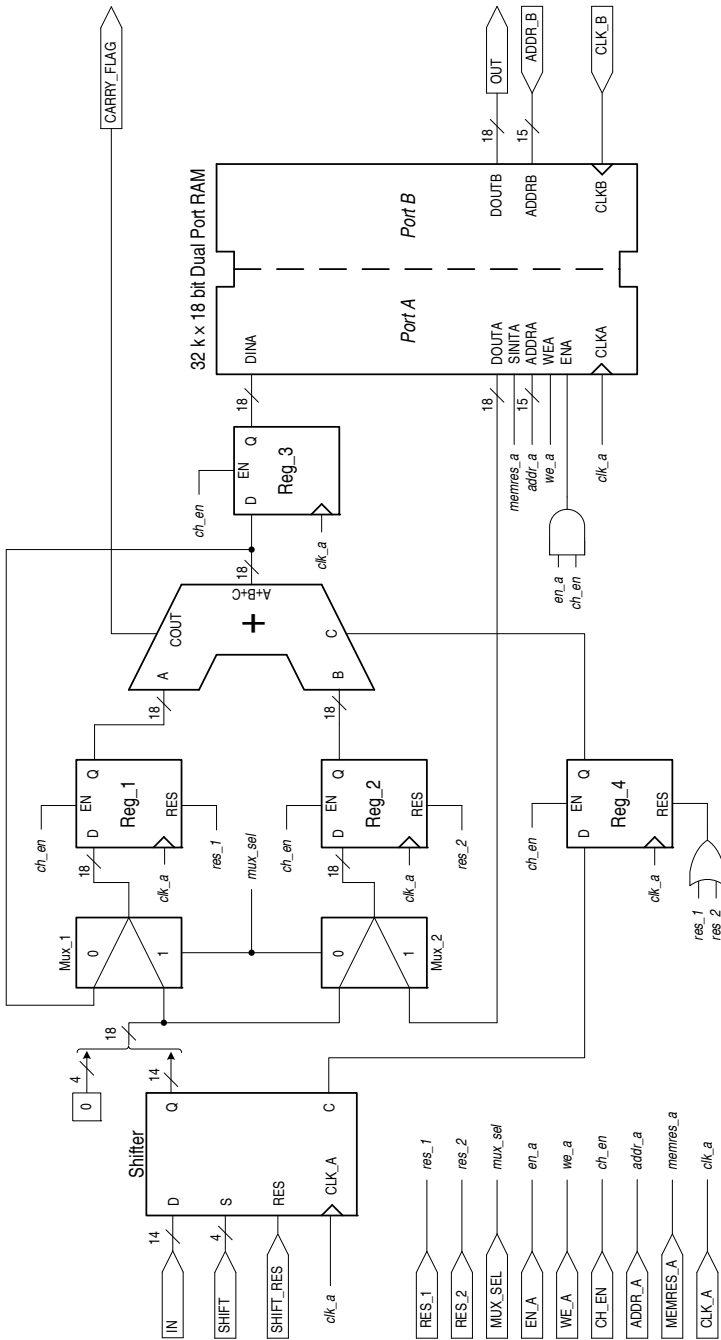
Figure 3.15: The Channel Module architecture
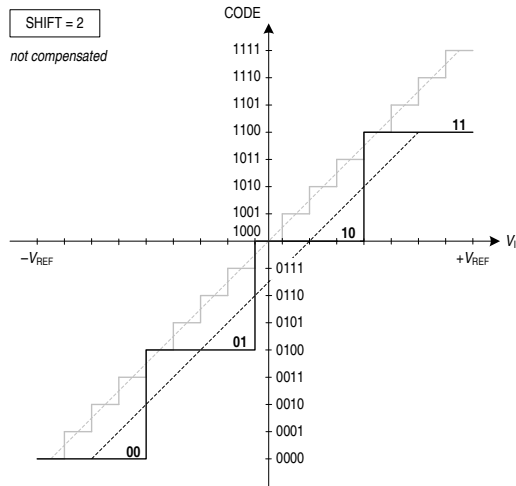
Figure 3.16: Example of not compensated Shifter offset for $n = 4$ and SHIFT $= 2$

where LSB $= V_{\mathrm{REF}}/2^{n-1}$ and n is the ADC resolution. As a consequence, as SHIFT increases, the offset may become too large and saturation may occur. If instead the output C is added to Q, the characteristic turns into the one reported in Figure 3.17. In particular, it can be verified that in this case the offset is kept at 0.5 LSB, and it does not depend on SHIFT.
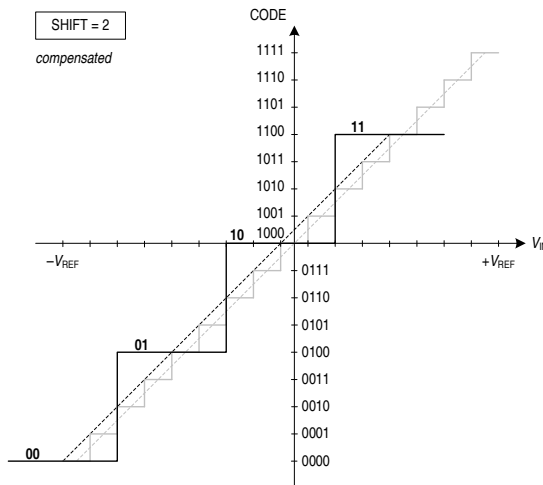
Figure 3.17: Example of carry–compensated Shifter offset for $n = 4$ and SHIFT $= 2$

Next to the Shifter, a synchronous processing block is implemented. The 18 bit full–adder receives data to be added from Reg_1 and Reg_2 registers, whereas the carry input is obtained from the Shifter C output through Reg_4. All these registers are enabled by the ch_en signal, and can be synchronously reset by means of res_1 and res_2. Enable and reset lines, along with mux_sel and the other signals related to the memory block, are properly handled by the ChM Controller according to the specific acquisition mode selected.

It is important to point out that the presence of the three registers along the data path is not theoretically necessary, as all the acquisition modes can in principle be implemented with just the shifter, the adder and the memory. However, their presence is required to break some critical paths which otherwise would arise between the memory output, the adder and the memory input. Consider in fact that all the available Block RAMs have been used to build the two channel modules. As these blocks are arranged in columns within the FPGA area, some of them will inevitably be placed far from the logic resources on which the adder is implemented. This means that without any pipeline register, data should come out from any memory location, reach the adder, be added to the incoming samples, and then come back to the memory in a single clock cycle of 6.6 ns. By means of post place–and–route simulations, we unfortunately found that the propagation delay could not be made lower than 8 ns, even when the best available place–and–route optimization algorithms were used. Hence, being the theoretical approach not feasible in practice, we adopted the pipelined architecture proposed in Figure 3.15.

The selected acquisition mode can be identified with the n parameter defined in Section 3.4.1. For $n = 1$ the normal acquisition is performed, whereas for $n = 2 \ldots 4$ a decimation algorithm is applied. In particular, once n is given, the acquisition process can be considered as divided into n subsequent steps, which are cyclically repeated while samples are being collected.

During the first step, which of course is always performed, mux_sel is kept at 1. This means that Reg_1 receives data from the Shifter, whereas Reg_2 from the memory (see Figure 3.18). If the incoming sample has to be accumulated in the current memory location, all the registers are enabled. If instead the incoming sample has to be dropped (as happens for a *one–every–n* decimation with $p \neq 1$), Reg_1 is reset, so that zero is added.

On the other hand, during the remaining steps mux_sel is kept at 0. This
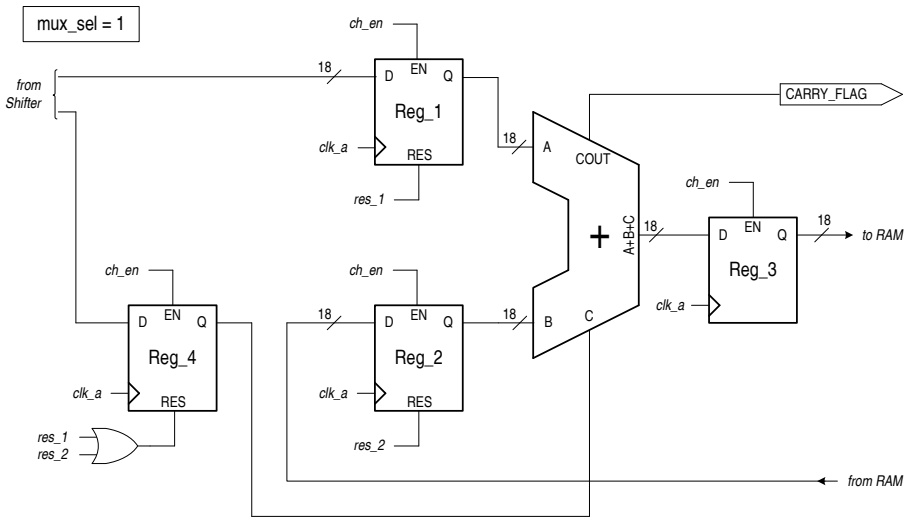
Figure 3.18: Channel Module configuration for mux_sel = 1

way, new incoming samples can be accumulated to those collected in the previous steps (see Figure 3.19). In particular, if now a sample has to be dropped, the ChM Controller activates the Reg_2 reset line accordingly.
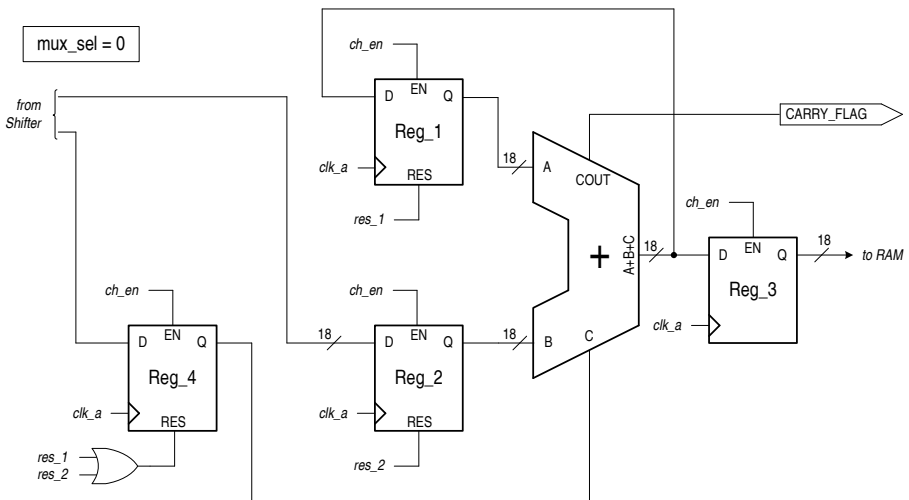
Figure 3.19: Channel Module configuration for mux_sel = 0

63

The dual–port RAM allows to transfer data between the different clock domains of the FPGA and the µC. The Port A is clocked by the 150 MHz master clock and it used during the acquisition, whereas the Port B, which is connected to the Read Interface, works at 48 MHz and it is used for uploading the acquired data to the PC.

Focusing on the Port A, it is worth noting that the maximum input data rate, which corresponds to the system sampling rate, is 1 sample per clock cycle. Hence, to properly accumulate the incoming samples in the sum–traces, the Port A must be able to perform a read and write operation in a single clock cycle. This can be achieved by configuring the Block RAMs in the *read–first* mode: when a new data is written at a given location, the old data that has been replaced is stored in the memory output register, and can be used at the next clock cycle.

Once the number S of samples per trace is given, the leading idea is therefore to implement a ring buffer of $S-1$ locations, ranging from address 1 to address $S - 1$. Suppose to start the acquisition of the first trace at address 1. The second–last sample of the first trace will be stored at location $S - 1$, and the last sample will be stored back to location 1, overwriting the first acquired sample. However, this old sample will be available in the memory output register, and it will be accumulated with the first sample of the incoming second trace. If their sum is stored at location 2, the second sample of the first trace will be moved to the output register, and used for the second sum, and so on.

While this process is being repeated, the whole sum trace contained in the ring buffer moves one location ahead after each acquisition. In particular, given the number N_0 of traces to be accumulated, the start address can be properly chosen so that the very last data is written in the last location $S-1$. When this happens, the memory output register contains the first sample of the accumulated trace. As a consequence, once the acquisition is ended the content of the output register is stored in the previously reserved location 0, so that the µC can easily read the sum trace in a linear way from address 0 to address $S - 1$ and upload it to the PC without the need of any unroll process.

The actually implemented technique is very close to the one described. The only difference is due to the presence of pipeline registers. In general, if k is the pipeline depth, the first k locations in memory have to be reserved. It
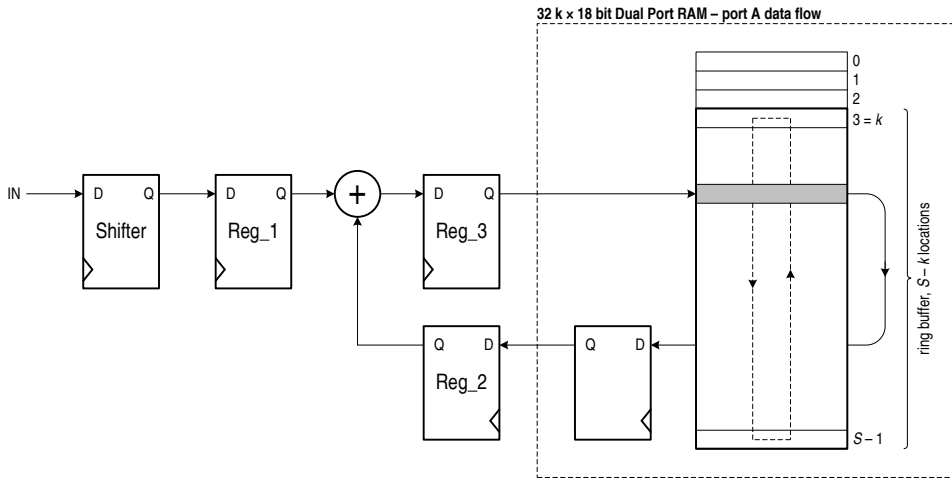
Figure 3.20: The ring buffer implementation

can also be verified that the start address a_{START} has to satisfy the following relation

$$a_{\mathrm{START}} = S - |SN_0|_{S-k} \tag{3.24}$$

In our case it is $k = 3$ (see Figure 3.20). At the end of the acquisition, the first three samples of the sum trace are stored in Reg_3, Reg_2 and in the memory output register, respectively. Hence, their content is stored in the first k reserved locations before allowing the µC to start the upload.

3.4.5 The Trigger Generator

The main task of the Trigger Generator block, whose simplified architecture is reported in Figure 3.21, is to produce the trigger and blank signals. These signals have the same period, which corresponds to the trace length S, whereas their pulse width can be specified independently. The Timebase controller receives the period and width parameters from the Configuration Module, and properly produces the set and reset signals for the two synchronous SR flip–flops FF-1 and FF-2.

As shown in the diagram, FF-1 directly drives the blank output, whereas FF-2 is connected to the *Trigger Resync* block. Here, func_mode[1] is a con-

Figure 3.21: The simplified Trigger Generator block diagram

figuration bit coming from the register bank, and clk_trig is the phase–tunable clock coming from the Clock Manager. According to the chosen trigger delay, clk_trig can be driven by one of the four possible phases produced by the Clock Manager, i.e. clk_trig_0, clk_trig_90, clk_trig_180 and clk_trig_270, whose relative delay is $T_{\mathrm{CLK}}/4$.

The value of func_mode[1] depends on the selected delay. To obtain a 0 or a $T_{\mathrm{CLK}}/4$ delay, func_mode[1] is forced to 1, so that FF-4 is driven by FF-3. If instead a $2T_{\mathrm{CLK}}/4$ or a $3T_{\mathrm{CLK}}/4$ delay is desired, func_mode[1] is forced to 0, so that FF-4 is driven by FF-2. This strategy allows to safely avoid any setup or hold time violation on FF-4.

Suppose for instance to select the 0 delay. In this case, it will be clk_trig = clk_trig_0. If clk_trig_0 is in its turn delayed of $2T_{\mathrm{CLK}}/4$ with respect to clk_a, the situation is the one reported in the first part of Figure 3.22. It can be observed that the rising edge of clk_trig is rather far from the transition of the FF-2 output. The same happens also when the $T_{\mathrm{CLK}}/4$ delay, and hence the clk_trig_90, is selected.

On the other hand, if the $2T_{\mathrm{CLK}}/4$ delay is selected, some timing problems may arise, as there is a rising edge of clk_trig_180 very close to the FF-2 output transition. For this reason, to avoid metastability conditions and output jitter, the FF-3 output is selected in this case. As FF-3 is clocked on the falling edge of clk_a, it can be seen in timing diagram that the previous dangerous situation is avoided. The same is true also for $3T_{\mathrm{CLK}}/4$ delay and clk_trig_270.

Figure 3.22: The Trigger Generator timings

Finally it is worth noting that, with this method, the safe operation of FF-4 is ensured if its setup and hold times are lower than $T_{\mathrm{CLK}}/2 = 3.33\,\mathrm{ns}$ and $T_{\mathrm{CLK}}/4 = 1.66\,\mathrm{ns}$, respectively. The actual timing specifications of FF-4 can be derived from the FPGA datasheet. As this flip–flop is mapped on an I/O logic block, its setup time is $0.34\,\mathrm{ns}$ and its hold time is $-0.1\,\mathrm{ns}$. Comparing these values with the previous ones, a good margin is achieved.
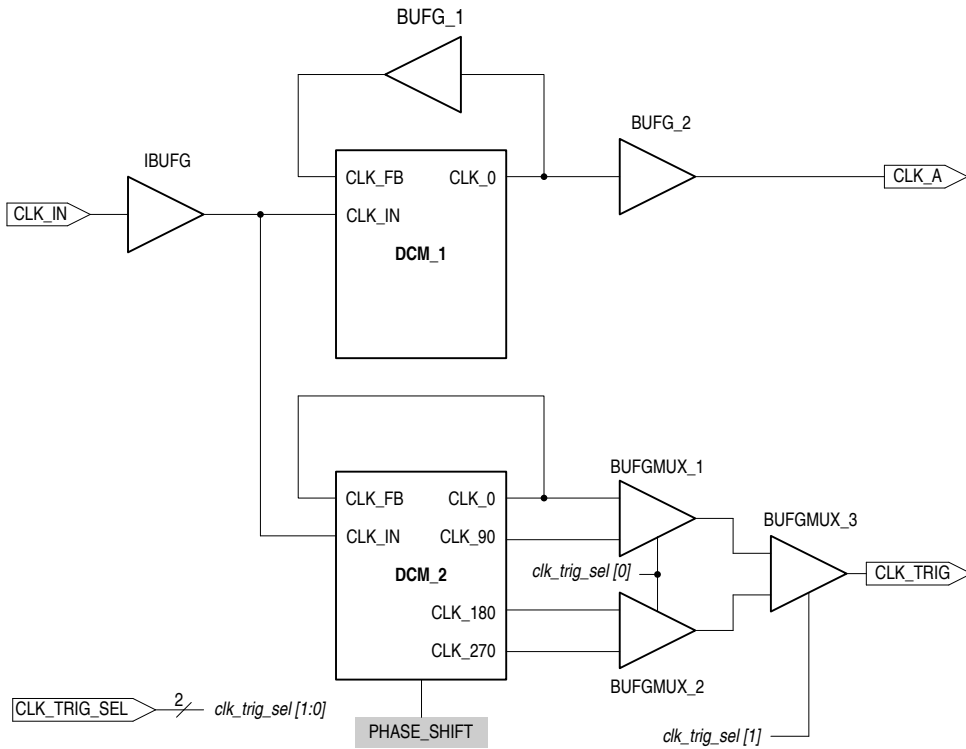
Figure 3.23: The Clock Manager architecture

3.4.6 The Clock Manager

The architecture of the Clock Manager is reported in Figure 3.23. As stated before, it is based on DCM blocks. The 150 MHz clk_in signal coming from the ADC is initially fed to an IBUFG block, which is an FPGA primitive specifically designed to buffer incoming clock signals. Indeed, the delay introduced by IBUFG is equal to the one of normal IBUF input buffers, which are typically used for data signals. This way, clock–to–data synchronization is maintained and the correct operation of sequential logic resources is ensured. Nevertheless, the IBUFG output needs to be further amplified so as to provide a good clock distribution over the whole chip area. This is achieved by means of BUFG_1, BUFG_2 and DCM_1 blocks.

BUFG primitives are internal clock buffers which show perfectly matched

propagation delays. Thanks to this feature, they can be used together with DCM blocks to implement phase matched clock distribution networks. Indeed, the main task that the DCM is able to perform is to fine tune the phase of its main output CLK_0 so that the phase error between the input CLK_IN and the feedback CLK_FB is cancelled. Looking at the circuit in which the DCM_1 is instantiated, it follows that the BUFG_2 output, i.e. the system master clock clk_a, is matched with the CLK_IN input, and hence with data. This is obtained thanks to BUFG_1, which introduces the same delay of BUFG_2 on the DCM_1 feedback path.

The IBUFG output is also routed to the DCM_2 input, which is used for the clk_trig generation. In this case, all the available output phases of DCM_2 are used, and the one that has to be routed to the Trigger Generation block is selected by means of delay–matched BUFGMUX primitives. It is worth noting that the feedback path does not contain any delay element. This is done because, in this case, an advanced feature of the DCM is used. In fact, each DCM primitive provides an attribute which can be specified *after* the place–and–route process, and which allows to set a particular phase offset to be maintained between CLK_IN and CLK_FB. This attribute, whose default value is 0, is called PHASE_SHIFT (PS) and it is bounded to the actual phase offset $\Delta\phi$ by the following relation

$$\Delta\phi = \frac{\text{PS}}{256} \, T_{\text{CLK}} \quad \text{with } -255 \leq \text{PS} \leq 255 \tag{3.25}$$

Without any delay element on the feedback path, the PHASE_SHIFT attribute allows to fine set the phase relation between clk_a and clk_trig_0. As a consequence, by means of post place–and–route simulations, the value of PHASE_SHIFT has been chosen so that the FF-4 timing requirements outlined in Section 3.4.5 were met.

3.4.7   Synthesis and Implementation

The whole FPGA architecture has been developed using the Verilog hardware description language. Each step of the design flow, i.e. the logical synthesis, the functional simulations, the implementation and the final post place–and–route simulations, have been carried out within the Xilinx Integrated Software Environment® (ISE) 10.1.03.

The most challenging requirement to met was related to the master clock frequency. Indeed, it is necessary to ensure that all sequential logic, and in particular all the synchronous Block RAMs which are organized in vertical columns crossing the whole chip area, operate correctly at the working frequency of 150 MHz without any setup or hold time violation.

As discussed in Section 3.4.4, a three–level pipeline approach has been adopted in each ChM to break some critical paths between the memory and the adder. Moreover, the automatic synthesis and implementation tools have been configured by selecting the most advanced *speed* optimization algorithms available, instead of the *area* optimization ones. Nevertheless, some low–level tuning has been required to let the whole implementation process complete successfully.

As far as the logical synthesis is concerned, the best results were obtained by enabling the *Register balancing* algorithm. Basically, it allows the software to reshape the time–critical combinatorial networks so that registers can be equally distributed along the data path, and propagation delays can be balanced. Combining this choice with the *Multi–pass place–and–route* algorithm, the 150 MHz constrain was satisfied. In particular, the multi–pass algorithm repeats the place–and–route process for a given number of times, and for each iteration tries to improve the performance obtained in the previous cycle by adopting different place–and–route strategies whenever a critical network is detected. With 3 iterations, the minimum allowed clock period was pushed down to 6.283 ns. This value corresponds to a maximum allowed clock frequency of about 159 MHz.

In Table 3.1 the utilization statistics of the main FPGA logic resources are reported. It can be noted that the Block RAMs are the most used, whereas only the 16 % of the available slices are occupied. This means that there is still a great room for the future development of other functionalities. In addition, the two DCM primitives that are used in the Clock Manager block can easily be recognized, as well as the three clock multiplexer implemented by means of the BUFGCTRL primitives. Finally, the DSP48 slices have been used in the Configuration Module to compute some internal parameters derived from those contained in the SPI register bank.

Besides the tuning of the automatic synthesis and implementation tools, some additional low–level optimization has been required to satisfy the clk_trig

Table 3.1: Main FPGA utilization statistics

Resources	*Used*	*Available*	*Utilization*
Number of occupied slices	1443	8544	16%
Number of slice flip–flops	913	17088	5%
Number of 4 input LUTs	2092	17088	12%
Number of DCM_ADVs	2	4	50%
Number of BUFG/BUFGCTRLs	9	32	28%
Number used as BUFGs	6		
Number used as BUFGCTRLs	3		
Number of FIFO16/RAMB16s	64	68	94%
Number used as RAMB16s	64		
Number of DSP48s	3	32	9%

requirements outlined in Section 3.4.5. Indeed, from a first analysis of the place–and–route result, it was observed that FF-4 was mapped on an I/O block, whereas FF-2, FF-3 and MUX-3 on different logic blocks (see Figure 3.21). Moreover, the phase delay between clk_trig_0 and clk_a was not the required $T_{\text{CLK}}/2$.

To address these issues, the Xilinx FPGA Editor has been used. This tool allows the designer to edit each resource attribute, change its position or connections, and even fix them before the automatic place–and–route is executed.

Therefore, the first operation was the mapping of FF-2, FF-3 and MUX-3 to the same logic block. This allowed to reduce the clock skew between FF-2 and FF-3, which is clocked on the clk_a falling edge, and also to equalize the propagation delays towards MUX-3. Then, the DCM_2 PHASE_SHIFT attribute was set to 0 and the place–and–route process was repeated.

The new implemented architecture was checked by means of simulations. In

particular, the phase delay between clk_trig_0 and clk_a was measured. It resulted that clk_trig_0 required a shift $\Delta\phi$ of about 1.82 ns to satisfy the $T_{\mathrm{CLK}}/2$ delay requirement. As a consequence, the PHASE_SHIFT (PS) attribute of DCM_2 was set to

$$\mathrm{PS} = 256 \; \frac{\Delta\phi}{T_{\mathrm{CLK}}} \approx 70 \qquad (3.26)$$

The effect of this tuning process was validated by simply repeating the post place–and–route simulations. Indeed, it is important to point out that the change of resource attributes does not require the execution of new implementation processes. The results obtained while evaluating the Trigger Generator performance during bench tests are reported in Section 5.3.

4

Firmware/Software Implementation

4.1 The platform–based approach

To design the firmware and the software part of the system, several state–of–the–art solutions proposed in the literature for PC–based measurement instruments have been investigated. PC–based architectures are widely used in scientific and industrial data acquisition applications, as they allow the user to take the most of the advanced computational resources provided by modern CPUs and analysis softwares, and use them to process the collected signals in a very fast, efficient and cost–effective way [52–54].

The basic diagram of such systems can be summarized by the generic architecture reported in Figure 4.1, in which several cascaded hardware blocks allow to condition the incoming signals, convert them from analog to digital, acquire their samples, and finally transfer them over a high–speed data link to the PC for processing, displaying or storing. Each block usually features a set of parameters which varies according to the block type and to its internal architecture. These parameters need to be fine tuned to the particular application the system is used for. Hence, an additional control link between the PC and the control devices is often required. This link is generally implemented by means of microcontrollers, DSPs or FPGAs, which receive commands from the PC, e.g. through USB or Ethernet interfaces, and route them to the control devices via SPI, I2C, UART, or using general purpose I/O lines [55].

Generally, the design of the system control part is a complex task, which can be faced in several different ways. The traditional approach consists in the

concurrent development of the hardware, firmware and software. Thus, every implementation detail, from the partitioning strategy to the physical resources and the control algorithms, can be freely chosen and individually optimized. However, this approach easily turns into a time–consuming job for the designer whenever a medium complexity architecture has to be implemented.

To address this issue, several platform–based approaches in which the speed up of the design process is counterbalanced by a reduced optimization level have been proposed in the literature. The basic idea of a platform based design is to use powerful firmware/software frameworks which are customized to the specific requirements of the targeted system. This way, the same architecture is adapted to the different applications, with obvious benefits in terms of development times and design re–use. However, the customization of these frameworks is rarely a simple task, as it requires a wide use of object–oriented programming [56, 57], not only when implementing complex shared or network–distributed systems [58, 59], but also when dealing with simpler architectures. In addition, most of them embed real–time operating systems, and hence are not suitable for small and cheap microcontrollers [55].

Given this scenario, we decided to develop a new stacked firmware/software platform, which allows the control link of PC–based instruments to be implemented also by low–end microcontrollers. The customization task can be accomplished in a simpler way with respect to the previous ones, as the PC–side of the platform is based on the LabVIEW® graphical developing environment. Moreover, since LabVIEW is widely used in both scientific and
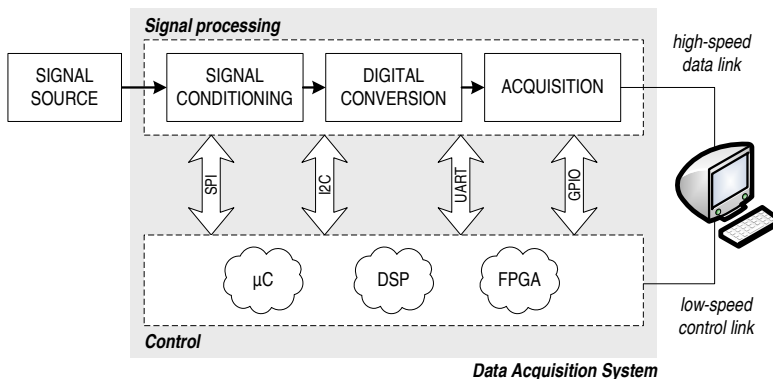
Figure 4.1: The generalized architecture of a PC–based data acquisition system
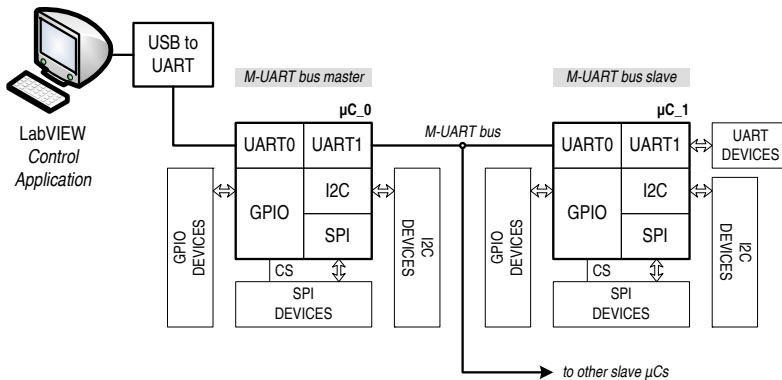
Figure 4.2: The platform reference hardware architecture

industrial fields, this choice also allows to rapidly embed the system in pre–existing applications with a very low effort.

4.2 The layered architecture

4.2.1 Overview

Without loss of generality, let us assume the hardware architecture shown in Figure 4.2 as a reference for the system control part. Comparing this diagram with Figure 2.2, a good part of the DTS architecture can be easily recognized. Indeed, all the block devices of the acquisition system are connected to a microcontroller. Each microcontroller is a node of a message–based Multiprocessor UART (M–UART) bus [27], and it is identified by a unique 8 bit address. The first node (μC_0) is the bus master, and communicates with the PC by means of a dedicated UART port. The other nodes (μC_0,...) are slaves. It is worth noticing that with this approach, once the number and type of block devices is given, the required number of nodes depends only on the connectivity resources provided by the chosen microcontrollers and by the adopted hardware partitioning strategy. Also, in the example of Figure 4.2, each microcontroller has two UART ports, so that only one UART device per slave node is allowed. Should this represent a limitation, a microcontroller with more ports can be freely chosen.

(a) Layer structure

(b) Application example

Main Application
Block API
Device API
Communication
Interface Bridge

| I2C | SPI | UART | GPIO |

LabVIEW

firmware

Set $V_{APD} = V_0$
APD_bias(setValue,V0);
MAX1932(node_addr, setDAC, value);
send_msg(node_addr, SPI, device_idx, data);
SPI_transfer(device_idx, data);

| I2C | SPI | UART | GPIO |

| MAX 1932 | V_0 |

LabVIEW

firmware

hardware
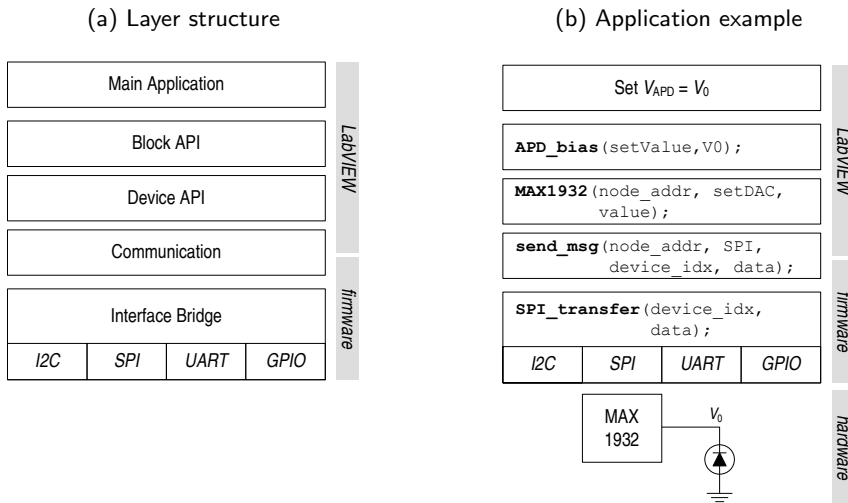
Figure 4.3: The platform architecture

The leading idea on which the platform is based is to let LabVIEW access the communication peripherals of each node and perform read and write operations directly on block devices. As a consequence, microcontrollers simply act as bridges, and the implementation of the different device protocols and control tasks is demanded to the PC. This approach allows to use the same firmware for all the nodes. Only a slight customization of some header files is indeed required to specify the node type and address, as well as to assign the available IO pins of the microcontroller to SPI chip selects (CS) or GPIOs. On the other hand, even if the complexity of the control algorithms is pushed towards the PC, the development of the LabVIEW application can be kept rather simple by following the layered structure represented in Figure 4.3(a).

The main control application is built on top of the *Block API* layer, which provides the handling functions of all the system blocks. These functions mask the actual hardware implementation of the system, and allow the programmer to directly set or retrieve a particular block parameter (e.g. the total gain of an analog front–end). To do this, the Block API layer maps each block parameter to one or more physical devices. When a Block API function is called, it determines which devices need to be handled (e.g. a variable gain

amplifier inside the front–end), the address of the nodes they are connected to, and the operation to be performed, and then it appropriately calls the underlying *Device API* layer. This layer implements in its turn the actual driver of each device. Indeed, according to the requested operation and to the protocol of the selected device, each *Device API* function produces a specific byte control sequence, and passes it to the *Communication* layer along with the node peripheral to be used and the read/write access mode. The Communication layer wraps the node address, the interface type, the access mode and the byte sequence in a single request message, and then transfers it to the master node. In particular, if this node is not the actual recipient, the message is forwarded over the M–UART bus to slave nodes. Finally, the addressed node serves the request, and sends back a response message containing the operation result, which is returned up to the application.

4.2.2 Firmware layer

The firwmare layer has been developed for AVR microcontrollers [27]. An interesting aspect of its implementation is the way in which messages are handled between nodes. When the master node receives a request from LabVIEW, it first parses the destination address. Then, if it has been addressed, it starts serving the request, otherwise it forwards the message over the M–UART bus. This bus is made only by the master node TX and RX lines, which are routed respectively to the RX and TX pin of each slave node. In particular, to avoid conflicts, slave nodes normally keep TX pins disabled. The master node is configured to send extended 9 bit characters, in which the 9th bit is used to tell if the remaining byte has to be treated as a normal data or as a node address, i.e. to distinguish *data frames* from *address frames*. The latter are used to select a particular slave node. In fact, before forwarding the message, the master node wraps the destination address in an address frame and sends it over the bus. All the slave nodes receive this frame along with the request, but only the one that has been addressed actually processes the message. Then, when the operation is completed, the selected slave temporarily enables its TX pin and sends a response message back to the master node, which finally returns it to LabVIEW. After receiving the response, LabVIEW is allowed to send a new request.

The operations that each node is able to perform slightly vary according

to the targeted interface. To handle I2C and UART devices, LabVIEW can issue conventional read and write commands. In particular, in the first case the I2C address byte of the selected device needs to be included in the request, in the latter the index of the microcontroller UART port to be used must be provided. SPI devices are instead controlled by means of a single data transfer command, which allows the node to exchange data with the selected device simultaneously. The device selection is obtained by specifying inside the request the index of the particular microcontroller pin to be used as the device chip select. These pins are defined in a C header file, which can easily be adapted to the specific hardware implementation. A similar customization is required to define GPIO pins, which can be configured as input, output or bidirectional ports. In this case, besides read and write operations, a dedicated command allows to switch a bidirectional pin between input and output modes.

Moreover, to give the platform a further degree of flexibility, the firmware provides a basic set of commands to let LabVIEW handle the microcontroller internal ADC. By editing a specific header file, it is possible to define the analog input pins to be used, as well as the ADC resolution and sampling frequency. The ADC works in free running mode, and cyclically converts the active channels. New samples are stored in a dedicated data buffer, which LabVIEW can read by means of a proper command. Each channel also features an optional first order IIR low–pass filter, whose activation and cut–off frequency can be controlled via software.

4.2.3 Software layer

As far as the LabVIEW application is concerned, the Communication layer has been developed using the LabVIEW built–in UART functions, which allows it to run also on those PCs that do not have true embedded serial ports, and therefore use the virtual ports provided by external adapters, such as the USB–to–UART converter included in Figure 4.2. The same solution can of course be applied as well to those microcontrollers which embed USB peripherals.

The Communication layer does not require any customization, and may be directly used to develop the control part of a simple data acquisition system in a short time. However, for complex systems the layer structure of Figure 4.3(a) may result more flexible and easy to maintain. Indeed, even if the Block API

and Device API layers depend on the hardware architecture and need to be customized for every implementation, they just require the development of a single function set per block type or device type. This means that if in the system there are several blocks or devices of the same type, neither the Block API nor the Device API layer need to be further extended. Hence, as happens for the hardware design re–use strategies, the user may build his own LabVIEW block and device libraries, and rely on them to speed up the development of the control application.

4.3   Layers interaction

To better describe how messages and information are exchanged between the different layers, it is worth referring to the DTS architecture of Figure 2.2. Comparing that diagram with the one of Figure 4.1, it can easily be observed that the Control block is the M–UART master node, whereas the Laser IF, the Optical Components Board, the Peltier Driver and the two Analog Front–ends are the slave nodes.

Each node features a simple 8 bit, 16 MHz Atmel ATmega164P microcontroller [27], which is configured to work at 250 kbit/s over the M–UART bus. Focusing for example on one of the two front–ends, its architecture can be re–arranged as depicted in Figure 4.4. Both the TMP275 temperature sensor and the two devices used to regulate the voltage gain, i.e. the AD5161 DAC and the AD5667R digital potentiometer, are connected on the I2C bus of the µC. For these devices no firmware customization is required. On the other hand, the devices used for the APD bias regulation, i.e. the MAX1932 APD supply and the ADG714 switch array, are controlled through the SPI interface. Their related chips select lines (CS_0, CS_1) have been defined in the µCheader file, along with the general purpose input line (GPIO_0) used by the MAX1932 to signal when the current limiter is active.

In Figure 4.5 the edited parts of the pin configuration file are reported. Each pin is identified by a structure which contains the addresses of its PIN, DDR, and PORT registers, as well as its number, access type and initialization value. In particular, the access type can be set to output, bidirectional, or input, in which case the initialization value represents the pull–up enable flag.

79
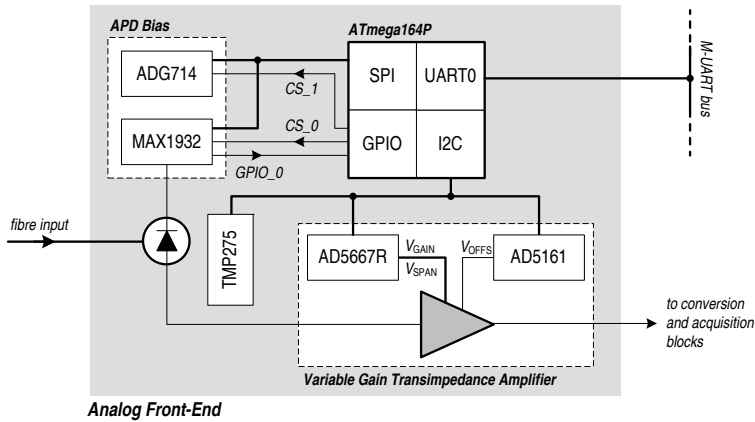
Figure 4.4: The front–end architecture

It can be observed that CS_0 and CS_1 lines have been assigned respectively to PA1 and PA0 pins, which are configured as outputs and initialized at the inactive high logical level. On the other hand, GPIO_0 has been assigned to PA2, which is instead configured as an input without pull–up. This way, CS_0 and CS_1 are directly handled by SPI transfer commands, whereas GPIO_0, being an input, can only be accessed through read operations.

It is also interesting to evaluate the maximum message rate allowed to Lab-VIEW. If we consider a typical SPI transfer of 2 B, the time required to send the related request from LabVIEW to the master node, or from LabVIEW to the slave node, is about 0.72 ms, including all the overheads due to the message

```
#define SPICS_NUM 2

// spiCSArray initialization values
// PIN, PORT, DDR, pin num, access type, init value / pull-up enable
#define SPICS_STRUCT_VALUES {\
    {PINA_ADDR, DDRA_ADDR, PORTA_ADDR, 1, OUTPUT, 1}, /* 0: PA1 -> /CS_0 */ \
    {PINA_ADDR, DDRA_ADDR, PORTA_ADDR, 0, OUTPUT, 1}  /* 1: PA0 -> /CS_1 */ \
    }
```

```
#define IOPINS_NUM 1

// PinArray initialization values
// PIN, PORT, DDR, pin num, access type, init value / pull-up enable
#define IOPINS_STRUCT_VALUES {\
    {PINA_ADDR, DDRA_ADDR, PORTA_ADDR, 2, INPUT,  0} /* 0: PA2 -> /CL   */ \
    }
```

Figure 4.5: The configuration file of the front–end µC

(a) Block API

(b) Device API

(c) Communication
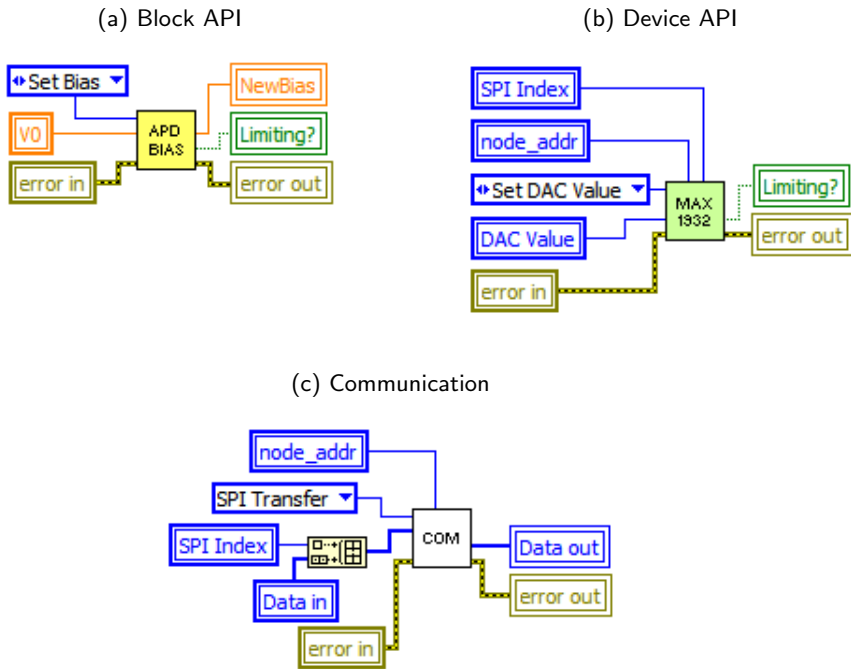
Figure 4.6: An example of LabVIEW functions

formatting and the M–UART protocol. This means that, in the worst case, a request needs about 1.44 ms to reach a slave node. If the processing time is neglected (order of few tens of microseconds for a 2 MHz SPI communication) the request–response round trip time is hence 2.88 ms, which corresponds to a rate of about 347 messages per second. This value allows slow control loops (such as for offset and gain regulation) to be implemented in LabVIEW rather than in firmware.

As far as the software is concerned, we developed all the LabVIEW functions required by the layer structure of Figure 4.3(a). In particular, to give an example of how the different layers interact, let us consider the case in which the main application wants to set a new value V_0 for the APD bias voltage. Following the sequence of Figure 4.3(b), at first the APD_Bias function of the Block API layer is called. In the LabVIEW environment, this task is carried out by the diagram of Figure 4.6(a). Then, according to the requested voltage

81

value, the function decides if the MAX1932 and/or the ADG714 devices need to be reconfigured, determines their new parameters, resolves the front–end node address and the device chip select indices, and calls the related Device API functions. Suppose, for simplicity's sake, that only the MAX1932 internal DAC register need to be updated. Hence, the MAX1932 function is called as represented in Figure 4.6(b). This function, according to the device protocol, prepares a proper byte sequence to be transferred to the device via SPI, and calls the send_msg function of the Communication layer. Figure 4.6(c) shows this last step. A request message is built and sent to the master node, which forwards it on the M–UART bus. The µC of the front–end receives the request and executes the SPI_transfer function, so that the V_0 voltage is definitely applied to the APD. Then, it prepares a response message, which passes back through the layers and returns to the application.

Finally, Figure 4.7 shows the LabVIEW front–end control panel which allows the user to control the APD bias voltage and temperature, the current of the Peltier cell, and the gain and offset parameters of the transimpedance amplifier.
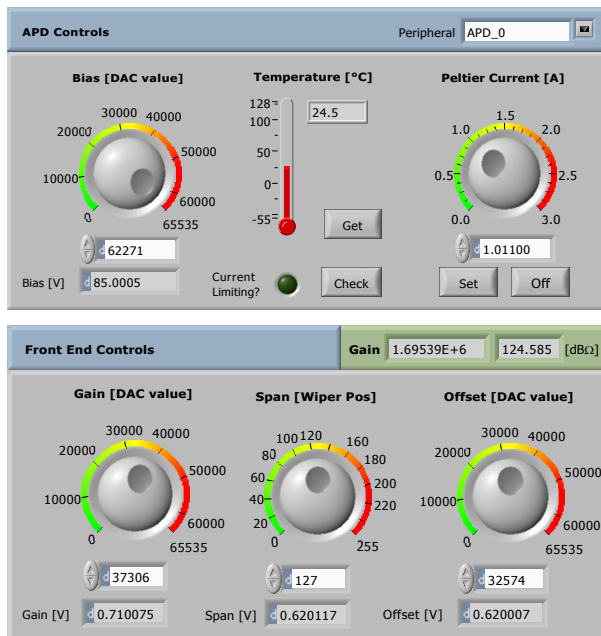
Figure 4.7: The LabVIEW front–end control panel

5

Experimental Results

5.1 APD biasing

Once implemented, the APD biasing circuit was tested using a high–precision digital multimeter (Agilent 34401A). Without mounting the APD, the circuit output was connected to the instrument, which was controlled by a PC via the GPIB interface. A specific LabVIEW program accesses the SPI peripheral of the front–end µC through the M–UART bus, and sweeps all the valid codewords (i.e. drops those with $\mu = 0$ and $\rho \neq 0$). For each codeword, it waits for 100 ms so that any transient can be safely considered ended, than it queries the multimeter for the output voltage. In particular, each voltage is read 100 times and saved to a database, so that averaging and other statistical processing can be performed off–line. The obtained results are reported in the following plots.

Figure 5.1 shows the transfer characteristic. It can be noted that the output voltage covers a range from 21 V to 101 V, as expected. More information can be derived by evaluating the non–ideal parameters of the circuit. To this end, in Figure 5.2 the differential non–linearity (DNL) error is reported as a function of the input codeword. To improve the readability of the collected data, in this plot as well as in the following ones, the invalid codewords have been dropped. It can be observed that DNL globally lies between $\pm 0.4\,\mathrm{LSB} = \pm 1.95\,\mathrm{mV}$, except for two peaks occurring at 0x1166 and 0x1B52 codewords. As these codes are far from the $\mu = 0$ condition at which the ADG713 changes its configuration, the two peaks are due to the MAX1932 internal DAC. In addition, as the first peak exceeds $-1\,\mathrm{LSB}$, for the 0x1166
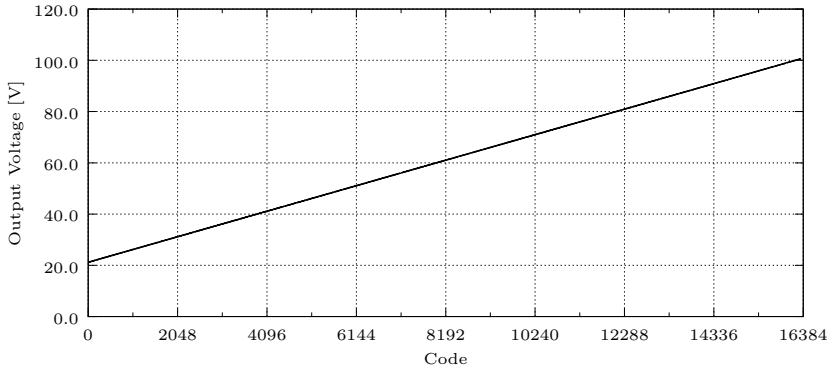
Figure 5.1: APD biasing circuit – Transfer characteristic
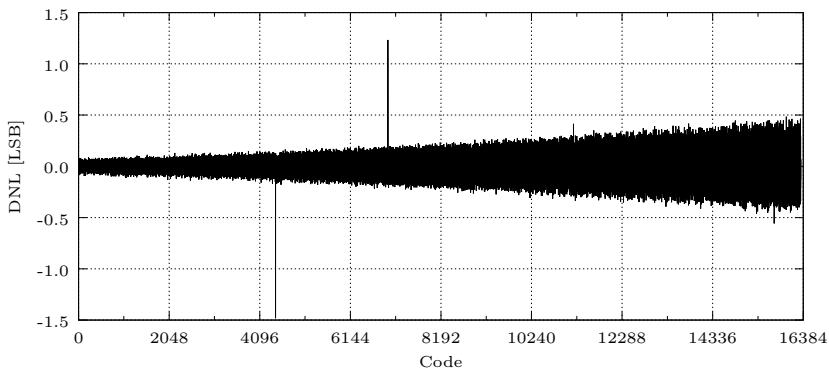
Figure 5.2: APD biasing circuit – DNL versus input code

codeword the characteristic is not monotonic. This aspect has been carefully taken into account when developing the LabVIEW application, so as to avoid stuck control loops.

Looking instead at the integral non–linearity (INL) error reported in Figure 5.3, it can be seen that it ranges between -2 and 3 LSB. The first two drops are due to the previous codewords affecting the DNL. Other peaks instead occur for $\mu = 0$, and hence are related to the ADG713 configuration change and to some mismatches between the different resistor arrays (see Section 3.1).
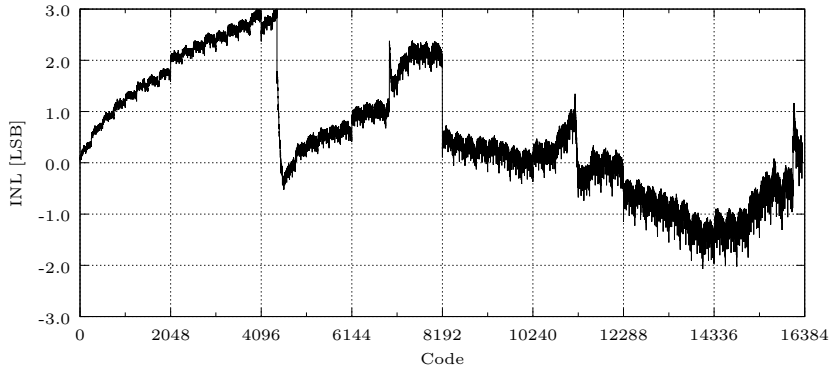
Figure 5.3: APD biasing circuit – INL versus input code
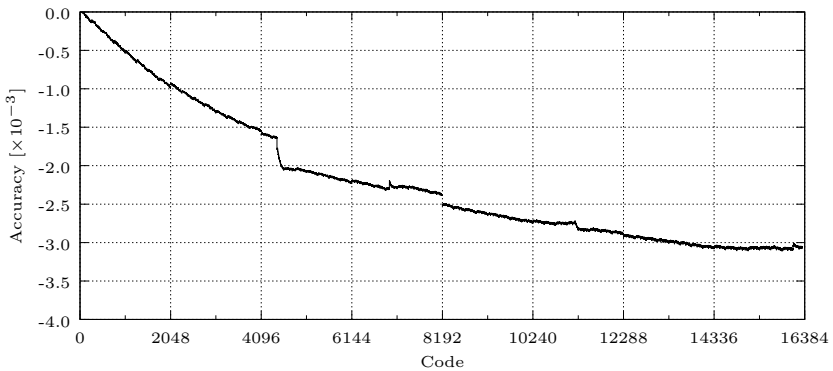
Figure 5.4: APD biasing circuit – Accuracy

Finally, Figure 5.4 reports the accuracy as a function of the codeword. As happens for INL, some peaks are present for 0x1166 and 0x1B52, and also when the ADG713 changes its configuration. It is worth noting however that, in the worst case, the absolute value of the accuracy is about 0.31 %, which is good agreement with the MAX1932 specifications.
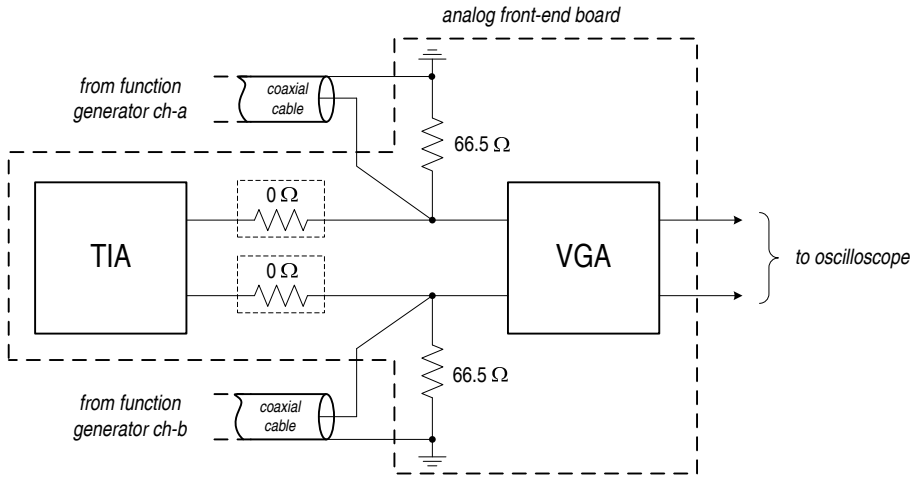
Figure 5.5: The analog front–end measurement setup

5.2   Amplification chain

To verify the performance of the amplification chain with respect to the simulated data reported in Section 3.2.2, the measurement setup reported in Figure 5.5 has been adopted. The output of a dual–channel arbitrary waveform generator (Tektronix AFG-3252) was connected to the VGA input through $50\,\Omega$ coaxial cables. This was achieved by removing the $0\,\Omega$ bridge resistors which link the TIA to the VGA, and by adding the $66.5\,\Omega$ resistors to match the input lines. In particular, these values were properly determined by analysing the AD8139 differential input impedance (see Section 3.2.2). The VGA differential output was finally connected to a multi–channel digital oscilloscope (Agilent MSO7104A) by means of $50\,\Omega$ matched lines.

In Figure 5.6 the measured frequency response of the VGA is reported and compared to the one obtained by means of simulations. It can be noted that the measured bandwidth is slightly lower than the simulated one. Recalling the diagram of Figure 3.3, this can be due to an actual reduced bandwidth of the first VGA stage. Moreover, the peak introduced in the response can be ascribed to the third stage, for which no feedback capacitors have been used. In general, the differences between simulated and measured curves are mainly due to the parasitic effects of PCB and passive components, which were not
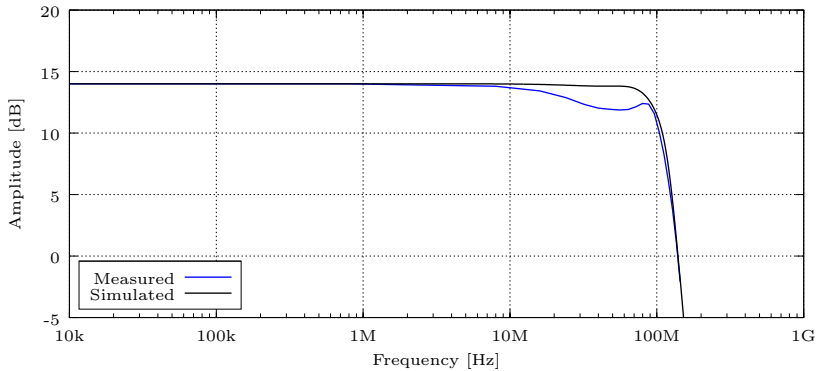
Figure 5.6: Frequency response of the analog front–end – simulated versus measured data

Table 5.1: Analog front–end dynamic parameters

Parameter	Simulated	Measured
-3 dB bandwidth	105 MHz	101 MHz
passband ripple	0 dB	−2.1 dB
rise time	3.2 ns	4.7 ns
settling time	5.7 ns	11.8 ns
overshoot	8.4 %	0 %

taken into account during simulations. Consider in fact that a parasitic capacitance of less than a picofarad can easily affect the performance of the system when the working frequency is in the order of 100 MHz or more. Nevertheless, the two curves can be considered in good agreement.

Figure 5.7 shows the comparison between the simulated and the measured step response. In this case, the waveform generator and the VGA were configured so that the input signals and the total gain matched the values used during simulations (see Section 3.2.2). As it could be expected from the previous plot, being the measured bandwidth lower than the simulated one, the measured step response is slower, and even if oscillating, it does not show any overshoot. The overall comparison results are resumed in Table 5.1.
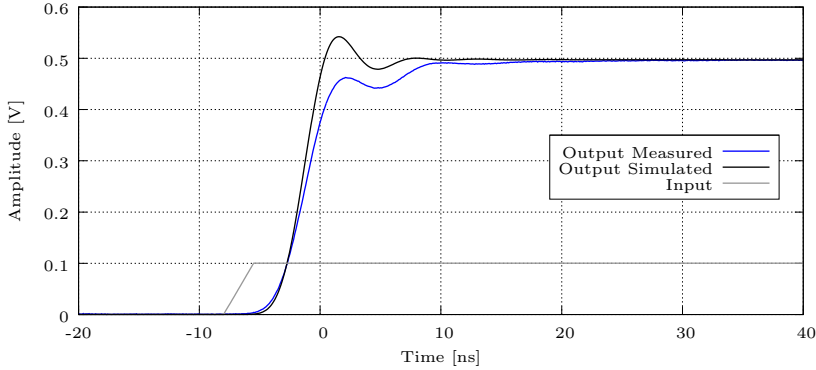
Figure 5.7: Step response of the analog front–end – simulated versus measured data

5.3   FPGA architecture

All the functionalities implemented on–board of the FPGA were extensively tested by means of an experimental setup comprising the two development boards, the waveform generator, the oscilloscope, and a PC running a proper LabVIEW test program built on top of the Application API layer.

A particular care was dedicated to the validation of the interleaved acquisition mode. Initially, the phase relationship between the master clock signal and the trigger output has been analysed for all the possible trigger delay values. For simplicity's sake, the blank signal has been taken as the reference for the phase measurement, since it is synchronous with the master clock and already available on external connectors. The various test steps are reported in Figure 5.8.

At first, the trigger delay parameter was set to 0, and both the trigger and blank signals were acquired by means of the oscilloscope. The instrument was configured to perform 256 averages, so as to reduce the unavoidable jitter. Then the acquired signals were deskwed, i.e. their delay was compensated to zero. The result of this operation is reported in Figure 5.8(a), in which the green trace is the blank reference signal, whereas the yellow trace is the trigger. Then the trigger delay parameter was set to $T_{\mathrm{CLK}}/4$, $2T_{\mathrm{CLK}}/4$ and $3T_{\mathrm{CLK}}/4$, respectively, and the actual delay was measured (see Figure 5.8(b)-5.8(d)). The obtained results are reported in Table 5.2. It can be noted that a very

(a) delay $= 0$
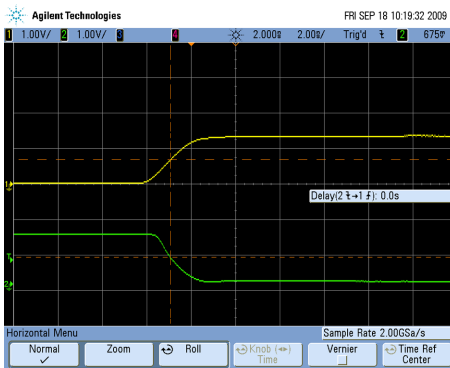
(b) delay $= T_{\mathrm{CLK}}/4$
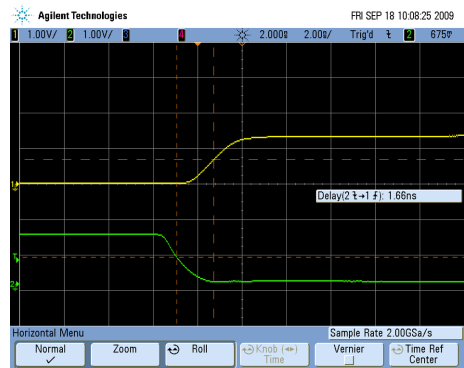
(c) delay $= 2T_{\mathrm{CLK}}/4$
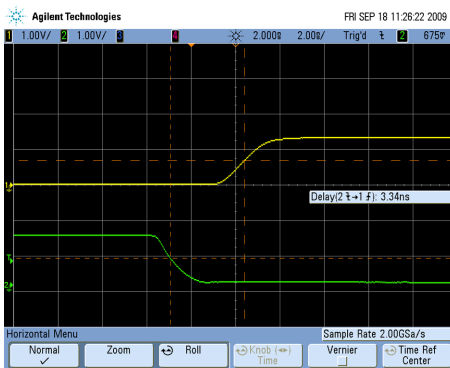
(d) delay $= 3T_{\mathrm{CLK}}/4$
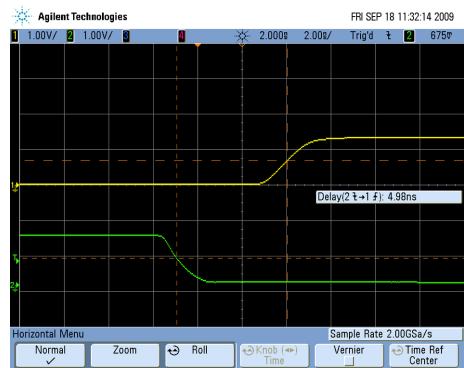
Figure 5.8: Trigger delay measurement – screenshots

Table 5.2: Trigger delay measurement – results

Delay	Nominal value	Actual value
0	0 ns	0 ns
$T_{\mathrm{CLK}}/4$	1.66 ns	1.66 ns
$2T_{\mathrm{CLK}}/4$	3.33 ns	3.34 ns
$3T_{\mathrm{CLK}}/4$	5 ns	4.98 ns
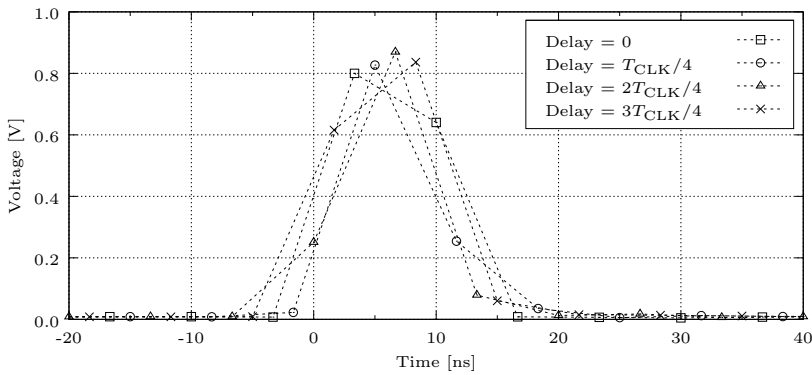
Figure 5.9: Interleaved acquisition – raw samples

high accuracy is achieved. Indeed, the greatest error occurs for $3T_{\mathrm{CLK}}/4$ and it is only 0.02 ns, i.e. the 0.4 % of the actual value.

Then, the waveform generator was connected to one of the two differential inputs of the ADC board, and configured to produce a 1.85 V voltage pulse, with a 2 V common mode, symmetrical rise and fall times of 2.5 ns and a duration of 10 ns, which is comparable to the 6.6 ns clock period. The trigger input of the generator was connected to the trigger output of the FPGA, which was configured to accumulate 10 traces at 14 bit resolution. The acquisition has been repeated with all the possible different delays. The averaged results are reported in Figure 5.9. It can be observed that the peak values are halved due to the inherent attenuation of the 50 Ω matched line.

The collected samples were finally interlaced via software to produce the reconstructed pulse shown in Figure 5.10. In the same plot the simulated pulse

response of the ADC driver stage is reported. Comparing the two signals, it can be concluded that the interleaved acquisition works as expected.

5.4 Temperature measurement

Once the system was assembled in an all–in–one 4U rack–mount case including the PC components, an extensive test campaign has been started at the CNIT laboratories, Scuola Superiore Sant'Anna (Pisa), to evaluate the measuring performance of the sensor. This characterization phase is still ongoing. As a consequence, only preliminary results are reported in the following.

The adopted test setup is shown in Figure 5.11. A 9.3 km fibre spool is connected to one of the four available optical ports of the instrument. The other end of the fibre is spliced to a 200 m spool, which is kept inside a temperature–controlled chamber (TCC). Then, by means of a second 9.3 km spool, the fibre is finally closed in loop to the instrument. This way, both the single–ended and the double–ended measurement configurations can be tested without changing the setup. Moreover, two additional 100 m fibre spools (not showed in the diagram) are embedded in the instrument case and used to link the 1×4 output switch to the optical ports. Their main purpose is however to allow for the system *self–compensation*. The internal case temperature is in fact measured by means of the integrated sensor mounted on the Control board
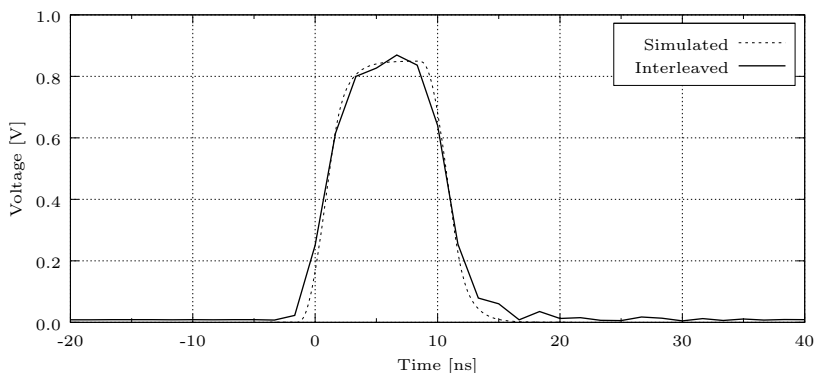
Figure 5.10: Interleaved acquisition – simulated versus interleaved pulse

(see Figure 2.2). By comparing this temperature with the one detected by the system in the very first 100 m fibre, it is possible to compensate for the thermal derating of optical components parameters, such as the laser peak power or the Raman filter insertion loss. Taking all the sections into account, the overall optical path is hence 20 km. However, the actual measurement range is $L = 18.8$ km. All the following plots refers to this range.

During all the reported tests, the FPGA is configured to acquire $N_0 = 1024$ traces of $S = 15$k samples with an 8 bit resolution. LabVIEW performs a further accumulation of FPGA data so as to reach $N = 98304$, corresponding to a 19.6 s acquisition time. The 9.3 km spools are kept at the room temperature of 24 °C. Figure 5.12 shows the averaged Stokes and anti–Stokes traces obtained in the single–ended configuration for three different values of the chamber temperature T_C, i.e. 0 °C, 24 °C and 50 °C. It can be noted that the amplitude of averaged signals is not occupying the full dynaminc range. This is expected, because the analog gain is tuned to the amplitude of the noisy raw traces, so as to avoid clipping. Looking instead at the zoomed TCC section, the effect of temperature variation can be observed. In particular, some discontinuities are present due to the splice connections. However, as they equally affect both Stokes and anti–Stokes wavelengths, their effect is cancelled when computing the AS–to–S ratio.

Using a reference trace $T_{ref}(z)$ previously acquired at 26.625 °C, the three different temperature profiles have been extracted by applying (1.20). The obtained results are reported in Figure 5.13. In the zoomed TCC section it can be observed that, in each case, T_C is well detected. In particular, the temperature resolution σ_T has been evaluated by computing the standard deviation of the $T_C = 24$ °C profile. As the noise affecting the system can be
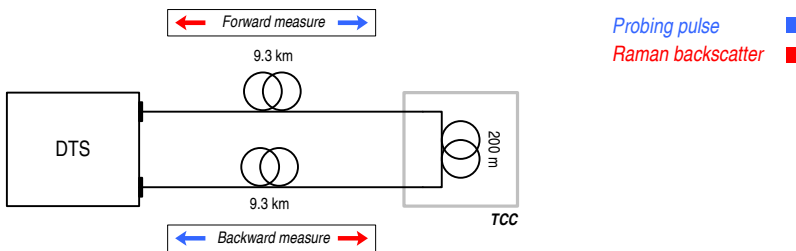
Figure 5.11: The temperature measurement setup
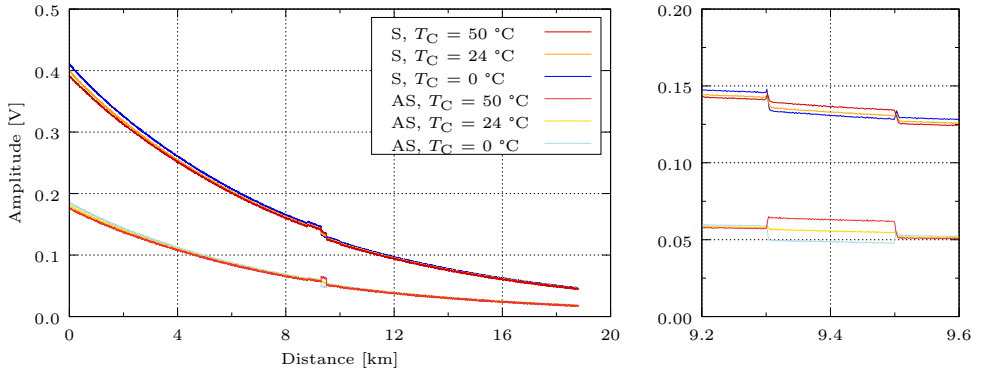
Figure 5.12: Backscattered Stokes and anti–Stokes traces for different temperatures – single–ended
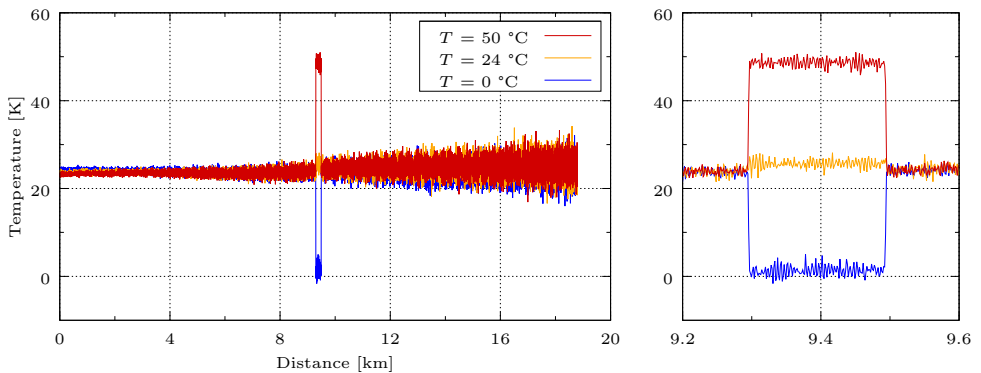
Figure 5.13: Temperature profiles – single–ended

considered as an ergodic process, this task has been carried out over time, using a moving window of 45 samples. This way, LabVIEW is not required to accumulate consecutive profiles, and can provide the result in real–time. The resolution is reported in Figure 5.14, along with an exponential fit curve to improve the plot readability. It can be noted that $2.5\,°C$ are achieved at the fibre–end, and about $1\,°C$ at $z = 10\,km$ which is a good preliminary result if compared to the state–of–the–art performance reported in Table 1.1.

In addition, the anti–Stokes signal–to–noise ratio on the raw traces has been evaluated at $z = 0$, obtaining $\mathrm{SNR_{AS}}(0) = 2 = 3\,dB$. This allowed for the validation of the approximated expression of σ_T derived in Section 2.4.2.

Indeed, by substituting this value in (2.32), it results $\sigma_{\mathrm{T}}(0) = 0.46\,°C$, which is in good accordance with the measured data.

The spatial resolution has been determined by analysing the rising step of the $T_{\mathrm{C}} = 50\,°C$ temperature profile. It followed that the $10\,\%$–to–$90\,\%$ temperature variation is covered in a distance of about $1.3\,m$. This also is a good result, as it is very close to those achieved by state–of–the–art DTS systems.

Finally, the results obtained for the double–ended configuration are reported in Figure 5.15 and 5.16. This time, the temperature profiles were obtained by applying (1.24) and by using a double–ended reference anti–Stokes to Stokes ratio previously acquired at the T_{ref} temperature. It is interesting to observe the effect of the backward propagating traces on the resolution. As expected, σ_{T} shows a concave shape, almost symmetrical around $z = L/2$. The slight asymmetry is due to the optical switch, which provides non–perfectly matched insertion loss values on all optical paths. If compared to the single–ended resolution, now σ_{T} is worse for $z < L/2$. Nevertheless, the maximum values of σ_{T}, which now occur at both fibre ends, are lower than the one obtained at $z = 19.8\,km$ in the single–ended configuration. This happens because in the double–ended configuration both fibre ends are used to launch the probing pulses.
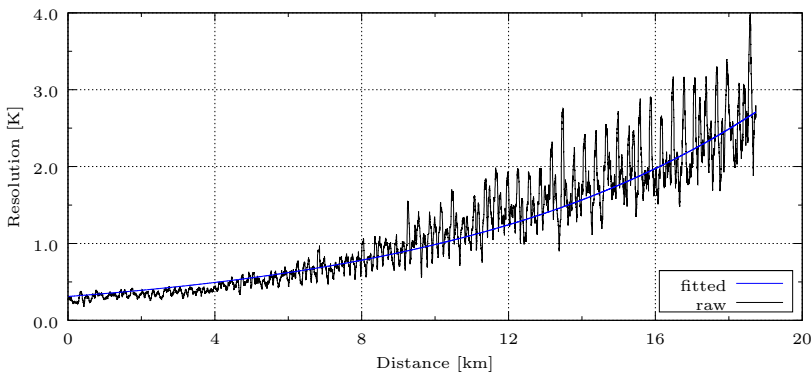
Figure 5.14: Temperature resolution – single–ended
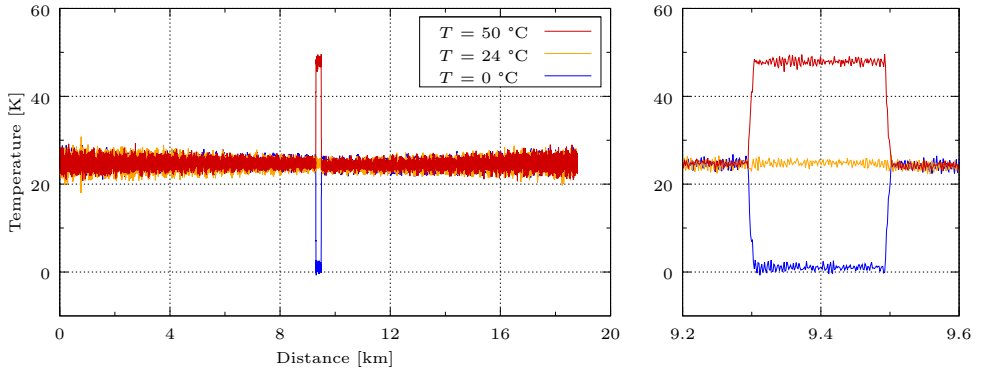
Figure 5.15: Temperature profiles – double–ended
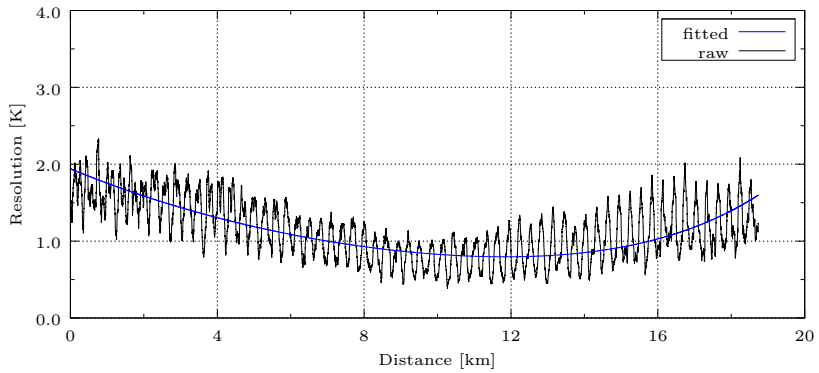
Figure 5.16: Temperature resolution – double–ended

6

SNR Enhancing Techniques

6.1 Averaging

As described in Chapter 1, one of the main issues to face in long–range Raman–based DTS design is the low SNR due to the weakness of the backscattered Stokes and anti–Stokes traces, which generally exhibit optical power levels lower than a few nW. This occurs because the energy of the probing laser pulse pulse cannot be freely increased. Indeed it is bounded by the targeted spatial resolution, which implies a small pulse width, and by the threshold for the onset of the fibre non–linearities, which upper bounds the pulse peak level. For this reason, to reduce the noise power introduced by the optical/electrical receiver, and reach a good temperature resolution at the fibre–end, trace averaging is typically employed.

It is well–known that, if N traces are averaged, the resulting SNR increases of a factor N. In practical applications, N is set according to the temperature resolution σ_{T} to be achieved at the fibre–end. In particular, for long–range measurements, N can easily reach very high values if a sub–degree resolution is desired at some tens of kilometres.

To give an example, the implemented prototype for $L = 30\,\mathrm{km}$ shows a $\mathrm{SNR}_{\mathrm{AS}}(L) = -16.96\,\mathrm{dB}$. If $\sigma_{\mathrm{T}}(L) = 1\,\mathrm{K}$ is desired, using (2.33) it results $N = 2.125 \times 10^6$. Considering also that the round–trip time T_{R} for a 30 km range is about 287 µs, this leads to a total acquisition time $T_{\mathrm{R}}N$ of about 610 s. Such a time may result not suitable for those killer applications in which a fast response is required. As a consequence, there's a great research interest

in developing different acquisition techniques which allow to improve the SNR without compromising the measurement time.

6.2 Pulse coding

6.2.1 Basic principles

Pulse coding is the typical solution adopted to address the issues of averaging. Basically, it consists in launching proper laser pulse *sequences* instead of a single pulse, so as to increase the probing energy without impairing the spatial resolution. These sequences are the optical representation of binary linear *algebraic codes*, which are widely used in communication theory for error detection and correction [60].

The various codes families are subdivided in *classes*, each of them containing codes of the same length. Once a code class of length M is selected, a *code set* of M codes is built. Then, each code of the set is launched, and its Stokes and anti–Stokes responses are acquired. Finally, the set of responses is decoded to obtain a couple of Stokes and anti–Stokes traces to be used for the temperature assessment. The most important aspect of pulse coding is that the SNR of decoded traces increases with M.

In DTS applications, coding techniques based on Simplex or Golay codes have been widely investigated [61–65]. In particular, it has been shown that for such applications Simplex codes provide the best performance in terms of *coding gain*, i.e. for a given M they allow to achieve the best SNR enhancement with respect to other coding schemes [26].

Simplex codes exist for any $M = 4n + 1$, with $n = 1, 2, \ldots$. In Figure 6.1 a qualitative example for $M = 3$ is reported. The code set is $\{[011], [101], [110]\}$. Whenever a laser pulse is launched, i.e. whenever the code–bit is 1, a new backscattered trace starts. This means that the response $R_c(t)$ to the code c acquired by the OTDR is given by the sum of some delayed replicas of the trace $\psi(t)$ to be recovered. The delay is a multiple of the chosen code–bit time
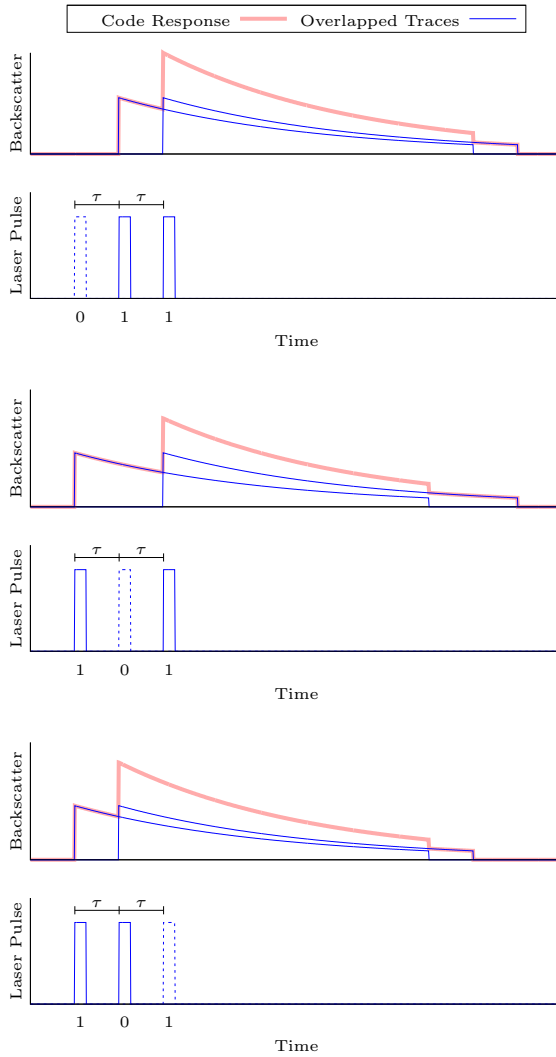
Figure 6.1: Conventional Simplex coding, example for $M = 3$

τ. In the reported example, the code responses are given by

$$\begin{cases} R_{011}(t) = \psi(t - \tau) + \psi(t - 2\tau) \\ R_{101}(t) = \psi(t) + \psi(t - 2\tau) \\ R_{110}(t) = \psi(t) + \psi(t - \tau) \end{cases} \qquad (6.1)$$

99

Rearranging (6.1) in matrix form, it follows

$$\begin{bmatrix} R_{011}(t) \\ R_{101}(t) \\ R_{110}(t) \end{bmatrix} = S \begin{bmatrix} \psi(t) \\ \psi(t-\tau) \\ \psi(t-2\tau) \end{bmatrix} \quad \text{with } S = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \tag{6.2}$$

The coefficient matrix S is called *Simplex matrix*. It it worth noting that each row and each column of S is a code, and that the set of rows or columns is the launched code set. $\psi(t)$ can be recovered by solving the system, thus obtaining

$$\begin{bmatrix} \psi(t) \\ \psi(t-\tau) \\ \psi(t-2\tau) \end{bmatrix} = S^{-1} \begin{bmatrix} R_{011}(t) \\ R_{101}(t) \\ R_{110}(t) \end{bmatrix} \quad \text{with } S^{-1} = \frac{1}{2} \begin{bmatrix} -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{bmatrix} \tag{6.3}$$

It is worth noting that all the coefficients of S^{-1} have the same absolute value. This property is strictly connected to the noise reduction. Indeed, if we indicate with σ_R^2 and σ_ψ^2 the variances of the acquired and recovered traces, respectively, from (6.3) it follows

$$\sigma_\psi^2 = \sigma_R^2 \sum_{j=1}^{M} \left| S_{ij}^{-1} \right|^2 = \frac{3}{4} \sigma_R^2 \quad \forall\, i = 1, \cdots, M \tag{6.4}$$

This means that for $M = 3$ the SNR coding gain C_G, defined as the ratio between σ_R^2 and σ_ψ^2, is 4/3.

The previous result can be extended to a generic $M \times M$ case. Indeed, S–matrices shows two important properties [66]:

- each row and each column contains $(M+1)/2$ "1"s and $(M-1)/2$ "0"s;

- two different rows or columns share $(M+1)/4$ "1"s in the same positions.

From these remarks it follows that

$$S^{-1} = \frac{2}{M+1} \left(2S^T - U \right) \tag{6.5}$$

where U is the $M \times M$ matrix of all "1"s. By substituting (6.5) in (6.4) and observing that $(2S^T - U)$ is a matrix of only "+1"s and "−1"s, it results

$$\sigma_\psi^2 = \sigma_R^2 \sum_{j=1}^{M} \left| S_{ij}^{-1} \right|^2 = \frac{4M}{(M+1)^2} \sigma_R^2 \quad \forall\, i = 1, \cdots, M \tag{6.6}$$

Therefore, for a given M the coding gain is

$$C_{\mathrm{G}} = \frac{\sigma_R^2}{\sigma_\psi^2} = \frac{(M+1)^2}{4M} \tag{6.7}$$

6.2.2 Measurement time reduction

The decoding algorithm represented by (6.3) actually allows to recover M different backscattered traces ψ delayed of τ multiples. Since τ is known, these decoded traces can be realigned and then averaged to further improve the gain of a factor M without affecting the measurement time. In addition, conventional averaging can also be combined with pulse coding. Indeed, each code can be launched N_{S} times, the acquired code responses can be averaged, and then decoded. Taking all these contributions into account, the total coding gain results

$$C_{\mathrm{G}}^{\mathrm{tot}} = \frac{(M+1)^2}{4M} \cdot M \cdot N_{\mathrm{S}} \tag{6.8}$$

Comparing this result with the one provided by simple averaging, it follows that the same SNR enhancement can be achieved with a lower number of averages. Indeed, if we require

$$\frac{(M+1)^2}{4M} \cdot M \cdot N_{\mathrm{S}} = N \tag{6.9}$$

it can be seen that, once N is given, it is immediately $N_{\mathrm{S}} < N$. However, this does not mean that the actual measurement time is automatically reduced. In fact, it can be verified that each code response to be acquired now lasts $T_{\mathrm{R}} + (M-1)\tau$. As a consequence, to reduce the measurement time the following relation has to be satisfied

$$[T_{\mathrm{R}} + (M-1)\tau] M N_{\mathrm{S}} < T_{\mathrm{R}} N \tag{6.10}$$

By replacing N with the expression given by (6.9), this relation simplifies to

$$M > \frac{T_{\mathrm{R}}}{T_{\mathrm{R}} - 4\tau} \tag{6.11}$$

and since M must be a positive number, it have also to be

$$\tau < \tau_{\mathrm{MAX}} = \frac{T_{\mathrm{R}}}{4} \tag{6.12}$$
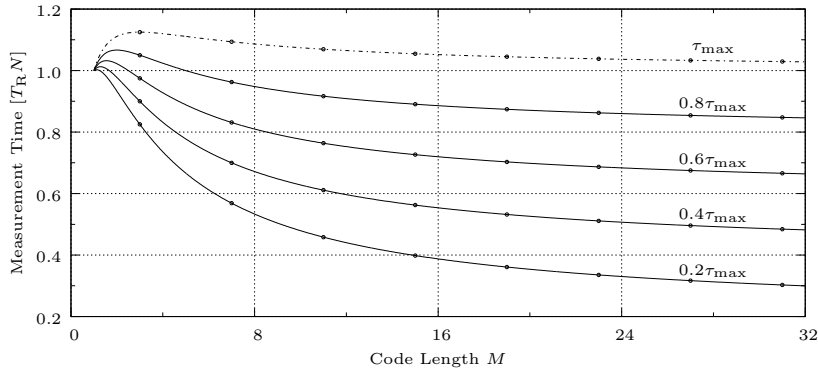
Figure 6.2: The measurement time reduction achievable with pulse coding

In conclusion, the pulse coding technique allows for the reduction of the measuring time only if (6.11) and (6.11) are verified. Figure 6.2 shows how the measuring time varies with M for different values of τ. The vertical axis is normalized to the time $T_{\mathrm{R}}N$ required by conventional averaging, so that the actual reduction can be appreciated. The curve for $\tau = \tau_{\mathrm{MAX}}$ is actually unreachable, and it is plotted only to show the theoretical limit.

Finally, it is important to point out that the choice of M and τ is bounded by the hardware. Indeed, as M increases, more memory resources are required to store the whole response set to be decoded. Hence, the maximum M is limited by the available RAM. On the other hand, the minimum value of τ is determined by the performance of the adopted laser source. A too small τ may in fact lead to a degradation of the pulse width, shape and peak [32].

6.3   Multi–pulse technique

6.3.1   Overview

In long–range systems, conventional pulse coding can be practically implemented only for small M values, as a large amount of memory resources required to store the response set is usually unavailable. In addition, as the number of acquired responses increases, the time required for their decoding
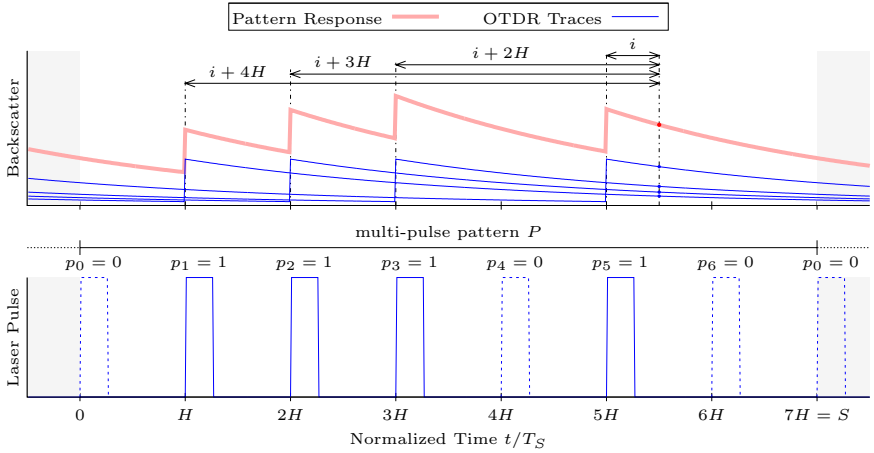
Figure 6.3: Multi–pulse technique, example for $M = 7$ and $P = \{0, 1, 1, 1, 0, 1, 0\}$

may result non–negligible, and hence efficient software algorithms are needed to handle such a great amount of data. To overcome these drawbacks, a new coding technique, named *multi–pulse* technique, has been developed [67].

First of all, suppose to set the acquisition length to T_R, and divide T_R into M equal time intervals. Then, consider a M–bit binary pattern $P = \{p_0, \ldots, p_{M-1}\}$, with $p_j = 0, 1$ for $j = 0, \ldots, M - 1$. Suppose now to launch p_j at the beginning of the j–th interval, and to repeat the pattern P periodically. Figure 6.3 shows what happens for $M = 7$ and $P = \{0, 1, 1, 1, 0, 1, 0\}$. In this plot, the time axis has been normalized to the system sampling period T_S. Hence, $S = T_\mathrm{R}/T_S$ are the total response samples and H are the samples contained in each M interval, so that $S = MH$.

Now suppose to acquire a single pattern response R_P. The acquired samples can be indicated with $R_P[i + jH]$, in which the index $j = 0, \ldots, M - 1$ scans intervals, and $i = 0, \ldots, H - 1$ scans samples within the j–th interval. If the same representation is adopted for the samples $\psi[i + jH]$ of the trace to be recovered, it can be verified that

$$R_P[i + jH] = \sum_{k=0}^{M-1} p_{|j-k|_M} \cdot \psi[i + kH] \tag{6.13}$$

Considering for example the highlighted samples in Figure 6.3, this equation says that the i–th acquired sample in the j–th interval is the sum of the

103

contribution of the ψ trace produced by the launch of p_j, and the contributions of the other ψ traces produced by the pulses launched in the $M - 1$ previous intervals.

If i is fixed and j is varied from 0 to $M - 1$, equation 6.13 produces the following $M \times M$ linear system

$$
\begin{bmatrix}
R_P[i + 0H] \\
R_P[i + 1H] \\
\vdots \\
R_P[i + (M - 1)H]
\end{bmatrix}
=
\begin{bmatrix}
p_0 & p_{M-1} & \cdots & p_1 \\
p_1 & p_0 & \cdots & p_2 \\
\vdots & \ddots & \ddots & \vdots \\
p_{M-1} & \cdots & p_1 & p_0
\end{bmatrix}
\begin{bmatrix}
\psi[i + 0H] \\
\psi[i + 1H] \\
\vdots \\
\psi[i + (M - 1)H]
\end{bmatrix}
\tag{6.14}
$$

Hence, to recover all the ψ samples, H systems (one for each i) sharing the same coefficient matrix S_C have to be solved. In particular, it is important to observe that S_C is cyclic by construction, i.e. any row is the right–shifted copy of the previous one. This means that to reduce the noise of the recovered samples, P can be appropriately chosen so that the rows of S_C are *cyclic Simplex codes* [60].

Cyclic Simplex codes are a subclass of Simplex codes, and can be built using the methods reported in [68]. They exist for $M = 4n + 1$, with $n = 1, 2, \ldots$, and provide the same gain C_G of conventional Simplex. As a consequence, if we indicate with σ_R^2 and σ_ψ^2 the variances of any R_P and ψ sample, respectively, the same result of (6.7) is achieved.

6.3.2 Performance comparison

As happens for conventional coding, the multi–pulse technique can be combined with averaging. In this case, to obtain the same total gain of (6.8), the pattern response has to be acquired MN_S times. However, the important difference is that the length of the response is now fixed to the round–trip time T_R, and it does not depend on either M or τ. As a consequence, the measurement time is $T_R M N_S$. If compared to simple averaging, using (6.9) it follows that

$$
T_R M N_S < T_R N = T_R M N_S \frac{(M + 1)^2}{M} \quad \forall M
\tag{6.15}
$$
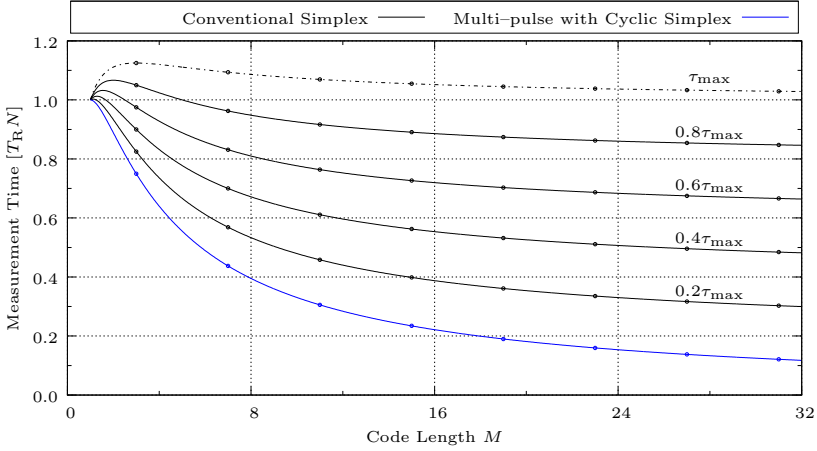
Figure 6.4: Comparison between the measurement time reduction achievable with multi–pulse technique and with conventional coding

This means that the time reduction occurs for any M. In addition, if compared to conventional coding, the result is even more interesting. It is in fact

$$T_{\mathrm{R}} M N_{\mathrm{S}} < [T_{\mathrm{R}} + (M-1)\,\tau]\,M N_{\mathrm{S}} = \qquad (6.16)$$
$$T_{\mathrm{R}} M N_{\mathrm{S}} + (M-1)\,\tau M N_{\mathrm{S}} \quad \forall\ M, \tau$$

In other words, the measurement time reduction is better than the one provided by conventional coding for any combination of M and τ. This result is plotted in Figure 6.4.

The only limitation of the multi–pulse technique is given by the laser. Indeed, if $\tau_{\min}$ is the minimum pulse repetition period, it has to be $M < T_{\mathrm{R}}/\tau_{\min}$. However, high–power lasers with repetition rates up to some tens of kilohertz, i.e. with $\tau_{\min}$ of a few microseconds, are available on the market, and allows to implement good code lengths.

To give an example, the laser adopted in the prototype features a maximum repetition rate of $300\,\mathrm{kHz}$, that is $\tau_{\min} = 3.3\,\mu\mathrm{s}$. For a $30\,\mathrm{km}$ range, it is $T_{\mathrm{R}} = 287\,\mu\mathrm{s}$. As shown in Section 6.1, to reach $1\,\mathrm{K}$ resolution at the fibre–end, $N = 2.125 \times 10^6$ averages are required. Using the multi–pulse technique, it must be $M < 86.1$. Choosing for instance $M = 83$, from (6.9) it follows $N_S = 1205$. This means that the measurement time, which is given by $T_{\mathrm{R}} M N_S$, is

105

about 28.7 s. If compared to the 610 s required for conventional averaging, it is about the 95 % less.

To obtain the same outstanding measurement time with conventional coding, it should be $M = 1783$ and $N_S = 3$. Such a great value of M would lead to long code responses of about 6.2 ms, which should be acquired and then all together decoded, with a huge consumption of memory resources. For this reason, such a performance would be unachievable in practical applications.

6.3.3 Implementation and results

One of the great advantages of the multi–pulse technique is related to the implementation. Indeed, as showed in the previous example, no additional memory resources are needed with respect to a conventional DTS, because the pattern responses can be acquired and averaged by the system as it were normal uncoded traces. The decoding algorithm can easily be implemented in software, and its execution can be optimized in speed (for example using threaded or pipelined architectures) as the different H systems given by (6.14) can be solved in parallel.

Moreover, the multi–pulse pattern can be obtained by triggering the laser at the fixed rate $1/\tau$ and by implementing the cyclic code through an external modulator (e.g. an acousto–optic one), which allows to filter out a pulse if the corresponding bit code is equal to 0. This way, the pulsed laser operates at a constant frequency, which guarantees a good repeatability of the generated pulse shape and peak power [32].

Figure 6.5 shows the bench test architecture that has been implemented to test the multi–pattern technique. If compared to the one reported in Figure 2.1, only the blocks highlighted in grey have been changed or added. In particular, the Trigger Generator (see Section 3.4.5) has been extended to produce the acusto–optic modulator (AOM) control signal. When a proper multi–pulse enabling flag is set, it disable the blank signals, starts triggering the laser every H clock cycles, and before any trigger pulse, it properly turns the AOM on or off according to the pattern P. Both H and P are stored in the register bank and can be uploaded by LabVIEW via SPI. The AOM control signal has been assigned to one of the debug pins available on the FPGA
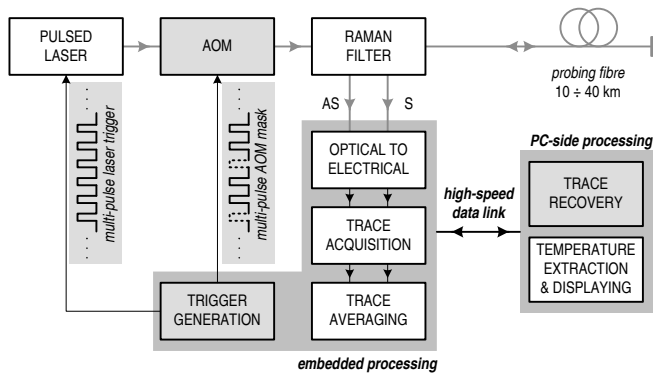
Figure 6.5: Multi–pulse enabled PC–based DTS architecture

board (see Figure 3.9) and routed to the modulator through a $50\,\Omega$ cable.

The Trace Recovery block has been implemented in LabVIEW according to (6.13). A dedicated interface was also developed to control the new FPGA parameters related to the multi–pulse technique. Bench tests were carried out using a $10\,\mathrm{km}$ fibre kept at room temperature. In this case, it is $T_{\mathrm{R}} = 95.67\,\mathrm{\mu s}$. Being $\tau_{\min} = 3.3\,\mathrm{\mu s}$, it has to be $M < 28.9$. As a consequence, we tested $M = 3, 7, 11, 15, 19, 23$. For each M, the FPGA was configured to acquire $N_S = 1024$ pattern responses with $8\,\mathrm{bit}$ resolution. For simplicity's sake, only the anti–Stokes traces for $M = 3, 11, 23$ are reported in the following Figure 6.3.3, 6.3.3 and 6.3.3, respectively.

The first thing to note is the presence of unwanted vertical peaks on the raw traces, which inevitably affect also the decoded traces, even if in a reduced way. They are due to a conducted EMI between the trigger pulse and the analog front–end input. In the future, this problem will be addressed by reviewing the integrity of all the coaxial connections and checking that no ground loops have been accidentally formed. Nevertheless, the effect of the coding gain can be appreciated. In figure captions are reported the values of the standard deviation measured on both raw and decoded data after 100 acquisitions. It can be observed that the actual code gain C_{G}^0 is very close to the theoretical gain C_{G} given by (6.7).

107
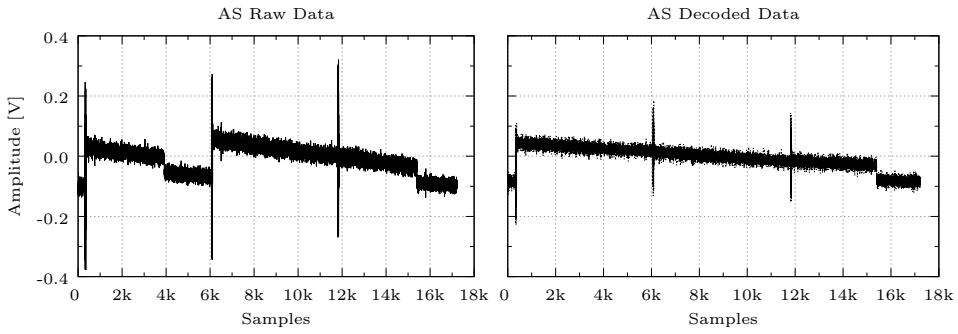
Figure 6.6: Multi–pulse technique – anti–Stokes traces for $M = 3$
$\sigma_{\mathrm{raw}} = 13.437\,\mathrm{mV}$, $\sigma_{\mathrm{dec}} = 11.375\,\mathrm{mV}$, $C_{\mathrm{G}}^0 = 1.1813$, $C_{\mathrm{G}} = 1.1547$
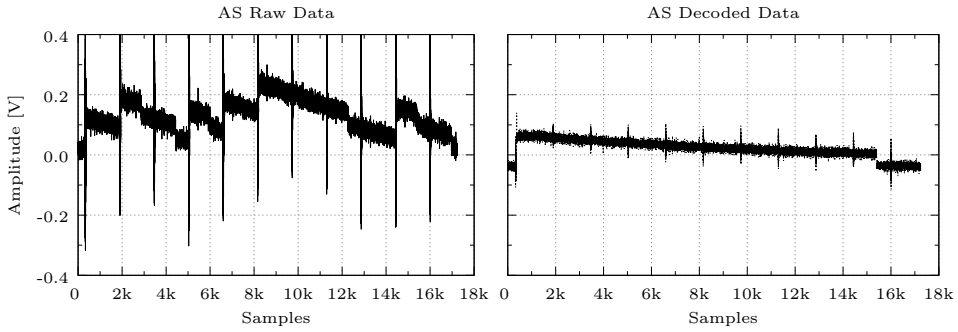
Figure 6.7: Multi–pulse technique – anti–Stokes traces for $M = 11$
$\sigma_{\mathrm{raw}} = 15.386\,\mathrm{mV}$, $\sigma_{\mathrm{dec}} = 8.237\,\mathrm{mV}$, $C_{\mathrm{G}}^0 = 1.8680$, $C_{\mathrm{G}} = 1.8091$
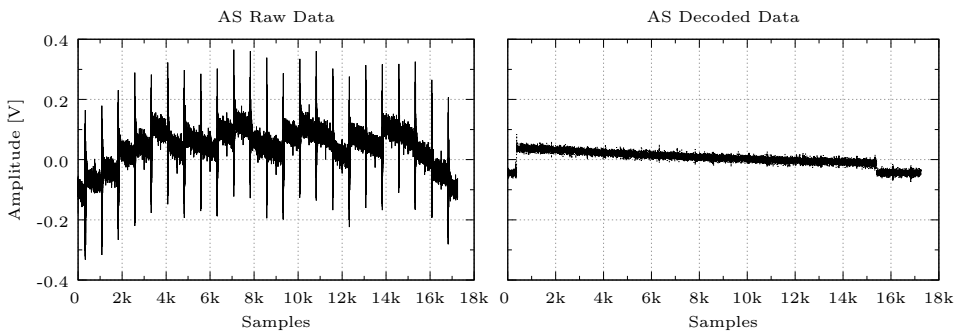
Figure 6.8: Multi–pulse technique – anti–Stokes traces for $M = 23$
$\sigma_{\mathrm{raw}} = 16.911\,\mathrm{mV}$, $\sigma_{\mathrm{dec}} = 6.632\,\mathrm{mV}$, $C_{\mathrm{G}}^0 = 2.5498$, $C_{\mathrm{G}} = 2.5022$

Part II

Design, Implementation and Test of Automotive Embedded Systems for a Formula SAE race car

7 Formula SAE

7.1 Introduction

Formula SAE is a student design competition organized by the International Society of Automotive Engineers. It started in 1979 in the US and thanks to its success and to the enthusiasm showed by participants, it rapidly spread all over the world. Nowadays, there are official competitions in Europe, Australia, Japan and Brazil, for a total of 10 events per year.

The concept behind Formula SAE is that a fictional manufacturing company has contracted a design team to develop a small Formula–style race car, targeted for the non–professional weekend autocross racer. Each university student team is asked to design, build and drive a prototype following a precise set of rules, whose purpose is to ensure the safety of both the drivers and all the participants, as well as to improve their design skills and the problem solving capabilities, not only in technical disciplines, but also in marketing, management and finance.

From a technical point of view, rules are very open, and leave a great freedom to designers. The only important restrictions are those related to the engine, whose displacement has to be not greater than $610\,\mathrm{cm^3}$. A $20\,\mathrm{mm}$ air restrictor must also be fitted downstream of the throttle and upstream of any compressor. This keeps power levels typically below 100 hp.

Every competition starts with a preliminary technical inspection. Each car is examined to test if it is built in the full respect of the rules. Specific

test are also carried out to check the rollover stability (at the tilt table), the break efficiency and the engine noise. Then, the actual contest takes place. It is organized in 3 static and 5 dynamic events, which are judged by the most prominent engineers, administrators and consultants of the automotive industry.

Static events include the design, cost analysis and business plan presentations. Here, all the proposed innovative solutions and the adopted trade–offs between cost and performance are analysed by judges, and the most interesting ones achieve the highest score. On the other hand, in dynamic events the engines run. Basically, dynamic events are time trials. Cars challenge each other in acceleration, skidpad and autocross races, and the faster ones get the higher score. In particular, the autocross results are used to determine the start order for the final endurance event. This is the most important and difficult event of the competition. It consists in a 20–22 km race, divided into two rounds performed by different drivers. In such a trial, all the car subsystems are deeply stressed, and it is frequent to see cars giving up the race for technical problems. At the end of the endurance, the fuel efficiency is eventually evaluated. Finally, all the scores collected in both static and dynamic events are added, and the final classification is determined.

7.2 E–Team Squadra Corse

E–Team Squadra Corse is the Formula SAE team of the University of Pisa (Italy) [69]. It was formed in the late 2007 involving 32 people among undergraduate and PhD students of engineering disciplines (such as mechanics, vehicular technology, electronics, computer science), as well as of economics and other arts faculties.

The first participation to Formula SAE competitions was in 2008, at the Formula SAE Italy event at the Ferrari's track in Fiorano. The first car, the ET1, was developed in just nine months of work, and it was one of the 14 cars over 31 to complete the endurance, which was a very good result for a rookie team. Of course, the performance were not at the top, there were a good room for improvement, but a good level of reliability was achieved.

In 2009 the team grew up to 42 members by including students in aerospace engineering. The second car, the ET2ev, showed an improved performance, which was achieved by optimizing each ET1 subsystem. In particular, the overall weight was significantly reduced without affecting the reliability. The ET2ev took part to Formula Student Germany at Hockenheim, and also to Formula SAE Italy at Varano de' Melegari, where it obtained very good results: second Italian team, 9th place in the design event and 5th in the cost event.

In 2010, the team reached 44 members. The ET2ev was completely re–designed to further push the weight optimization. To develop the third car, the ET3, carbon fibre was used for the first time. At the Formula SAE Italy event, the team reached the best result ever, defeating all other Italian teams and cutting the finish line at the 9th place overall.

7.3 Why students should build race cars

Sometimes it happens that the activities related to Formula SAE teams are unfortunately considered as a time–consuming, money–wasting, unproductive game, carried out just for fun. Actually, things are deeply different. The fun, which of course is present, is the glue element which allows voluntary students to have an unparalleled formative experience, not only from a theoretical, but, above all, from a practical and multidisciplinary point of view. All these aspects are well summarized by the following text, which appeared on a poster during the Formula Student Germany event in 2010.

Why students should build race cars

Experiences with teamwork, time and project management in general along with construction, manufacturing and the economical aspects of automotive engineering in particular, greatly improve the qualifications of young engineers. In addition, heaving English as the official language of the competition contributes significantly to the improvement of the students foreign language skills.

Formula Student Germany enhances the visibility of the participants' commitment to engineering and thereby increases job placement opportunities at German companies and Universities. Both, sponsors of the competition and of the individual teams are able to build valuable contacts with potential employees.

How businesses and sponsors benefit from the competition

The motorsport, automotive and supplier industry all need qualified young engineers to preserve their quality standards. Formula Student Germany helps to offer a clear indication of the quality education of engineering students and provides an excellent basis for contacts. Companies, sponsoring the event, providing awards, and supplying judges, get the possibility to be convinced by the know–how and skills of their potential future employees.

Why universities should support their constructing engineers

The recognition by the industry of formula Student activities as additional qualification also affects the universities. Excellent graduates improve the reputation of the universities, which are therefore in return interested in supporting the students with knowledge and money.

Moreover, the overall positive perception at universities leads to the formation of new teams, which reflect a positive image of engineering in general. Thus, Formula Student Germany also enhances the attractiveness of engineering studies.

Formula Student Germany, Hockenheimring, Aug. 2010

8

The ET1 project

8.1 Shift–by–wire

The ET1, which is shown in Figure 8.1, was equipped with an Aprilia 4–stroke V–twin–cylinder RXV 550 motorcycle engine. The main task assigned to the electronic team was to robotize the original gear and clutch levers of the engine, so as to allow the driver to perform shift and start procedures by just pulling flipper–paddles mounted behind the steering–wheel. In other words, a shift–by–wire system was required.

Figure 8.1: The ET1 during the 2008 Formula SAE Italy event at the Fiorano race–track

The shift–by–wire approach is acknowledged as a valid solution to improving the performance and reliability of both race cars and commercial vehicles. The electronic control of gear shifting has important benefits in terms of a reduction in shift time, transmission chain efficiency, and savings in fuel consumption [70–73]. In fact, the conventional mechanical link from shift lever and clutch pedal to gearbox is replaced with sensors, electronic control units and actuators, which make the gearbox robotized. Thus, shift–by–wire can be considered as an effective application of X–by–Wire, a very promising automotive technology which can also be extended to other vehicle functions, such as steering, braking, and throttle control [74]. Replacing the classic hydraulic or pneumatic actuators with electrical ones controlled by an appropriate DSP unit may lead to great improvements in all these automotive sub–systems. The DSP unit can also be connected to the vehicle dashboard for human–machine interfacing and, through feedback sensors, to the controlled sub–system.

Steer–by–wire, brake–by–wire and active suspension systems are much more critical than shift–by–wire systems. This happens because they have a dramatic impact on vehicle behaviour and thus on its safety [75, 76]. As a consequence, fault–tolerant architectures to meet the very high demanding safety requirements of these X–by–Wire applications are currently being investigated in depth [77, 78]. On the other hand, the research focus on shift–by–wire systems is on improving their performance, given that appropriate architectures for their implementation are already available. This aim can mainly be achieved through an effective selection and control of the actuators employed [70, 74, 79–83].

Generally speaking, DC servomotors are the most commonly used X–by–Wire actuators, because they are easy to control in terms of speed, position and torque [84]. Nevertheless, some interest has also been shown in linear electromagnetic actuators [70, 74, 79, 82], and particularly in Voice Coil Actuators (VCAs), which are able to provide a wide range of forces (from less than 1 N up to 2000 N) together with very low inertia [85–89].

Linear VCAs basically consist of a permanent–magnet cylindrical frame and a coaxial moving copper coil. Their working principle is based on the Lorentz force. According to the amplitude and the direction of the current flowing in the windings, the coil is subject to a coaxial force, which enables it to push or pull a mechanical load. VCAs can easily be scaled and adapted to different applications, in both industrial and scientific fields, because the

force provided depends on their geometrical parameters. Thanks to their high reliability, small VCAs are commonly mounted inside cell phone cameras for auto–focusing functions [90, 91], or inside hard disk drives to move head arms [92–94], whereas larger ones are used to move loudspeaker cones (which is where their name comes from) [95], for laboratory shock and vibration testing [96], or generally in robotics and factory automation [97].

The shift–by–wire system designed for the ET1 is based on VCAs. To the best of our knowledge, this is the first time that such actuators have been employed in a race car transmission system. In fact, the gear and clutch automation of Formula SAE vehicles is usually achieved by pneumatic actuators, which are lighter and smaller than VCAs. However, the use of VCAs – as will be shown later – may lead to several significant advantages. In fact, the complete assembly of a pneumatic system requires the use of valves, air tanks, and even compressors if an unlimited endurance is required. All these components may lead to a higher overall weight when compared to a more compact VCA–based system, which just comprises a power supply in addition to the actuators. Moreover, a purely electric solution is more flexible and responsive than a pneumatic counterpart, and hence more suitable for implementing advanced control strategies, which may result in a higher performance [98].

8.2 Gear and clutch characteristics

To independently actuate the clutch and gearbox levers, two VCAs are required. The first VCA is used to pull the lever of the clutch so as to disengage it. The other is mounted so that its idle position is at midstroke and it is used to either push or pull the gear lever, so as to scan the 1-N-2-3-4-5 gear sequence, where N is Neutral gear. To choose the two VCAs in the most appropriate way, the force–displacement characteristics of the two levers need to be measured. The experimental results are reported in Figure 8.2 and Figure 8.3, respectively.

The first plot shows that a 300 N actuator with a stroke greater than 10 mm is needed to fully disengage the clutch. Instead the second plot refers to the 1–N–2 upshifting sequence. Note that Neutral is a sort of special gear which falls between 1st and 2nd gears, as shown by the two 180 N peaks on the gear
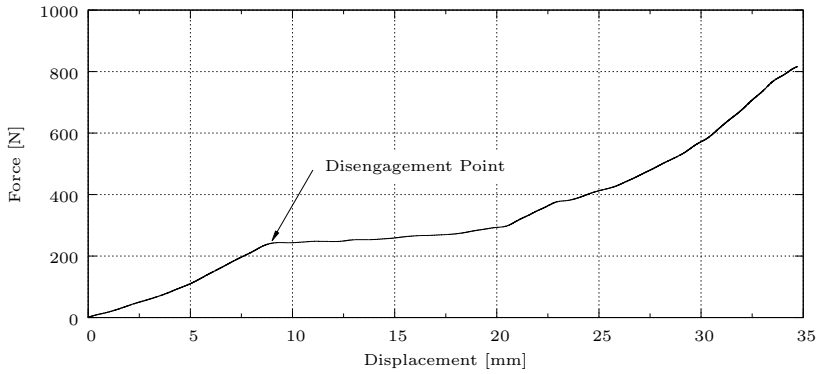
Figure 8.2: Clutch lever measurement – force vs displacement

lever characteristic. Neutral is shifted in at about 6 mm, and 2nd gear at about 10 mm, which is also the displacement required by any other upshift or downshift operation. As a consequence, to completely push and pull the gear lever from its idle position, a 20 mm actuator is required.
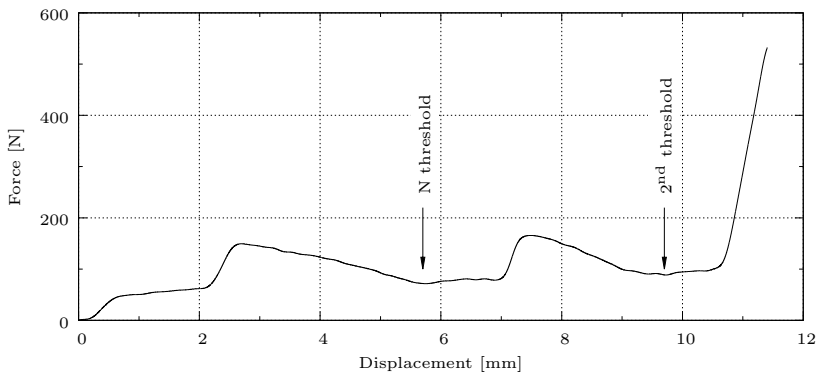
Figure 8.3: Gear lever measurement – force vs displacement

8.3 Actuator modelling

8.3.1 Dynamic equations

The basic operating principle of voice coil actuators is ruled by the Lorentz force. The current I flowing in the coil produces the force F_{VCA} given by (8.1), in which the force sensitivity K_{F} of the actuator depends on the position x of the cylinder along the axis

$$F_{\mathrm{VCA}} = K_{\mathrm{F}}(x)I \tag{8.1}$$

From an electrical point of view, a VCA can be represented by the equivalent circuit of Figure 8.4(a) in which E is the external supply, R and L are the resistance and the inductance of the coil, respectively, and $v \cdot K_{\mathrm{B}}$ is the back–electromotive force (BEMF) induced by the displacement of the cylinder. In particular, $v = dx/dt$ is the cylinder speed, and K_{B} is the BEMF constant. The equation that describes the circuit is

$$V_{\mathrm{S}} = RI + K_{\mathrm{B}}\frac{dx}{dt} + L\frac{dI}{dt} \tag{8.2}$$

Equations 8.1 and 8.2 represent the two fundamental relationships of linear VCAs. To analyse the dynamic behaviour of the actuator, the Newton's

(a) Electrical (b) Thermal
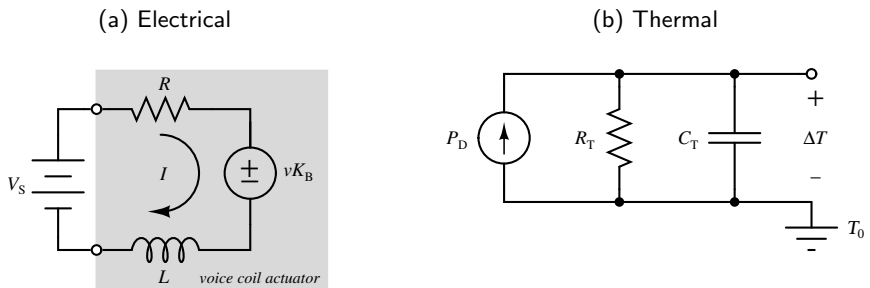
Figure 8.4: Equivalent circuits for the voice coil actuator

second law can be applied to the cylinder, thus resulting

$$F_{\text{VCA}} - F_{\text{EXT}} = m\frac{d^2x}{dt^2} \tag{8.3}$$

where m is the cylinder mass and F_{EXT} is the external force along the x axis due to the mechanical load connected to the actuator (e.g. the clutch spring, the gear lever, etc.). Combining this equation with (8.1), the following relation is finally obtained

$$K_{\text{F}}(x)I - F_{\text{EXT}} = m\frac{d^2x}{dt^2} \tag{8.4}$$

8.3.2 Thermal equations

From a thermal point of view, the Joule effect is the principal cause of the coil heating, which leads to a consequent resistance variation. In particular, the value of R can be expressed by the following equation

$$R(\Delta T) = R_0(1 + \alpha\Delta T) \tag{8.5}$$

in which R_0 is the value of R at $T_0 = 25\,°\text{C}$, α is the temperature coefficient, and $\Delta T = T - T_0$ is the temperature variation. The evolution of ΔT can be studied using the thermal equivalent circuit reported in Figure 8.4(b), in which $P_{\text{D}} = RI^2$ is the dissipated power, whereas R_{T} and C_{T} are the thermal resistance and capacitance between the coil and the free air, respectively. The equation that describes the circuit is

$$P_{\text{D}} = \frac{\Delta T}{R_{\text{T}}} + C_{\text{T}}\frac{d\,\Delta T}{dt} \tag{8.6}$$

8.3.3 Model implementation

To better analyse the system behaviour, the previous equations have been used to develop a Simulink numerical model of the actuator. The model is divided into two sub–models. The first implements the dynamic equations given by (8.2) and (8.4), whereas the latter takes into account the thermal effects described by (8.5) and (8.6).

(a) Dynamic

(b) Thermal

Figure 8.5: The VCA models

The dynamic sub–model is reported in Figure 8.5(a). The inputs are V_{S}, F_{EXT} and R, (labelled I1, I2 and I3 respectively) whereas the outputs are x (coil position), dx/dt (coil speed), and the current I (labelled O1, O2 and O3). First, F_{EXT} is subtracted from F_{VCA}, and the result is divided by m, thus obtaining the coil acceleration. This signal is then conditioned by the *A–block*, that will be explained later, and reaches the following integrator, which provides the coil speed dx/dt. The speed is further integrated to obtain the coil position x, and – at the same time – it is routed to the *B-block*, which implements (8.2) and evaluates the current I. Note that R is an input, so that the model can take into account any coil resistance variation due to thermal effects. Finally, x determines the force sensitivity K_{F} by means of a look–up table. Then, K_{F} is multiplied by I to obtain F_{VCA} and to close the loop.

The function of the A–block is to keep the displacement x within the $[x_{\min}, x_{\max}]$ range representing the actual stroke of the actuator. If x reaches its upper (lower) bound, the block forces the cylinder speed to zero by resetting the acceleration integrator, and also limits the acceleration to non–positive values (non–negative) so that the bound is not exceeded. If instead x lies within the allowed range, the A–block behaves as transparent.

The thermal model is shown in Figure 8.5(b). The input is the current I (labelled I1) and the outputs are T and R (labelled O1 and O2). The value of I is provided by the dynamic model, and it is used in the *C–Block* to determine T. This result is then processed by the *D–Block*, which evaluates R and sends the obtained value back to the dynamic model.

8.3.4 VCA selection

The choice of actuators has been carried out according to the requirements represented by the gear and clutch force–displacement characteristics. In particular, to simplify the mechanical implementation of the system, we adopted the same actuator for both the gear and clutch levers. The selected VCAs (Accel Technologies VLR0436-0250-00A, see Figure 8.6) provide a continuous force up to 435 N with a stroke of 25 mm. As the force sensitivity is about 37.2 N/A, the maximum continuous force is obtained with a coil current of 11.7 A. The series resistance and inductance of the coil are $2.7\,\Omega$ and 4.8 mH, respectively. This data, along with other parameters, are resumed in Table 8.1.

Figure 8.6: The selected VCA, Accel Technologies (formerly Usas Motion) VLR0436-0250-00A

Table 8.1: Main parameters of the selected VCA

Parameter	Value	Unit
Copper coil Resistance R	2.7	Ω
Copper coil Inductance L	4.8	mH
Maximum voltage	31.7	V
Maximum current	11.7	A
Force sensitivity K_F @ $x = 12.5\,\text{mm}$	37.2	N/A
Back–electro–motive force constant K_B	37.2	Vs/ m
Thermal resistance R_T	2.3	K/W
Thermal capacitance C_T	24	J/K
Movable part displacement $x_{\max}$	25	mm
Movable part mass (cylinder)	0.74	kg
Total actuator mass	3.06	kg

8.3.5 Model application

The parameters of the selected VCA were entered into the dynamic and thermal models, which were connected as reported in Figure 8.7. Then, by choosing the lever characteristics of Figure 8.2 and 8.3 as the mechanical load, the achievable gear and clutch actuation performance was estimated.

As an example, Figure 8.8 shows the time t_S needed by the gear actuator to reach the shift–in position of the Neutral and 2nd gear, respectively, versus the supply voltage V_S, as simulated by the model. When 2nd gear is shifted in, t_S can be kept well below the typical 100 ms shift time of pneumatic systems [99] by simply applying a V_S greater than 16 V. To disengage the clutch, a supply voltage of 21.8 V is needed to produce the 300 N force required. This voltage must carefully be controlled to let the lever gradually move around the disengagement point when the vehicle is starting, thus avoiding wheel slip and engine shutdown.

Taking all this into account, a 28 V, 350 W PWM (Pulse Width Modulation) supply stage was adopted for each actuator. Consequently, both the shifting time and the clutch disengagement can be controlled by varying the related PWM duty–cycle, thus giving the system good flexibility. In addition, it follows from Figure 8.8 that both Neutral and 2nd gear can be selected by controlling the actuator activation time, once the supply voltage of the gear
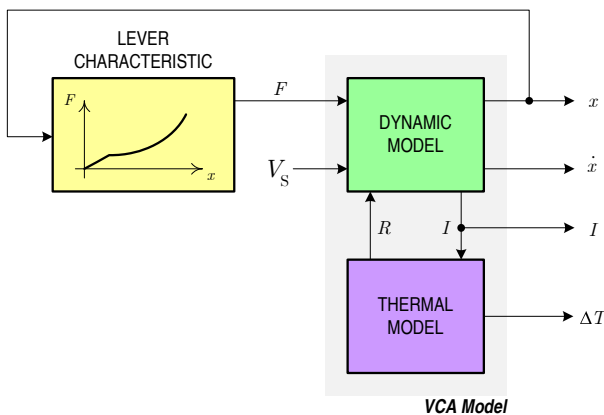
Figure 8.7: Application of the dynamic and thermal VCA models
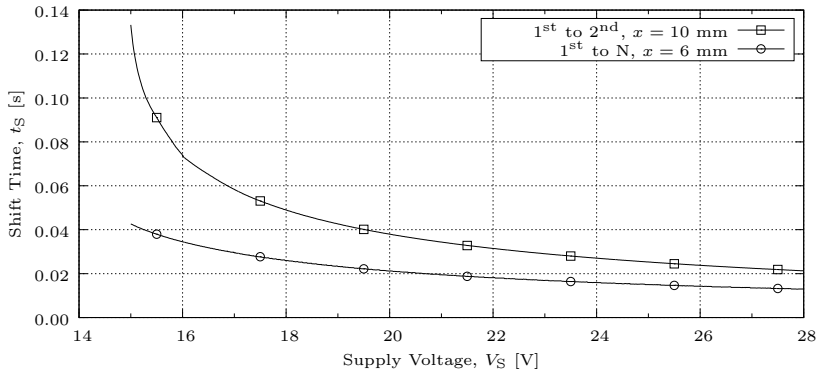
Figure 8.8: Upshift and Neutral selection time versus VCA supply voltage

VCA has been set. This is the solution we adopted to implement the Gear Control Unit shifting algorithms.

8.4 Gear control unit

To enable the driver to control both the gearshift and the start procedure by means of controls located on the steering wheel, a specific electronic *Gear Control Unit* (GCU) has been designed. The developed block diagram is reported in Figure 8.9.

The GCU is divided into a Control Section and a Power Section. The Control Section is connected to the Engine Control Unit (ECU), from which it receives information on the engine RPM. In addition, by means of a dedicated cut–off trigger signal, the GCU is able to trigger the ECU for the temporary suspension of the fuel injection and/or ignition, i.e. to cut-off the engine. This is very useful when upshifting, as it makes clutch disengagement unnecessary. This means that the clutch VCA can be kept off during the whole upshift procedure, thus saving energy and significantly reducing the shift–in time. The Control Section is also connected to the wheel speed sensors and to the driver's interface. This interface has two flipper–paddles which trigger the shifting procedures, as well as a button to select Neutral. A 7–segment display and a led bar mounted on the dashboard show the currently selected gear and
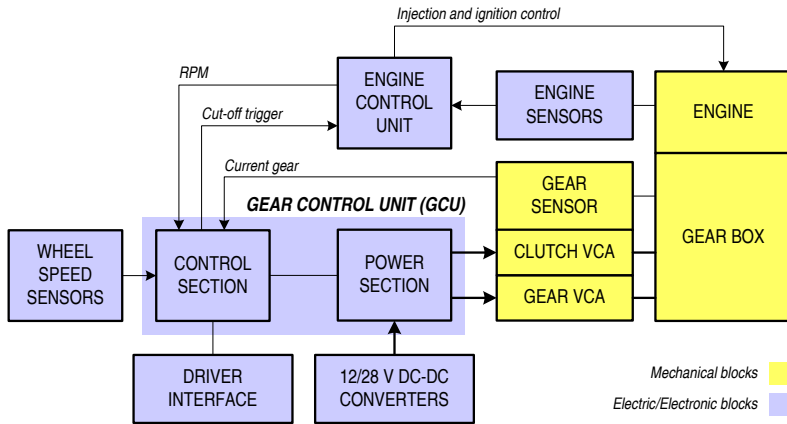
Figure 8.9: The ET1 robotized gearbox architecture

the engine RPM, respectively.

Depending on the driver's shift request and on the vehicle's status, the Control Section produces the appropriate PWM control signals for the two VCAs. These signals are amplified by the Power Section, and finally routed to the actuator coils. The Power Section directly drives the VCAs by means of two NMOS full H–bridge drivers (ST VNH3SP30), which are supplied by two external 12/28 V, 350 W DC/DC converters connected to the car battery.

8.4.1 Hardware architecture

The Control Section is implemented with off–the–shelf components. We opted for a DSP–based platform, in order to simplify the implementation of the control algorithms with the available on–chip resources. The selected board (GAO Tek 2812EVM-I) features a 150 MHz 32 bit fixed–point Texas Instruments TMS320F2812 DSP, which embeds an advanced multichannel PWM peripheral, as well as some 16 bit timer/counters and a 12 bit ADC.

A set of header connectors also enables a custom–designed daughter board to be plugged–in and interfaced to the DSP peripherals. This is the solution we adopted to implement the Power Section, which comprises the two H– bridge drivers, a 5 V voltage regulator to supply the Control Section itself,
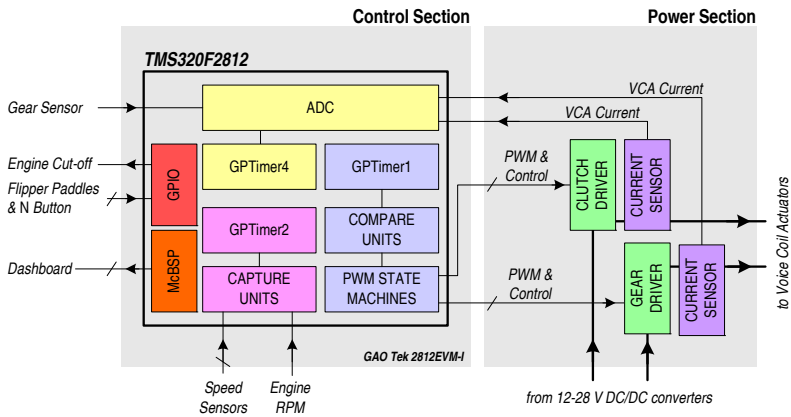
Figure 8.10: Gear Control Unit architecture: Control and Power Sections

and some signal conditioning circuits to interface the different kinds of sensor to the DSP.

Figure 8.10 shows the final GCU architecture along with the main internal DSP blocks employed to implement the system. The *GPTimer2 Capture Units* are used to measure the frequency of the square wave signals coming from the wheel sensors and the ECU RPM output and hence to detect the angular speed of the wheels and the camshaft. Each Capture Unit implements an efficient frequency meter continuously running in the background, so that no computational resources of the DSP are needed for this purpose.

PWM signals are obtained by the dedicated *PWM State Machines*. The PWM period is determined by the *GPTimer1*, whereas the duty–cycle of each output is directly controlled by the DSP through the *Compare Units*. The DSP also handles two control signals for each VCA driver, which enable the H–bridge operation and set the direction of the current in the coil. Two Hall–effect current sensors (Allegro ACS756) measure the amplitude of the current flowing in each VCA coil. Their outputs are routed to the on–chip ADC, thus allowing the Control Section to get feedback from the actuators.

The ADC also receives the gear sensor signal, so as to detect which gear is currently selected. This resistive–type sensor is embedded in the gearbox and is simply made up of a set of different resistances – one per gear – and a rotary switch directly operated by the gearbox. While shifting, i.e. while the

rotary switch is moving, the sensor output remains open till the end of the operation. This means that open circuit will be detected if the gearbox gets stuck between two gears, thus allowing the Gear Control Unit to recover by repeating the shift procedure. The ADC conversion is periodically triggered by the *GPTimer4*. The acquired data are low–pass filtered for denoising and are immediately made available to the main program, which implements the control algorithms.

The flipper–paddles and the Neutral selection button, as well as the ECU connections (RPM and engine cut–off signals), are handled with *General Purpose I/O* (GPIO) pins. The GPIO block provides a built–in input debouncing feature, which avoids multiple–hit detections when the driver operates the GCU controls. We decided to assign the Neutral gear to a dedicated control rather than to the paddles so that the driver cannot accidentally select it during a rapid downshift sequence. Finally, as the dashboard is equipped with a SPI interface, it is connected to the DSP by the *Multi–channel Buffered Serial Port* (McBSP).

8.4.2   Firmware architecture

A simplified flow–diagram of the main program loop is shown in Figure 8.11. First, the inputs of the Gear Control Units are sampled. Then, depending on the driver's request (through flipper–paddles for Upshift or Downshift, or through the button for Neutral) and vehicle's status, i.e. the current gear and speed, one of four main procedures – namely *UpShift*, *DownShift*, *Neutral select* and *Start* – is executed.

UpShift and DownShift procedures are activated by pulling the right and the left paddle, respectively. UpShift is executed only if the gearbox is neither in 5th gear nor in Neutral, which is coded as gear 0. Initially, the ECU is triggered for engine cut–off. Then, a 19 V voltage with an appropriate polarity is applied to the gear actuator for about 45 ms (see Figure 8.8), to enable the gear lever to be fully pushed. Finally, the gear sensor signal is acquired to check if the new gear has been correctly shifted in, and if not, the procedure is retried for a programmable number of times before aborting.

On the other hand, DownShift is executed only if the 1st gear is not in, and
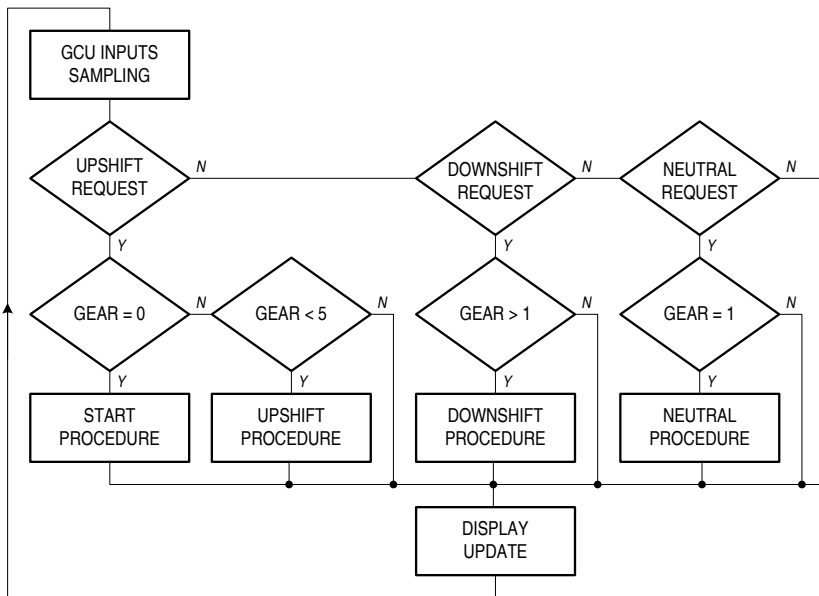
Figure 8.11: Simplified flow–diagram of the Gear Control Unit main program loop. Neutral is coded as gear 0

if the vehicle is not standing in Neutral. This procedure is exactly the same as the previous one, except that it does not trigger the ECU for the engine cut–off and it disengages the clutch before pulling the gear lever. Initially, the clutch VCA is powered with a 28 V voltage. The gear VCA is then actuated and the gear sensor is checked. If the correct gear has been shifted in, the clutch is re–engaged by turning the VCA off, otherwise a new gear actuation is retried for a programmable number of times before aborting.

For the Formula SAE application, the above shifting procedures are retried for a maximum of three times. This is because the system can be calibrated before each race and hence the activation time for the gear lever displacement can be programmed to keep both the likelihood of failure and the shifting time as low as possible. As demonstrated by the on–track endurance test reported in Section 8.5, the control robustness achieved well satisfies the requirements for a race car. Should the system be transferred from race cars to road vehicles, different requirements would apply, as an increased control robustness would be needed to deal with component performance degradation and to ensure a fail–safe operation. In this case, however, a reduction in gear shifting speed is
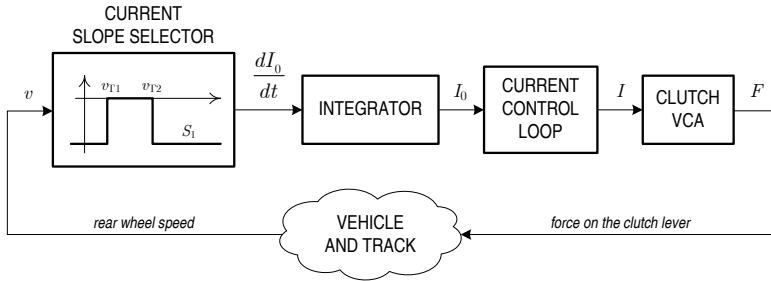
Figure 8.12: Simplified block–diagram of the Start procedure main loop

acceptable, so that the number of trials before aborting can be increased. In addition, the gear sensor response can still be used to check if the new gear has been correctly shifted in, and in case of errors, the activation time should be properly increased or decreased in the next procedure retry.

Unlike Upshift and Downshift, the Neutral select procedure is activated by a dedicated button, and is executed only if the gearbox is in 1st gear. The clutch is initially disengaged, and then the gear VCA is powered at 19 V for 30 ms. As shown in Figure 8.8, this is enough time to select Neutral but too short to get into 2nd gear. Finally, the clutch is re–engaged. Comparing the Neutral select procedure with Upshift and Downshift, it is worth noting that the gear lever displacement is controlled by the VCA activation time. Other strategies, based on the supply voltage regulation using the PWM driver, or combining both PWM and activation time controls, are possible but at the cost of an increase in the control algorithm complexity. However, the implemented method already leads to much better shifting times than the ones obtained with pneumatic systems, as will be shown in Section 8.5.

Finally, the Start procedure in Figure 8.11 is a special case of UpShift from neutral to 1st gear. While the car is stationary, the clutch is disengaged and 1st gear is shifted in. Then the clutch is slowly re–engaged by lowering the actuator current – and thus the pulling force on the lever – in a time–linear way. The current variation is controlled by the loop reported in Figure 8.12. The Gear Control Unit monitors the speed v of the rear wheels. As long as v is lower than the v_{T1} threshold (which we set at 0.1 km/h), the slope S_1 for the actuator current I_0 is selected and integrated. Next, I_0 is applied to the clutch VCA by a PID (Proportional–Integrative–Derivative) control loop,

which acts on the PWM duty–cycle of the actuator driver. Thus, the clutch lever is slowly released. When the wheels start moving, i.e. v reaches v_{T1}, the lever is near to the engagement point. As a consequence, the slope is forced to 0, so that the current flowing in the actuator, as well as the lever position, is locked, and the car can accelerate. To make this operation easier, the driver applies light pressure to the accelerator pedal. Finally, when v is greater than v_{T2} (which we set at $2\,\mathrm{km/h}$), S_1 is selected again, and maintained until the actuator is finally switched off. The two speed thresholds v_{T1} and v_{T2}, as well as the slope S_1, can be fine tuned to the different conditions of the wheels and race–track, thus making the procedure very flexible and also suitable for a fast start.

8.5 Implementation and test

The Control Section board and the custom–made Power Section plug–in board are shown in Figure 8.13 and Figure 8.14, respectively. They are approximately $165\,\mathrm{mm} \times 100\,\mathrm{mm}$ large, and are mounted inside a rugged aluminium case featuring two separate multipolar automotive connectors. The first connector is used to handle the power signals, such as the $28\,\mathrm{V}$ and the $12\,\mathrm{V}$ supplies from the DC/DC converters and the battery, as well as the H–bridge outputs to drive each VCA coil. Instead the second connector is used to handle other control signals from the driver interface or from the vehicle sensors.

The final assembly was installed on board the ET1 for the 2008 Formula SAE Italy event at the Ferrari race–track in Fiorano (Modena, Italy). On that occasion, the ET1 successfully completed the $22\,\mathrm{km}$ run of the endurance event, in which the robustness and reliability of the engine, the gearbox and the GCU were toughly tested. Indeed, the heat required about 500 shift operations, which were all performed correctly by the system.

Before the Formula SAE event, the Gear Control Unit functionality and the robotized gearbox were extensively tested both in a laboratory and during the vehicle's tuning with the chassis dynamometer. As a significant example of the test campaign, the results for 2$^{\text{nd}}$ to 3$^{\text{rd}}$ gear upshift sequence are reported in the first plot of Figure 8.15, which shows the activation signal of the gear VCA and the voltage provided by the gear sensor versus time. This voltage
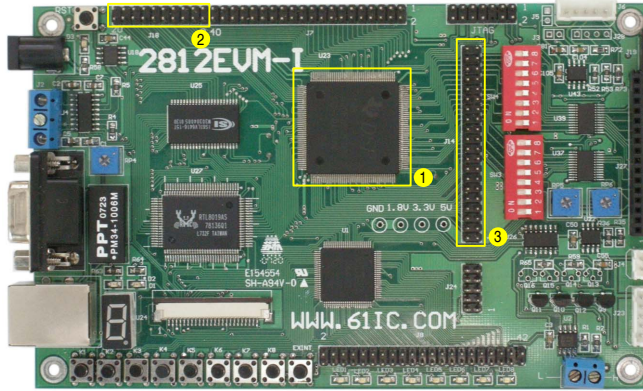
Figure 8.13: The Control Section board (GAO Tek 2812EVM-I): 1. TMS320F2812 DSP, 2. analog input header, 3. digital I/O header
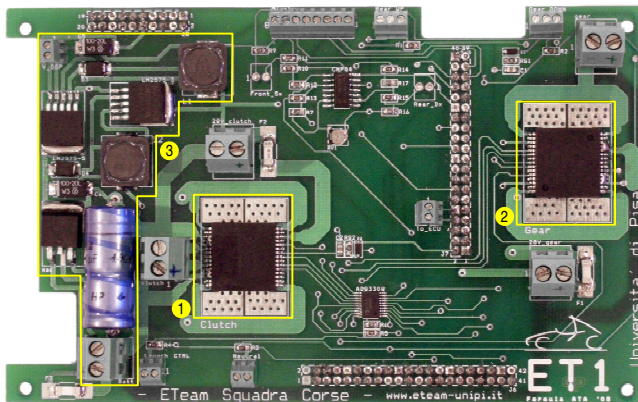
Figure 8.14: The custom–designed Power Section daughter board: 1. VNH3SP30 clutch VCA driver, 2. VNH3SP30 gear VCA driver, 3. power regulators for 5 V and 3.3 V logic
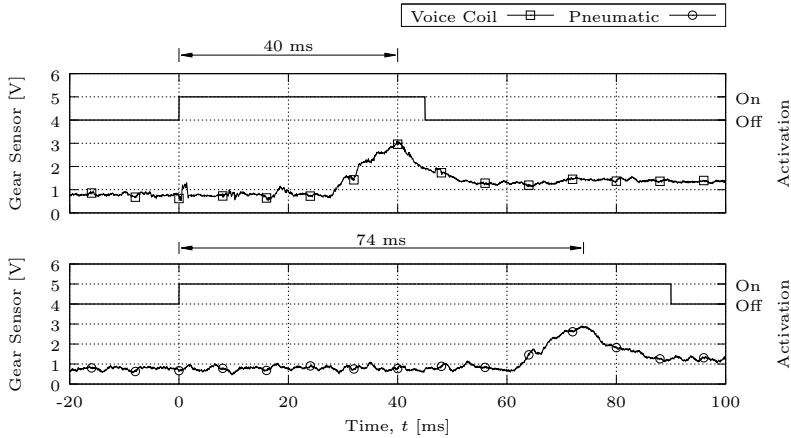
Figure 8.15: 2nd to 3rd gear upshift performance: comparison between the proposed VCA–based system and an equivalent pneumatic counterpart

is obtained through a linear 5 V voltage divider, made up of a $2.2\,k\Omega$ pull–up resistor and the gear sensor itself. A 1 nF capacitor is used to filter the output of the voltage divider. When 2nd gear is shifted out, the gear sensor opens, and the measured voltage starts rising to 5 V with an exponential transient. Then, as soon as 3rd gear is in, the sensor voltage changes slope, as a falling transient starts. The time at which the voltage peak occurs identifies the shifting time, which is about 40 ms. As the VCA was powered at 19 V, this result is in good agreement with the expected performance provided by the model (see Figure 8.8).

Finally, the VCA–based shift–by–wire system was compared with an equivalent pneumatic system. To make a fair comparison, we replaced the gear VCA with a pneumatic actuator and performed all the tests on–board the ET1. We adopted a 25 mm diameter cylinder with the same stroke as the VCA, and a 5.3 bar regulated air supply. Thus, the achievable peak force, i.e. about 260 N, was the same provided by the VCA when powered at 19 V. Also, a fast–switching high–flow solenoid valve was used to control the air flow path towards the cylinder by a dedicated activation signal. A connection between the valve outlet and the cylinder was made with a short 15 cm pipe, which represents a reasonable distance between the two devices when they are mounted on–board. The second plot in Figure 8.15 shows that the upshift time in this case is about 74 ms, which is shorter than the typical 100 ms value

for pneumatic systems [99], but almost double that achieved with the VCA solution. This longer shifting time is mainly due to the inherent propagation delay of the air flowing from the valve to the cylinder, which can only be slightly reduced by using advanced pneumatic solutions, such as valves with a higher flow coefficient. Thus, the 46 % reduction obtained using VCAs could lead to a significant improvement in the overall vehicle performance, particularly in terms of acceleration time [99]. The achieved 40 ms upshift time is also interesting when compared to state–of–the–art mechatronic systems, such as [81] where 76 ms are needed just to send the shift command from the lever–based man–machine interface to the gearbox.

Table 8.2 summarizes the comparison between the VCA–based and the pneumatic systems. Regarding the hardware implementation, our VCA–based solution requires a few additional components besides the two actuators, i.e. DC/DC converters to supply the coils at a voltage higher than the standard vehicle battery voltage. The pneumatic system is more complex, as it requires directional and proportional valves or advanced control techniques [100] (to implement the linear control of the clutch), an air tank, and several pressure regulators. Nevertheless, the VCA–based system is heavier, mainly because of the permanent magnet weight. The total cost is similar for the two implementations, but if we consider as a figure of merit the product of the upshift time and the total cost, the VCA–based system provides a better trade–off between cost and performance, thanks to its faster response. Finally, it is worth noting that the pneumatic system considered in Table 8.2 does not

Table 8.2: Comparison test summary

	VCA–based system	*Pneumatic system*
Upshift time	40 ms	74 ms
Additional hardware	DC/DC converters	Directional and proportional valves, pressure regulators, air tank
Total weight	6 kg	2.5 kg
Total cost	1300 $	1100 $
Upshift time per $	52 s $	81.4 s $

include an air compressor. In fact, in a race–car application, the maximum number of shift activations during a run can be estimated in advance, and the air tank can be sized correctly so that the unnecessary weight of an on–board air refilling system is saved. This does not happen in road vehicles, for which an unlimited shifting endurance must be provided. As a consequence, if we include the compressor in the pneumatic approach, the VCA solution may outperform the other system in terms of all the aspects considered.

9 The ET2ev project

9.1 Aims

If on the one hand the ET1 showed a good performance, on the other hand there was still a great room for improvement. The main issue to face was the overall weight, about 280 kg, which limited the power–to–weight ratio with negative effects on both the vehicle acceleration and handling.

As a consequence, each engineering team was asked to optimize the various subsystems. The aim was to achieve a 10 % weight reduction at least, without

Figure 9.1: The ET2ev during the 2009 Formula Student Germany event at the Hockenheimring

affecting the reliability. As far as the shift–by–wire system is concerned, an important decision was taken together with the mechanical engineers working on the powertrain. The VCA–based system was abandoned in favour of a lighter pneumatic solution. Even if this choice led to a reduced shifting performance (see Table 8.2), about 3.5 kg were recovered.

The final weight of the ET2ev, which is shown in Figure 9.1, was 250 kg, as expected. With respect to the ET1, this new car was also smaller both in length (2885 mm versus 2910 mm) and in height (1279 mm versus 1286 mm). The overall improvement in terms of available power and acceleration time was very good. As an example, Section 9.3 reports the results obtained at the Acceleration Event during the Formula SAE Italy in 2009.

9.2 The pneumatic shift–by–wire system

The new architecture implemented on–board the ET2ev is reported in Figure 9.2. The basic idea was to keep the pneumatic system as simple as possible. To this end, a compressor was avoided in favour of an air tank, which was properly sized to ensure at least 1500 shift operations, as required to complete
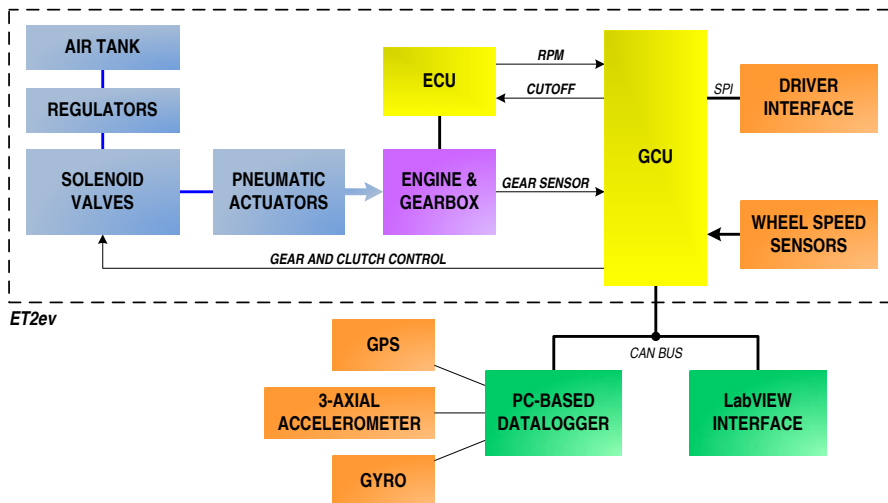
Figure 9.2: The ET2ev architecture

all the dynamic events. In addition, a proportional valve, which is typically used during the start procedure to gradually control the clutch engagement, was avoided. In fact, the original clutch was substituted with an automatic centrifugal clutch, like those commonly mounted on commercial scooters. This new device automatically transfers the engine torque to the wheels whenever the RPM crosses a given threshold, which can be mechanically tuned. The smoothness of the engagement can be regulated as well. As a consequence, the shift–by–wire system is required to control the clutch disengagement only during downshifts by driving the clutch lever in a simple on/off way.

(a) Air tank and regulators   (b) Gear cylinder
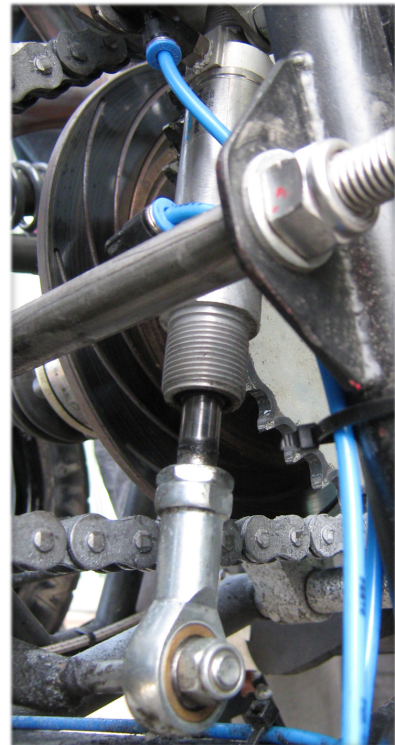
Figure 9.3: The pneumatic system – air tank with low pressure regulator and double–effect cylinder mounted on the gear lever

(a) Valve manifold
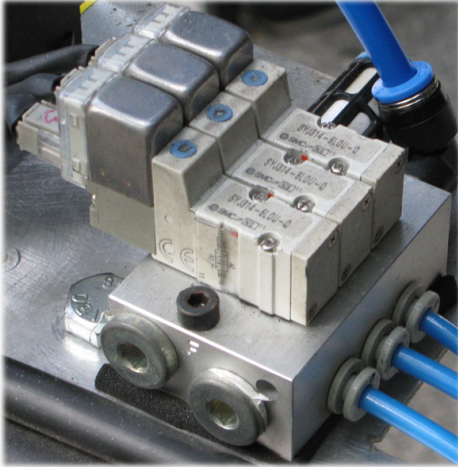
(b) GCU
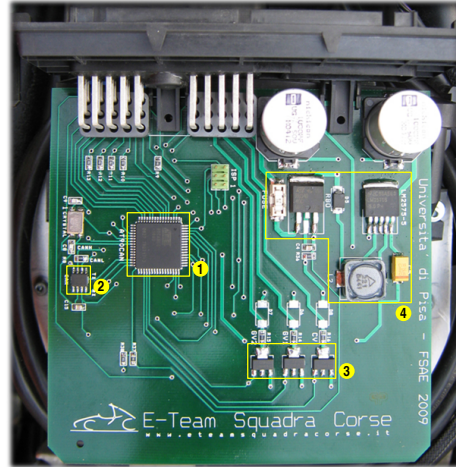
Figure 9.4: The gear and clutch solenoid valve manifold, and the new μC–based GCU (1. AT90CAN64 μC, 2. TJA1050 CAN transceiver, 3. N–MOSFET low–side valve switches, 4. voltage regulators)

To reduce the overall weight of the system, a 1.2 kg carbon–fibre air tank featuring a 1 l, 300 bar capacity was selected (see Figure 9.3(a)). Following the diagram of Figure 9.2, the tank output is connected to a two–stage regulator, which produces the low 6 bar operating pressure for the solenoid valve manifold, which is shown in Figure 9.4(a).

Three fast–switching 2–way valves have been adopted to control the gear and clutch actuators. As discussed in Section 8.2, the gear lever has to be pushed and pulled from its idle position. For this reason, the gear actuator is a double–effect cylinder, and it is controlled by the first two valves. On the other hand, the clutch actuator is only required to push the clutch lever, as the lever spring is able to return it to its idle position when it is switched off. As a consequence, a single–effect cylinder has been adopted for the clutch, and it is controlled by the third valve.

9.3 The new Gear Control Unit

From the previous remarks it follows that the complexity of the algorithms to be implemented by the GCU is now dramatically reduced. Therefore, the DSP–based platform was abandoned and a new custom–made μC–based unit was designed from scratch.

The developed board, which is shown in Figure 9.4(b), features a 16 MHz, 8 bit Atmel AT90CAN64 microcontroller [27]. This device drives the 12 V solenoid valves by means of three N–MOSFET acting as low–side switches, and implements the same UpShift, DownShift, and Neutral select procedures described in Section 8.4.2. In particular, the gear lever control strategy is still based on the actuator activation timings, which have been fine tuned on–board to ensure the correct Neutral selection.

9.3.1 Automatic upshift

Some new features have been added to the firmware besides the basic shift procedures. The first one is an *automatic upshift algorithm*, which is able to automatically trigger an UpShift procedure when RPM crosses a given threshold. Actually, for each gear except for the 5th, an auto–upshift RPM threshold can be specified. Indeed, it can be proved that for each selected gear there is an optimal RPM value at which the upshift can be triggered to maximize the acceleration. Threshold values have been determined by the vehicular engineers team through the numerical model analysis of the ET2ev performance. These values mainly depend on the engine power/torque versus RPM characteristics, as well as on the available gear ratios, the grip factor of wheels on track, and on other dynamic parameters, above all the total vehicle mass.

Basically, when the algorithm is enabled, it continuously polls the RPM value to see if the threshold for the current gear has been crossed. In that case, it triggers the UpShift procedure by producing a software interrupt which emulates the pressure of the upshift paddle by the driver. A specific control is also carried out during downshifts. Indeed, when the driver manually triggers a downshift, the RPM usually rises up to follow the shorter gear ratio. This

Table 9.1: Acceleration Event results – Formula SAE Italy 2009

Run	Time	UpShift mode
I	4.682 s	auto
II	4.643 s	auto
III	4.666 s	auto
IV	4.835 s	manual

RPM increment may be so high to cross the auto–upshift threshold. Therefore, to avoid the auto–triggering of an unwanted UpShift procedure, the algorithm detects downshift operations and suspends the threshold checking for a given timeout, which can be fine tuned on–board.

The benefits of the automatic upshift algorithm have been experimented at the 2009 Formula SAE Italy event in Varano de' Melegari. Numerical simulations provided an estimated time for the 70 m Acceleration Event of about 4.717 s. The actual performance achieved in the four allowed runs are reported in Table 9.1. It can be noted that the values are extremely close to the expected one. The best time was 4.643 s, which represents an improvement of about 40 % with respect to the ET1 performance. In particular, this value was achieved in the second run, with the use of the automatic upshift. If compared to the fourth run, in which the algorithm was disabled, the benefit is about 0.20 s. Such a difference is extremely significant, as the first ten top–teams are exactly concentrated in a 0.20 s time gap.

9.3.2   CAN connectivity

Another important feature added to the new GCU is the *CAN bus connectivity*, obtained by means of the CAN peripheral embedded in the μC and by the TJA1050 transceiver [101] (see Figure 9.4(b)). Even if the GCU was the only CAN–enabled device on–board the ET2ev, this task was carried out in anticipation of future improvements on E–Team vehicles. Every 10 ms, the GCU sends over the CAN bus a status message containing the current gear number, along with a set of flag indicating if the automatic upshift algorithm is active, or if there is a shifting procedure ongoing. In addition, by means
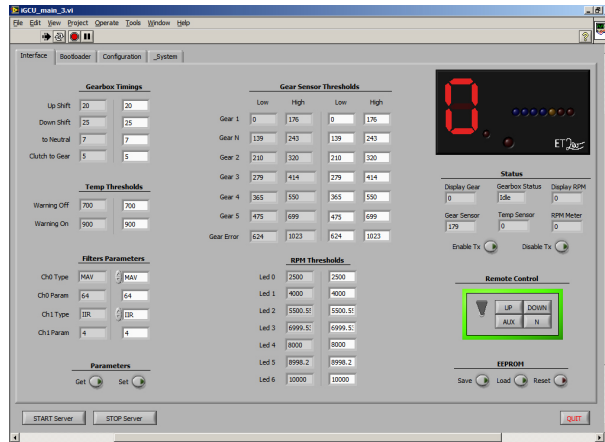
Figure 9.5: The GCU configuration utility

of the CAN interface, each parameter of the GCU can be remotely read or written. This allows for a rapid on–board configuration of the shifting timings and thresholds using, for instance, a PC with a CAN adapter.

Finally, the CAN interface may also be used to reflash the GCU. A bootloader application has been developed and stored in a reserved section of the µC flash memory, thus allowing to program the application section via CAN without the need of removing the unit from the vehicle and without the use of the programmer. Figure 9.5 shows the front panel of a custom LabVIEW application which allows to configure all the GCU parameters, monitor the status message, and update the firmware.

In Figure 9.2 a data–logging system is also shown. Actually, this is a net-book PC–based data–logger, implemented in LabVIEW, which is mounted on–board only during tests. It is able to log the GCU activity through a USB–to–CAN adapter, as well as the data coming from external USB sensors comprising a 3–axial $\pm 8g$ accelerometer, a $\pm 320\,\mathrm{deg}$ /s gyroscope, and a 5 Hz GPS.

10

The ET3 project

10.1 Aims

From a mechanical point of view, the ET3 was the result of a completely new design process (see Figure 10.1). The main target was still the overall weight reduction, but also the optimization of the layout and the reduction of dimensions were taken into account.

A smaller steel frame was developed, with a new geometry for the differential

Figure 10.1: The ET3 during the 2010 Formula Student Germany event at the Hockenheimring

support and for suspensions. In particular, suspension arms, as well as the entire bodyworks, were made of carbon–fibre. This allowed to reach a overall weight of 210 kg, corresponding to a 16 % reduction with respect to the ET2ev, and to a 25 % reduction with respect to the ET1. In addition, the total length and height were pushed down to 2650 mm and 1020 mm, respectively.

This optimization process led to a higher power–to–weight ratio and to a better handling, with a consequent improvement of the overall performance. This was confirmed by the results obtained at the 2010 Formula SAE Italy event in Varano de' Melegari, i.e. first Italian team, ninth overall.

10.2   On–board electronics

From an electronic point of view, the new architecture implemented on–board the ET3 is reported in Figure 10.2. Even if the pneumatic shift–by–wire system and the GCU have been confirmed, several changes with respect to Figure 9.2 can be noted. The main difference is that the CAN bus is now largely used. Point–to–point connections have been reduced, so that units exchange data only via the 2–wire CAN bus. This way, a significant reduction of the wiring harness complexity, cost and weight is achieved.

The ECU has been replaced with an advanced CAN–enabled model (Walbro HPUH–1), which continuously sends over the bus the engine status information, such as internal temperatures, pressures, and injection/ignition timings, as well as the RPM. The dashboard interface has been also redesigned to be connected on the CAN bus. Each control actuated by the driver is now translated into a proper message which is sent over the bus to the different units.

Moreover, two new important subsystems have been designed and implemented on–board the ET3. The first one is the *Data Logging System*, which is a very flexible device able to acquire almost any information available on the vehicle. Indeed, it can log up to 8 analog inputs and 4 digital inputs (for frequency measurements), as well all the data coming from the CAN bus and those provided by several internal sensors, comprising a 3–axial accelerometer, a gyroscope, and a GPS.
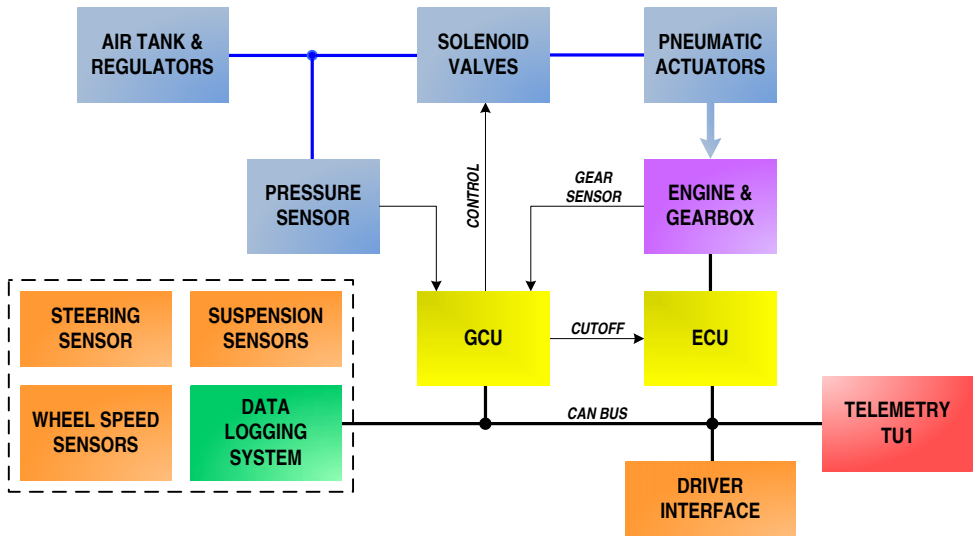
Figure 10.2: The ET3 architecture

The second one is the *Telemetry System*. Basically, it is made up by two twin units working at $2.4\,\mathrm{GHz}$ in a $2\,\mathrm{km}$ range over an encrypted link. The first unit (TU1) is mounted on–board the ET3, and listens to the CAN activity. The detected messages are encoded and transmitted to the other unit (TU2), which is connected to a PC via USB. In particular, the link is full–duplex, i.e. the PC is also able to send CAN message over the ET3 bus.

10.2.1 Gear Control Unit extension

The firmware of the GCU has been revised according to the new architecture. Indeed, the information which previously came through point–to–point connections from the dashboard (flipper–paddles, Neutral button) and the ECU (engine RPM), now are available only on the CAN bus. Hence, all the functions which handle the shifting procedures, as well as the auto–upshift algorithm, were updated.

A new pressure sensor was also added to the pneumatic circuit after the regulators, and routed to the GCU. The pressure information has been used to develop a correction algorithm for the gear cylinder activation timings.

This way, the time required to select the Neutral can be adapted to the actual pressure available in the circuit, and wrong selection can be avoided.

Finally, a new *semi–automatic* upshift mode has been implemented. From a functional point of view, it is identical to the previous full–automatic one. The only difference is in the activation of the auto–upshift algorithm. Indeed, when the GCU is in semi–automatic mode, the auto–upshift algorithm is activated only if the driver keeps the right paddle down, and it is deactivated as soon as the paddle is released. This allows the driver to have the full control over the vehicle, and at the same time to rapidly require the help of the auto–upshift algorithm whenever he wants, for instance after a chicane or when entering a straight.

10.2.2 Data logging system

The data logging system is shown in Figure 10.3. Like the GCU, it is based on an 16 MHz, 8 bit Atmel AT90CAN64 microcontroller. As reported in Section 10.2, this system is able to log a large amount of information coming both from external and internal sensors.

For external sensors, eight 0–5 V analog inputs are buffered by dedicated op–amps and cyclically acquired by the μC through its internal 10 bit ADC. In addition, four digital inputs for frequency measurement are routed to specific timer interrupt pins. Thanks to an efficient use of the available 16 bit timer/counter peripherals, the measurement is carried out in background with a very reduced load for the CPU.

Once mounted on–board the ET3, the analog inputs have been used to measure the stroke of each suspension and the steering angle. To this end, four 30 mm linear potentiometric sensors have been mounted on dampers, and a smaller 10 mm sensor have been applied on a steering arm. The four digital inputs have been used to detect the wheel speed. In particular, the clock frequency of the μC timers was selected according to the teeth number of the adopted phonic wheels, so as to achieve a good measurement resolution. As a final result, using 8–teeth phonic wheels, a resolution lower than 1 km/h has been achieved for speeds up to 209 km/h, which is about twice the maximum speed reached by Formula SAE vehicles.

Internal sensors are instead connected to the µC by means of standard digital interfaces. To simplify the final assembly of the PCB, sensor modules providing header connectors have been adopted. In particular, the gyroscope (Analog Devices ADIS16250, ±320 deg /s) and the 3–axial accelerometer (ST LIS3LV02DL, ±6g) are routed to the µC SPI interface, whereas the GPS (RF Solutions GPS-610FA, 10 Hz, external antenna) to the UART interface.

The system is also able to log a set of incoming CAN messages defined by the user. Once installed on–board, this function has been applied to the ECU and GCU status messages, which, together with the information provided by sensors, allow to reconstruct the dynamic behaviour of the vehicle with a good precision. In addition, the CAN interface may be used to send sensor data back to other units. This allows to virtually share sensors without increasing the wiring harness complexity. On the ET3, this function has been used to send speed data to the dashboard, and suspension data to the telemetry, so that vehicle engineers were able to monitor in real–time the car behaviour during the setup phase.
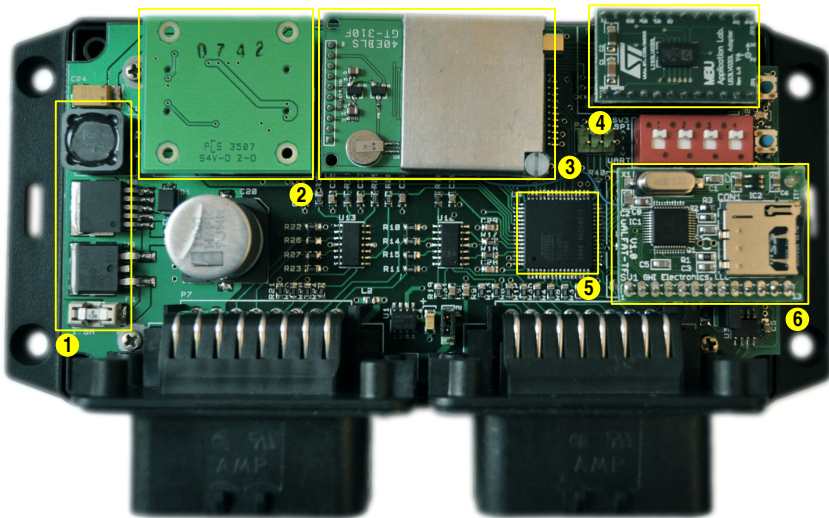
Figure 10.3: The data logging system. (1. voltage regulators, 2. ADIS16250 gyro module, 3. GPS-610FA GPS module, 4. LIS3LV02DL accelerometer module, 5. AT90CAN64 µC, 6. uALFAT SD module)

All the data are sampled at 40 Hz, which is a very good sampling rate if compared to state–of–the–art commercial data loggers. The samples collected are stored in a human–readable CSV format on a SD memory card, which is handled by means of a dedicated module (GHI Electronics uALFAT) connected to the µC via SPI. The SD card is also used to configure the data logger. Indeed, during the power–up initialization phase, the µC checks if a configuration text file is present on the card, and in that case it loads its content. A basic set of configuration commands allow to enable the analog or digital input channels, as well as to define the set of CAN messages to be acquired and to be sent back to the other units. The card can be extracted without opening the data logger case and directly inserted into a PC for offline analysis. With a 2 GB SD card, the acquisition can run for 97 h if all the channels are enabled.

10.2.3   Telemetry system

Both the twin units (see Figure 10.4) of the telemetry system are based on the same Atmel AT90CAN64 microcontroller. The wireless link has been obtained by means of the LT2510 radio module by Laird Technologies, which works in the 2.4 GHz ISM band with a spread–spectrum frequency–hopping modulation, according to the IEEE 802.11 standard.

The maximum theoretical data rate achievable by the link is 500 kbit/s, whereas the measured outdoor operating range is about 2 km when the maximum 200 mW transmission power is selected. Such values well meet the requirements for Formula SAE applications, in which both the amount of data to be transferred and the range to be covered are rather small.

The selected radio modules allow to create an wireless server–client encrypted network, in which the server is able to send broadcast messages to all clients, as well as to serve single client requests. The ET3 unit is the server, whereas the PC unit is the client. This allows to add in future other clients if, for instance, telemetry data have to be received by more than a single PC for advanced monitoring purposes.

Each module is connected to the related µC through the UART interface. The firmware of the on–board unit allows to select the set of CAN messages

to be forwarded to the PC. This set comprises the GCU and ECU status messages, as well as the suspension sensors information provided by the data logger. On the other hand, the μC of the PC unit is connected to the USB bus by means of a USB–to–UART bridge implemented with the FTDI FT232 chip.

One of the most important features of the developed telemetry system is the full–duplex communication, which often is not provided even by state–of–the–art commercial solutions. Thanks to bidirectional capabilities, the telemetry system allows to virtually replace a wired CAN–to–USB adaptor with a wireless one. All the LabVIEW applications previously developed to configure the different units or to analyse status parameters, now are able to run wireless with the vehicle in motion. his provides a high level of flexibility. Indeed, both vehicle diagnostics and unit configuration can be carried out remotely and in real–time, with a significant reduction of the vehicle setup time. This way, a possible fault can be rapidly detected and repaired, and the effect of a new configuration parameter can be immediately evaluated.

This important benefit was appreciated during the setup phase of the GCU timings. This task, which required about a day of stop–and–go tests for the ET2ev, was completed in a few hours without the need of stopping the vehicle to update the GCU configuration.

(a) on–board unit

(b) PC–side unit

Figure 10.4: The ET3 telemetry units

Conclusions

In this work, the results obtained from the research activities on electronic systems, carried out during the three year PhD program, were reported. In particular, the main issues related to the design of advanced electronic systems for two specific state–of–the–art sensing and control applications have been analysed and addressed.

The first application focused on the design of a long–range optical fibre Distributed Temperature Sensor (DTS) based on Raman–scattering. After describing the basic working principles of DTS systems and analysing the trade–offs between their performance parameters, a new PC–based architecture which allows to cover distance ranges up to 87.4 km with a metre–scale spatial resolution has been proposed and examined in detail.

The system has been implemented using state–of–the–art commercial–off–the–shelf components. Several solutions have been adopted to increase the overall performance. To shorten the measurement time, Stokes and anti–Stokes traces are simultaneously acquired by means of two identical APD–based optoelectronic receivers working in parallel. A particular effort has been made to fine control the bias voltage (21–101 V with 5 mV steps) and the operating temperature of each APD, so as to precisely track the optimum multiplication factor for which the highest available SNR is achieved. In addition, a differential transimpedance amplification chain with configurable gain and offset has been implemented to let the amplitude of incoming signals properly match the dynamic range of the following ADC.

Both the stability and the frequency response of the amplification chain have

been analysed. Proper feedback networks have been determined to obtain the bandwidth required for the targeted spatial resolution.

To acquire backscattered signals, a state–of–the–art ADC/FPGA platform has been adopted. To give the system a high flexibility level, the architecture developed on–board the FPGA implements several user–selectable decimation algorithms which allow to extend up to 4 times the maximum acquisition length of 21.85 km due the limited FPGA memory resources. It is well known that the main drawback of decimation algorithms is the inherent degradation of the sampling resolution. Such an issue is commonly left unresolved in commercially available DTS systems. In this work, instead, this issue has been addressed and successfully solved by implementing an interleaved sampling algorithm which is able to recover the original sampling resolution also when the longest 87.4 km range is selected.

Acquired data are transferred via USB to a PC, where a custom LabVIEW application has been developed to extract the temperature profile. In addition, to let the PC handle the large amount of parameters, each block of the system has been equipped with a µC, which receives configuration commands from LabVIEW through a multiprocessor UART (M–UART) bus. A particular effort has been made to develop a flexible, efficient and easy to configure firmware/software framework, which allowed to rapidly implement the system control functionalities, and which can easily be reused whenever a PC–based instrument have to be designed.

Preliminary test results showed the expected performance, and confirmed the validity of the adopted design solutions.

Finally, a novel laser pulse coding technique which allows to significantly improve the SNR of the acquired traces, and hence to reduce the measurement time, has been developed. Unlike other methods reported in the literature, this solution is well suited for long–range DTS systems, as it can easily be implemented also with high–power pulsed lasers featuring limited repetition rates. A theoretical analysis applied to the developed DTS prototype showed that the time required to appreciate a temperature resolution of 1 K at 30 km can be dramatically reduced of about 95 %. Preliminary experiments over a 10 km range have confirmed the expected results. Thanks to its strong impact on DTS performance, for this technique a national design patent has been filed.

The second application focused instead on the development of embedded automotive systems for Formula SAE race cars. The first activity was the design and implementation of a innovative shift–by–wire system based on linear Voice Coil Actuators (VCAs). A numerical model which describes both the dynamic and the thermal behaviour of linear VCAs has been developed. The model allowed to select proper actuators for the targeted application, as well as to determine their basic control strategies. A custom–made electronic Gear Control Unit (GCU) based on a high–performance DSP platform has been developed. A particular attention has been paid to properly map the embedded DSP resources to the targeted hardware architecture, and to optimize the efficiency of the shifting algorithms implemented in firmware. The final system was mounted on–board the ET1, the first Formula SAE race car developed at the University of Pisa. The obtained on–track results showed a very good shifting performance, particularly if compared with other solutions reported in the literature.

To optimize the power–to–weight ratio of the vehicle, a lighter but slower shift–by–wire system based on pneumatic actuators was developed for the second year car, the ET2ev. Thanks to the use of an automatic clutch, the control algorithms were dramatically simplified, ad a new μC–based GCU was implemented. A novel automatic upshift algorithm, which helps the driver during accelerations, was also developed and its benefits were confirmed by on–track results. Finally, a data logging system and a telemetry system were designed for the third year car, the ET3. The first allows to record on a removable SD card a large amount of data coming from several on–board sensors, with a sampling frequency of 40 Hz. On the other hand, the telemetry system allows to establish a bidirectional wireless link between the vehicle CAN bus and a PC, with a covering range of 2 km. This way, the status of the car can be monitored in real–time and possible faults can be rapidly detected while the car is running. In addition, the PC can configure the on–board units remotely, thus allowing to significantly speed–up the vehicle setup operations. All these features make these two last systems very interesting if compared with high–end commercially available products.

Bibliography

[1] A. H. Hartog, "Progress in distributed fiber optic temperature sensing," in *Proc. of Fiber Optic Sensor Technology and Applications Conference 2001*, M. A. Marcus and B. Culshaw, Eds., vol. 4578. SPIE, 2002, pp. 43–52.

[2] Y. Wang, Z. Li, and Z. Jiang, "An improved distributed optical fiber sensor (DOFS) for monitoring long-distance buried oil pipeline leakage and intrusion," in *Proc. 4th IEEE Conference on Industrial Electronics and Applications ICIEA 2009*, 2009, pp. 318–320.

[3] S. J. Collocott, "A device for the remote sensing of the temperature of high-voltage transmission conductors," *IEEE Transactions on Industrial Electronics and Control Instrumentation*, vol. IECI-28, no. 2, pp. 116–119, 1981.

[4] X. Zhang, X. Jiang, and H. Xie, "The application of fiber optic distributed temperature sensor to fault detection of XLPE insulated underground cable," in *Proc. 6th International Conference on Properties and Applications of Dielectric Materials*, vol. 1, 2000, pp. 154–156.

[5] T. Hara, K. Terashima, H. Takashima, H. Suzuki, Y. Nakura, Y. Makino, S. Yamamoto, and T. Nakamura, "Development of long range optical fiber sensors for composite submarine power cable maintenance," *IEEE Transactions on Power Delivery*, vol. 14, no. 1, pp. 23–30, 1999.

[6] A. F. Fernandez, P. Rodeghiero, B. Brichard, F. Berghmans, A. H. Hartog, P. Hughes, K. Williams, and A. P. Leach, "Radiation-tolerant Raman distributed temperature monitoring system for large nuclear infrastructures," *IEEE Transactions on Nuclear Science*, vol. 52, no. 6, pp. 2689–2694, 2005.

[7] A. Hartog, "Distributed fibre-optic temperature sensors: technology and applications in the power industry," *Power Engineering Journal*, vol. 9, no. 3, pp. 114–120, Jun 1995.

[8]   A. A. Khan, V. Vrabie, G. D'Urso, and J. I. Mars, "SVD based automated dike monitoring system using DTS data," in *Proc. 34th Annual Conference of IEEE Industrial Electronics IECON 2008*, 2008, pp. 1931–1936.

[9]   R. M. Odic, R. I. Jones, and R. P. Tatam, "Distributed temperature sensor for aeronautic applications," in *Proc. 15th Optical Fiber Sensors Conference Technical Digest OFS 2002*, vol. 1, 2002, pp. 459–462.

[10]   K. Aoyama, K. Nakagawa, and T. Itoh, "Optical time domain reflectometry in a single-mode fiber," *IEEE Journal of Quantum Electronics*, vol. 17, no. 6, pp. 862–868, 1981.

[11]   S. Adachi, "Distributed optical fiber sensors and their applications," in *Proc. SICE Annual Conference*, 2008, pp. 329–333.

[12]   P. C. Wait, K. D. Souza, and T. P. Newson, "A theoretical comparison of spontaneous Raman and Brillouin based fibre optic distributed temperature sensors," *Optics Communications*, vol. 144, no. 1–3, pp. 17–23, 1997.

[13]   M. N. Alahbabi, Y. T. Cho, and T. P. Newson, "Simultaneous distributed measurements of temperature and strain using spontaneous Raman and Brillouin scattering," in *Proc. of Second European Workshop on Optical Fibre Sensors*, vol. 5502, 2004, pp. 488–491.

[14]   R. Singh, "C. V. Raman and the Discovery of the Raman Effect," *Physics in Perspective (PIP)*, vol. 4, pp. 399–420, 2002, 10.1007/s000160200002.

[15]   S. Maughan, H. Kee, and T. Newson, "Simultaneous distributed fibre temperature and strain sensor using microwave coherent detection of spontaneous Brillouin backscatter," *Measurement Science and Technology*, vol. 12, p. 834, 2001.

[16]   D. Long, *The Raman effect: a unified treatment of the theory of Raman scattering by molecules.*   John Wiley & Sons Inc, 2002.

[17]   M. A. Farahani and T. Gogolla, "Spontaneous Raman scattering in optical fibers with modulated probe light for distributed temperature Raman remote sensing," *IEEE/OSA Journal of Lightwave Technology*, vol. 17, no. 8, pp. 1379–1391, 1999.

[18]   D. Derickson, *Fiber Optics Test and Measurements.*   Prentice–Hall, 1998, ch. OTDRs and backscatter measurements, J. Beller, pp. 434–474.

[19]   J. Zou, S. Wang, and Z. Yang, "Distributed temperature measurement based on single-channel anti-Stokes light," *International Journal of Infrared and Millimeter Waves*, vol. 21, pp. 793–803, 2000, 10.1023/A:1026425013158.

[20]   J. Dakin, D. Pratt, G. Bibby, and J. Ross, "Distributed optical fibre Raman temperature sensor using a semiconductor light source and detector," *Electronics Letters*, vol. 21, no. 13, pp. 569–570, 20 1985.

[21]   A. H. Hartog, A. P. Leach, and M. P. Gold, "Distributed temperature sensing in solid-core fibres," *Electronics Letters*, vol. 21, no. 23, pp. 1061–1062, 1985.

[22]   J. P. Dakin, D. J. Pratt, G. W. Bibby, and J. N. Rose, "Distributed antistokes ratio thermometry," in *Optical Fiber Sensors*.   Optical Society of America, 1985, p. PDS3.

[23]   M. Jaaskelainen, "Temperature monitoring of geothermal energy wells," in *Proc. of Fourth European Workshop on Optical Fibre Sensors*, vol. 7653, 2010.

[24]   *Gemini^{TM}*, SensorTran, Austin, TX (USA), www.sensortran.com.

[25]   *DiTemp$^{®}$*, Smartec SA, Manno (CH), www.smartec.ch.

[26]   M. D. Jones, "Using Simplex codes to improve OTDR sensitivity," *IEEE Photonics Technology Letters*, vol. 5, no. 7, pp. 822–824, 1993.

[27]   A. Corporation, "AVR Embedded RISC MCU Core Peripheral - UART," Technical Note 1130, available at www.atmel.com.

[28]   P. J. Samson, "Analysis of the wavelength dependence of Raman backscatter in optical fibre thermometry," *Electronics Letters*, vol. 26, no. 3, pp. 163–165, 1990.

[29]   S. Shibata, M. Horiguchi, K. Jinguji, S. Mitachi, T. Kanamori, and T. Manabe, "Prediction of loss minima in infra-red optical fibres," *Electronics Letters*, vol. 17, no. 21, pp. 775–777, 15 1981.

[30]   S. Adachi and Y. Koyamada, "Analysis and design of Q-switched erbium-doped fiber lasers and their application to OTDR," *Journal of Lightwave Technology*, vol. 20, no. 8, pp. 1506–1511, Aug 2002.

[31]   *ELPM-30 Erbium pulsed fibre laser datasheet*, IPG Photonics Corp., Oxford, MA (USA), downloadable at www.ipgphotonics.com.

[32]   P. Myslinski, J. Chrostowski, J. Koningstein, and J. Simpson, "High power Q-switched erbium doped fiber laser," *IEEE Journal of Quantum Electronics*, vol. 28, no. 1, pp. 371 –377, Jan 1992.

[33]   G. Agrawal, *Fiber-Optic Communication Systems*.   John Wiley & Sons, Inc., 1997, vol. 1.

[34]   B. E. A. Saleh and M. C. Teich, *Fundamentals of Photonics*.   John Wiley & Sons, Inc., 2001.

[35] R. K. Willardson and A. C. Beer, *Infrared Detectors II*, ser. Semiconductors and Semimetals. Academic Press, 1977, vol. 12.

[36] R. J. McIntyre, "The distribution of gains in uniformly multiplying avalanche photodiodes: Theory," *IEEE Transactions on Electron Devices*, vol. 19, no. 6, pp. 703–713, Jun 1972.

[37] *NR8800 Avalanche Photo Diode Series datasheet*, NEC Electronics Corp., Japan, downloadable at www.nec.com.

[38] *C-1520 Raman Filter datasheet*, Cube Optics AG, Mainz (DE), downloadable at www.cubeoptics.com.

[39] W. Bennett, "Spectra of quantized signals," *Bell System Technical Journal*, vol. 27, no. 3, pp. 446–471, July 1948.

[40] *MAX1932 datasheet and AN1805 application note*, Maxim Integrated Products, Inc., Sunnyvale, CA (USA), downloadable at www.maxim-ic.com.

[41] *ADG714 datasheet*, Analog Devices, Inc., Norwood, MA (USA), downloadable at www.analog.com.

[42] *TMP275 datasheet*, Texas Instruments, Inc., Dallas, TX (USA), downloadable at www.ti.com.

[43] *SA5211 datasheet*, NXP Semiconductors, Eindhoven (NL), downloadable at www.nxp.com.

[44] *AD8139 datasheet*, Analog Devices, Inc., Norwood, MA (USA), downloadable at www.analog.com.

[45] *AD5667R datasheet*, Analog Devices, Inc., Norwood, MA (USA), downloadable at www.analog.com.

[46] *ADG713 datasheet*, Analog Devices, Inc., Norwood, MA (USA), downloadable at www.analog.com.

[47] *AD8330 datasheet*, Analog Devices, Inc., Norwood, MA (USA), downloadable at www.analog.com.

[48] *AD5161 datasheet*, Analog Devices, Inc., Norwood, MA (USA), downloadable at www.analog.com.

[49] B. Pellegrini, "Improved Feedback Theory," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, no. 9, pp. 1949–1959, sept. 2009.

[50] *AD9640 datasheet*, Analog Devices, Inc., Norwood, MA (USA), downloadable at www.analog.com.

[51] *Virtex$^{®}$–4 FPGA user guides*, Xilinx, Inc., San Jose, CA (USA), downloadable at www.xilinx.com.

[52] V. Haasz and A. Platil, "Virtual Instrument - no Virtual Reality but Real PC Based Measuring System," in *Proc. IEEE Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 2005. IDAACS 2005.*, 5-7 2005, pp. 261–266.

[53] A. Rodriguez, L. Parrilla, A. Simon-Muela, M. Prats, C. Querejeta, and F. de Blanes, "Real time sensor acquisition platform for experimental UAV research," in *Proc. IEEE/AIAA 28th Digital Avionics Systems Conf. 2009. DASC '09.*, oct. 2009, pp. 5.C.5–1 – 5.C.5–10.

[54] G. Swiszcz, A. Cruden, C. Booth, and W. Leithead, "A data acquisition platform for the development of a wind turbine condition monitoring system," in *Proc. Int. Conf. on Condition Monitoring and Diagnosis, 2008. CMD 2008.*, apr. 2008, pp. 1358–1361.

[55] A. Abba, A. Manenti, A. Suardi, C. Fiorini, and A. Geraci, "Digital configurable processor for acquisition and elaboration of data from detector arrays," in *Proc. IEEE Nuclear Science Symp. Conf. Record (NSS/MIC)*, 2009, pp. 559–562.

[56] R. A. Andrews and B. Webster, "A component oriented software engineering approach to a deeply embedded firmware based control platform," in *Proc. Canadian Conf. Electrical and Computer Engineering*, 2005, pp. 1982–1984.

[57] ——, "Managing component variability with a firmware calibration model; reduced manufacturing costs and freedom in vendor selection," in *Proc. Canadian Conf. Electrical and Computer Engineering*, 2005, pp. 1985–1987.

[58] C. C. W. Robson, A. Bousselham, and C. Bohm, "An FPGA-Based General-Purpose Data Acquisition Controller," *IEEE Transactions on Nuclear Science*, vol. 53, no. 4, pp. 2092–2096, 2006.

[59] C. C. W. Robson, S. Silverstein, and C. Bohm, "An Operation-Server Based Data Acquisition System Architecture," *IEEE Transactions on Nuclear Science*, vol. 55, no. 1, pp. 217–219, 2008.

[60] R. Blahut, *Algebraic codes for data transmission.* Cambridge University Press, 2003.

[61] G. Bolognini, J. Park, P. Kim, D. Lee, F. Di Pasquale, and N. Park, "Performance enhancement of Raman-based distributed temperature sensors using simplex codes," in *Proc. and the 2006 National Fiber Optic Engineers Conference Optical Fiber Communication Conference OFC 2006*, 2006, p. 3.

[62] M. A. Soto, P. K. Sahu, S. Faralli, G. Bolognini, F. Di Pasquale, B. Nebendahl, and C. Rueck, "Distributed temperature sensor system based on Raman scattering using correlation-codes," *Electronics Letters*, vol. 43, no. 16, pp. 862–864, 2007.

[63] J. Park, G. Bolognini, D. Lee, P. Kim, P. Cho, F. Di Pasquale, and N. Park, "Raman-based distributed temperature sensor with Simplex coding and link optimization," *IEEE Photonics Technology Letters*, vol. 18, no. 17, pp. 1879–1881, 2006.

[64] D. Lee, H. Yoon, N. Y. Kim, H. Lee, and N. Park, "Analysis and experimental demonstration of Simplex coding technique for SNR enhancement of OTDR," *Proceedings of the Lightwave Technologies in Instrumentation and Measurement Conference, 2004*, pp. 118–122, Oct. 2004.

[65] D. Lee, H. Yoon, P. Kim, J. Park, N. Y. Kim, and N. Park, "SNR enhancement of OTDR using biorthogonal codes and generalized inverses," *IEEE Photonics Technology Letters*, vol. 17, no. 1, pp. 163–165, 2005.

[66] M. Harwit and N. J. A. Sloane, *Hadamard Transform Optics*. Academic Press Inc., US, 1979.

[67] G. Bolognini, F. Di Pasquale, F. Baronti, A. Lazzeri, and R. Roncella, Italian Patent Pending PI2010A000 046, April 9, 2010.

[68] H. Song and S. Golomb, "Some new constructions for Simplex codes," *IEEE Transactions on Information Theory*, vol. 40, no. 2, pp. 504–507, Mar 1994.

[69] *E–Team Squadra Corse*, University of Pisa (Italy), www.eteamsquadracorse.it.

[70] A. Turner, K. Ramsay, R. Clark, and D. Howe, "Direct-Drive Electromechanical Linear Actuator for Shift-by-Wire Control of an Automated Transmission," in *Proc. IEEE Vehicle Power and Propulsion Conf., VPPC '06*, Sep. 2006, pp. 1 – 6.

[71] B. Gao, H. Chen, K. Sanada, and Y. Hu, "Design of Clutch-Slip Controller for Automatic Transmission Using Backstepping," *IEEE/ASME Transactions on Mechatronics*, no. 99, pp. 1–11, 2010.

[72] J. Kim and S. B. Choi, "Design and Modeling of a Clutch Actuator System With Self-Energizing Mechanism," *IEEE/ASME Transactions on Mechatronics*, no. 99, pp. 1–14, 2010, doi 10.1109/TMECH.2010.2059034.

[73] P. Tandon, A. Awasthi, B. Mishra, P. Rathore, and R. Shukla, "Design and Simulation of an Intelligent Bicycle Transmission System," *IEEE/ASME Transactions on Mechatronics*, no. 99, pp. 1–9, 2010, doi 10.1109/TMECH.2010.2045431.

[74] D. Iles-Klumpner, I. Serban, and M. Risticevic, "Automotive electrical actuation technologies," *IEEE Vehicle Power and Propulsion Conf., VPPC*, Sept. 2006.

[75] H.-R. Choi and G.-H. Choe, "A Multiobjective Parametric Optimization for Passenger-Car Steering Actuator," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 3, pp. 900–908, Mar. 2010.

[76] Y. Yamaguchi and T. Murakami, "Adaptive Control for Virtual Steering Characteristics on Electric Vehicle Using Steer-by-Wire System," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 5, pp. 1585 – 1594, May 2009.

[77] M. Naidu, S. Gopalakrishnan, and T. Nehl, "Fault-Tolerant Permanent Magnet Motor Drive Topologies for Automotive X-By-Wire Systems," *IEEE Transactions on Industry Applications*, vol. 46, no. 2, pp. 841 – 848, Mar. 2010.

[78] F. Baronti, E. Petri, S. Saponara, L. Fanucci, R. Roncella, R. Saletti, P. D'Abramo, and R. Serventi, "Design and Verification of Hardware Building Blocks for High-Speed and Fault-Tolerant In-vehicle Networks," *IEEE Transactions on Industrial Electronics*, vol. 58, 2011.

[79] A. Turner, K. Ramsay, R. Clark, and D. Howe, "Direct-Drive Rotary-Linear Electromechanical Actuation System for Control of Gearshifts in Automated Transmissions," in *Proc. IEEE Vehicle Power and Propulsion Conf., VPPC 2007*, Sep. 2007, pp. 267 – 272.

[80] Y.-P. Yang, J.-J. Liu, T.-J. Wang, K.-C. Kuo, and P.-E. Hsu, "An Electric Gearshift With Ultracapacitors for the Power Train of an Electric Vehicle With a Directly Driven Wheel Motor," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 5, pp. 2421 – 2431, Sep. 2007.

[81] M. Lindner and T. Tille, "Design of highly integrated mechatronic gear selector levers for automotive shift-by-wire systems," *IEEE/ASME Transactions on Mechatronics*, vol. 15, no. 6, pp. 961 – 968, Jun. 2010.

[82] I. Martins, J. Esteves, G. Marques, and F. P. da Silva, "Permanent-magnets linear actuator applicability in automotive active suspension," *IEEE Trans. on Vehicular Technology*, vol. 55, no. 1, pp. 86–94, 2006.

[83] H. Langjord and T. A. Johansen, "Dual-mode switched control of an electropneumatic clutch actuator," *IEEE/ASME Transactions on Mechatronics*, vol. 15, no. 6, pp. 969–981, 2010.

[84] F. Khoucha, S. Lagoun, K. Marouani, A. Kheloui, and M. Benbouzid, "Hybrid Cascaded H-Bridge Multilevel-Inverter Induction-Motor-Drive Direct Torque Control for Automotive Applications," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 3, pp. 892 – 899, Mar. 2010.

[85] A. Madni, J. Vuong, M. Lopez, and R. Wells, "A smart linear actuator for fuel management system," in *Proc. 5th IEEE Biannual World Automation Congress*, vol. 14, 2002, pp. 615 – 624.

[86] A. Babinski and T.-C. Tsao, "Acceleration feedback design for voice coil actuated direct drive," in *Proc. IEEE American Control Conf.*, vol. 5, 1999, pp. 3713 – 3717.

[87] M. Khandaker, H. Hong, and L. Rodrigues, "Modeling and controller design for a voice coil actuated engine valve," in *Proc. IEEE Conf. Control Applications, CCA 2005*, Aug. 2005, pp. 1234 – 1239.

[88] W. Jones, "Easy ride: Bose Corp. uses speaker technology to give cars adaptive suspension," *IEEE Spectrum*, vol. 42, no. 5, pp. 12 – 14, May 2005.

[89] C. Larouci, E. Dehondt, A. Harakat, and G. Feld, "Modeling and Control of the Vehicle Transmission System Using Electric Actuators; Integration of a Clutch," in *Proc. IEEE Int. Symp. Industrial Electronics, ISIE 2007*, 4-7 2007, pp. 2202 – 2207.

[90] C. S. Liu, P. D. Lin, P. H. Lin, S. S. Ke, Y. H. Chang, and J. B. Horng, "Design and Characterization of Miniature Auto-Focusing Voice Coil Motor Actuator for Cell Phone Camera Applications," *IEEE Transactions on Magnetics*, vol. 45, no. 1, pp. 155 – 159, Jan. 2009.

[91] M. G. Song, H. W. Baek, N. C. Park, K. S. Park, T. Yoon, Y. P. Park, and S. C. Lim, "Development of Small Sized Actuator With Compliant Mechanism for Optical Image Stabilization," *IEEE Transactions on Magnetics*, vol. 46, no. 6, pp. 2369 – 2372, Jun. 2010.

[92] C. K. Pang, F. Lewis, S. Ge, G. Guo, B. Chen, and T. H. Lee, "Singular Perturbation Control for Vibration Rejection in HDDs Using the PZT Active Suspension as Fast Subsystem Observer," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 3, pp. 1375 – 1386, Jun. 2007.

[93] R. Oboe, F. Marcassa, and G. Maiocchi, "Hard disk drive with voltage-driven voice coil motor and model-based control," *IEEE Transactions on Magnetics*, vol. 41, no. 2, pp. 784 – 790, Feb. 2005.

[94] J. Zheng, M. Fu, Y. Wang, and C. Du, "Nonlinear Tracking Control for a Hard Disk Drive Dual-Stage Actuator System," *IEEE/ASME Transactions on Mechatronics*, vol. 13, no. 5, pp. 510 – 518, Oct. 2008.

[95] P. P. Chao, C. W. Chiu, and Y. H. Pang, "Magneto-Electrodynamical Modeling and Design of a Microspeaker Used for Mobile Phones With Considerations of Diaphragm Corrugation and Air Closures," *IEEE Transactions on Magnetics*, vol. 43, no. 6, pp. 2585 – 2587, Jun. 2007.

[96]   Y. D. Chen, C. C. Fuh, and P. C. Tung, "Application of voice coil motors in active dynamic vibration absorbers," *IEEE Transactions on Magnetics*, vol. 41, no. 3, pp. 1149 – 1154, Mar. 2005.

[97]   L. Li, "Copper loss reduction of a voice-coil motor for cutting tool positioning using an auxiliary rotary motor," *IEEE Transactions on Industrial Electronics*, vol. 46, no. 1, pp. 227 – 229, Feb. 1999.

[98]   M. Jiang, W. Chen, Y. Zhang, L. Chen, and H. Zhang, "Multi-domain modeling and simulation of clutch actuation system," in *Proc. IEEE Intelligent Vehicles Symposium*, Jun. 2009, pp. 1365 – 1370.

[99]   C. H. Ping, "Shift-Time Limited Acceleration: Final Drive Ratios in Formula SAE," in *Proc. SAE Motorsports Engineering Conf. and Exhibition*, 2004.

[100] A. Grancharova and T. A. Johansen, "Design and Comparison of Explicit Model Predictive Controllers for an Electropneumatic Clutch Actuator Using On/Off Valves," *IEEE/ASME Transactions on Mechatronics*, no. 99, pp. 1–9, 2010, doi 10.1109/TMECH.2010.2049365.

[101] *TJA1040 datasheet*, NXP Semiconductors, Eindhoven (NL), downloadable at www.nxp.com.

List of Figures

List of Tables

Publications

Journals

1. F. Baronti, A. Lazzeri, R. Roncella, R. Saletti, "FPGA/DSP-based Implementation of a High-Performance Multi-Channel Counter," *Journal of Systems Architecture*, 2009, **55**, (5-6).

2. F. Baronti, A. Lazzeri, R. Roncella, R. Saletti, A. Signorini, M. A. Soto, G. Bolognini, F. Di Pasquale, "SNR enhancement of Raman-based long-range distributed temperature sensors using cyclic Simplex codes," *Electronic Letters*, 2010, **46**, (17).

3. F. Baronti, A. Lazzeri, R. Roncella, R. Saletti, S. Saponara, "Design and Characterization of a Robotized Gearbox System based on Voice Coil Actuators for a Formula SAE race car," submitted to *IEEE/ASME Transactions on Mechatronics*, 2011.

Conference proceedings

4. D. Audino, F. Baronti, A. Lazzeri, R. Roncella, R. Saletti, "FPGA/DSP-based Configurable Multi-Channel Counter," *10th Euromicro Conf. on Digital System Design Architecture, Methods and Tools*, DSD2007, Lubeck, 2007.

5. F. Baronti, A. Lazzeri, R. Roncella, M. Rubertelli, A. Tomasi, "An electronic barrier system to improve blood transfusion safety," *15th IEEE Int. Conf. on Electronics, Circuits and Systems*, ICECS2008, Malta, 2008.

6. S. Saponara, F. Baronti, E. Petri, F. Lenzi, A. Lazzeri, R. Roncella, L. Fanucci, R. Saletti, P. Terreni, B. Neri, "Electronic Circuits and Systems for Sustainability and Safety Improvement of Road Transport," GE2009, pp 1-2, Trento, 2009.

7.   F. Baronti, A. Lazzeri, F. Lenzi, R. Roncella, R. Saletti, "Electronic Control System of a Formula SAE Gearbox Using Electric Voice Coil Actuators," *9th SAE Int. Conf. on Engines & Vehicles*, ICE2009, Naples, 2009.

8.   F. Baronti, A. Lazzeri, F. Lenzi, R. Roncella, R. Saletti, "Voice Coil Actuators: from Model and Simulation to Automotive Application," *35th Annual Conf. of the IEEE Industrial Electronics Society*, IECON09, Porto, 2009.

9.   A. Signorini, S. Faralli, M. Soto, G. Sacchi, F. Baronti, R. Barsacchi, A. Lazzeri, R. Roncella, G. Bolognini, F. Di Pasquale, "40 km long-range Raman-based distributed temperature sensor with meter-scale spatial resolution," *Conf. on Optical Fiber Communication* (OFC), OFC2010, San Diego, 2010.

10.   F. Baronti, A. Lazzeri, R. Roncella, R. Saletti, "Firmware/Software Platform for Rapid Development of PC-based Data Acquisition Systems," *17th IEEE Int. Conf. on Electronics, Circuits, and Systems*, ICECS2010, Athens, 2010.

11.   M. Soto, T. Nannipieri, A. Signorini, A. Lazzeri, F. Baronti, R. Roncella, G. Bolognini, and F. Di Pasquale "Advanced cyclic coding technique for long–range meter–scale Raman DTS systems on standard single–mode fibers," submitted to *21st Int. Conf. on Optical Fiber Sensors*, OFS21, Ottawa, 2011.

Patents

12.   F. Baronti, G. Bolognini, F. Di Pasquale, A. Lazzeri, R. Roncella, "Apparato di Misura," *Brevetto Industriale Italiano*, PI2010A000046, 9 Apr 2010.