Politecnico di Torino

MASTER OF SCIENCE IN AUTOMOTIVE ENGINEERING

Master's degree thesis

Transmission durability embedded system

Mentors:
Prof.ssa Cristiana Delprete
Ing. Gianluca Camboni
Ing. Martina Ferraccioli
Ing. Riccardo Aguzzi

Candidate:
Giorgio Concu

Academic Year 2017-2018

Ai miei genitori, a Benedetta,
a Bea, per aver creduto
sempre in me.

This page intentionally left blank

Contents

List of Figures

List of Tables

Abstract

Over the past 30 years there has been a heightened interest in improving quality, productivity, and reliability of manufactured products in the ground vehicle industry due to global competition and higher customer demands for safety, durability and reliability of the products. As a results, these products must be designed and tested for sufficient fatigue resistance over a large range of product populations so that the scatters of the product strength and loading have to be qualified for any reliability analysis. To make matters even more complicated, a global company has to deal with the market diversification. Consequently, the mission profile for validation tests have to be differentiated depending on the country or region requirements, in terms of reliability and driving style.

In such complex environment, are increasingly important the testing methodologies and standards adopted in the validation procedure of a new component. They have to assure the same quality and reliability level independently from the destination market, but they have also to be standardized and efficient as possible. Current testing methodology is a relevant time consuming activity and, furthermore, it implies a non-negligible economical effort for what concern the vehicle test equipment. Consequently, all these constraints reflects in a limited number of available test.

The Transmission Durability Embedded System (TDES) represents the answer to these necessities. The TDES is a tool intended to acquire different analog signals measured from the sensors positioned in an automotive transmission. The acquired data have to be elaborated in real-time in order to compute typical fatigue life parameters. It is composed essentially by three subsystems: a Transmission Sensor Module (TSM), a Transmission Black Box (TBB) and a Transmission Durability Software (TDS). The aim of the TDES is to reduce the economical effort deriving from a validation test by using low cost hardware and to cut the deadlines. Obviously, all these considerations allow the extension of such methodology to a large test population, i.e. a more meaningful analysis can be carried out.

Preface

Chapter 1 is first presented to address the importance of sufficient knowledge of service loads/stresses and how to measure these loads/stresses. The service loads have significant effects on the results of fatigue analysis and therefore accurate measurements of the actual service loads are necessary. The first portion of the chapter is focused on the strain gage as a transducer of the accurate the accurate measurement of the strain/stress, which is the most significant predictor of fatigue life analysis. Afterwards, attention is paid to other important factors influencing the fatigue life prediction, such as rotational speed and temperature, measured by dedicated sensors.

In Chapter 2 is described the practical application of the measurement techniques presented in the Chapter 1 for a mission profile analysis in case of a dry dual clutch transmission (DDCT) for passenger cars.

Chapter 3 is dedicated to the cycle counting method used to reduce a complicated loading time history into a series of simple constant amplitude loads that can be associated with the equivalent damage. Moreover the rotating moment histogram (RMH) theory is introduced together with the equivalent torque and damage model used for the fatigue life analysis.

In Chapter 4 the numerical implementations of the methodologies explained in Chapter 3 are presented. Furthermore an investigation about the sampling frequency used in analog signal acquisition is performed for meeting performance and resource target requirement.

In Chapter 5 the development and the implementation of the logics governing the operation of embedded system are addressed: starting from the acquisition of an analog input up to the completion of the data logger scheme.

In Chapter 6 the validation test procedure is illustrated and the experimental results are showed.

In Chapter 7 final considerations about the embedded system state of the art and its future development are discussed.

Chapter 1

Transmission measure equipment for mission profile

1.1 Strain gage

The usual way of assessing structural parts of machines, buildings, vehicles, aircraft etc. is based on strength of material calculations.
This method is satisfactory provided the component loads are known both qualitatively and quantitatively. Problems arise particularly where the loads are unknown or where they can only be roughly approximated. Formerly the risk of overloading was countered by using safety margins, i.e. through over-dimensioning. However, modern design strategies demand savings in material, partly for reasons of cost and partly to save weight. In order to satisfy requirements and to provide an adequate component service life, the material stresses must be known. Therefore measurements under operational conditions are necessary.

1.1.1 Fundamentals

A strain gage is a resistive device that experimentally evaluate the load or the strain an object experiences. In any resistance transducer, the resistance R, measured in Ohms [Ω], is a material and geometry dependent property. The resistivity of the material ρ is expressed as resistance per unit length $\times$ area, with cross sectional area A along the length of the material l making up the geometry. Resistance increases with the length and decreases with the cross sectional area for a material of constant resistivity. The expression which correlates all these parameters is the

Equation 1.1:

$$R = \rho \frac{l}{A} \tag{1.1}$$

If the material experiences a mechanical load along its length, as shown in Figure 1.1, all the three parameters (l,ρ,A) change, and, as a result, the end-to-end resistance of the wire changes.

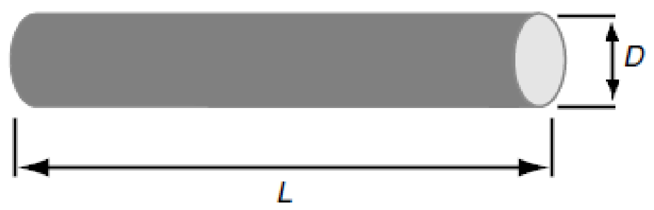

Figure 1.1: A simple resistance wire

The resistance change that occurs in a wire under mechanical load makes it possible to use a wire to measure small dimensional changes that occur because of a change in component loading. In particular it makes possible to evaluate the strain ε, defined as the change in the component length (Δl) over the original length (l), as described by the Equation 1.2 and illustrated in Figure 1.2:

$$\varepsilon = \frac{\Delta l}{l} \tag{1.2}$$

It is possible, with proper bonding of a wire to a structure, to accurately measure the change in length that occurs in the bonded length of the wire. This is the fundamental principle of the strain gage. In a strain gage, the gage grid physically changes length when the material to which it is bonded changes its length. This variation in length, together with the change of the cross sectional area and the change in the resistivity ($\Delta\rho$), implies a change in the wire resistance.

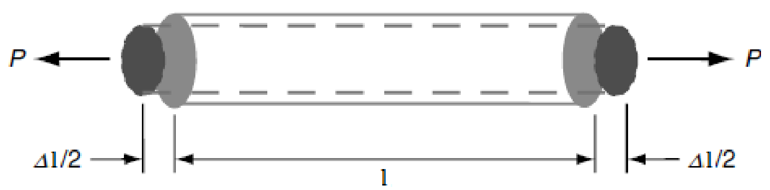

Figure 1.2: A simple wire as a strain gage

The resistance strain gage is convenient because the change in resistance that occurs is directly proportional to the change in length per unit length that the transducer undergoes. In industrial application, two fundamental types of strain gages are available: the wire gage and the etched foil gage. Both types show similar basic design; however the etched foil gage introduces some additional flexibility in the gage design process, providing additional control, such as temperature compensation.
The product of gage width and length defines the active gage area, as shown in Figure 1.3. It is a key parameter because it affects the measurement surface and the power dissipation of the gage.

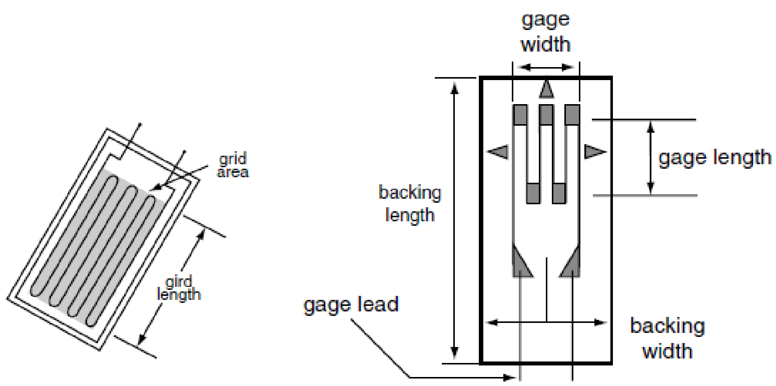

Figure 1.3: Gage dimensional nomenclature

1.1.2 Gage resistance and excitation voltage

Gage nominal resistance is most commonly either 120 Ω or 350 Ω. Higher-resistance gages are available if the application requires either a higher excitation voltage or the material to which it is attached has low heat conductivity. Increasing the gage resistance allows increased excitation levels (V) with an equivalent power dissipation P_w requirement as shown in Equation . With analog-to-digital (A-D) conversion for processing in computers, a commonly used excitation voltage is 10 V.

$$P_w = \frac{V^2}{R} \tag{1.3}$$

High excitation voltage leads to higher signal-to-noise ratios and increase the power dissipation requirement which is especially important in case of small grids.

1.1.3 Gage length

The gage averages the strain field over the length l of the grid. If the gage is mounted on a non-uniform stress field the average strain to which the active gage area is exposed is proportional to the resistance change. If a strain field is known to be non-uniform, proper location of the smallest grid is frequently the best option, as shown in Figure 1.4.

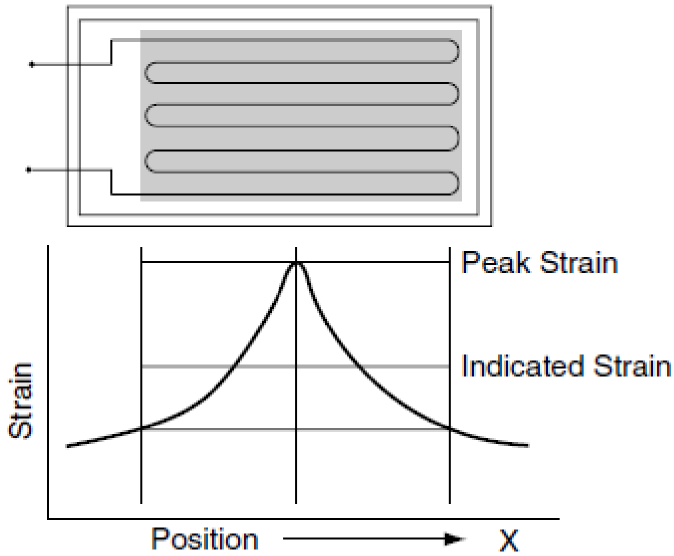

Figure 1.4: Peak and indicated strain comparison

1.1.4 Gage material

Gage material from which the grid is made is usually constantan. The material used depends on the application, the material to which is bonded, and the control required. If the gage material is perfectly matched to the mechanical characteristic of the material to which is bonded, the gage can have pseudo temperature compensation with the gage dimensional changes offsetting the temperature-related component changes. The gage itself will be temperature compensated if the gage material selected has a thermal coefficient of resistivity of zero over temperature working condition range. If the gage has both mechanical and temperature compensation, the system will not produce apparent strain as a result of ambient temperature variations in the testing environment. Selection of the proper gage material that has a minimal temperature-dependent resistivity and some temperature-dependent mechanical characteristics can result in a gage system with minimum sensitivity to temperature changes. A common choice in the industrial field is to use either aluminium or steel, which provide acceptable temperature compensation for ambient temperature variations. The major function of the strain gage is to produce a resistance change proportional to the mechanical strain experienced by the object to which it is mounted. The gage proportionality factor, commonly called gage factor (K), makes possible the equivalence between the Equations 1.2 and 1.4, resulting in Equation 1.5

$$\varepsilon \propto \frac{\Delta R}{R} \tag{1.4}$$

$$\frac{\Delta R}{R} = K \varepsilon R \tag{1.5}$$

The gage factor results from the mechanical deformation of the gage grid and the change in resistivity of the material (ρ) due to the mechanical strain.

1.1.5 Wheatstone bridge

The change in resistance that occurs in a typical strain gage is quite small and, generally, to measure changes in resistance is rather difficult. An easier way consists in measuring the voltage change as a result of resistance change and it is always used. A Wheatstone bridge is used to provide the voltage output due to a resistance change at the gage. The strain gage bridge is simply a Wheatstone bridge with the added requirement that either gages of equal resistance or precision resistors be in each arm of the bridge, as shown in Figure 1.5.

The bridge circuit can be viewed as a voltage divider circuit and so each leg of the circuit is exposed to the same excitation voltage (E_{ex}). The current that flows through each leg of the circuit is the excitation voltage divided by the sum of the resistances in the same leg, as expressed in Equation 1.6. If the resistance value of all resistors is equal (R_1=R_2=R_3=R_4=R), the current flow from the source is the excitation voltage divided by R, as shown in Equation 1.7.

$$I_A = \frac{E_{ex}}{(R_1 + R_4)} \qquad I_C = \frac{E_{ex}}{(R_2 + R_3)} \tag{1.6}$$

$$I_{ex} = I_A + I_C = \frac{E_{ex}}{R} \tag{1.7}$$

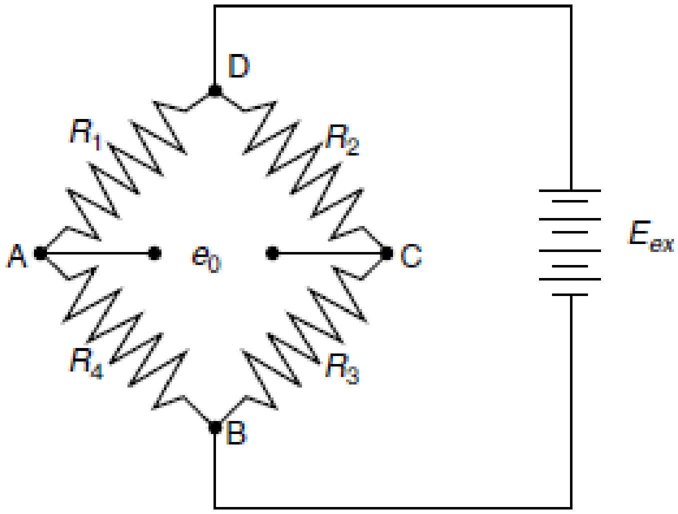

Figure 1.5: A Wheatstone bridge circuit

As a voltage divider circuit, the voltage measured between points A and D and between C and D, at the midpoint, are expressed by Equation 1.8.

$$e_A = \frac{R_4}{R_1 + R_4} E_{ex} \qquad e_C = \frac{R_3}{R_2 + R_3} E_{ex} \tag{1.8}$$

The strain gage exploits the voltage differential measured between points A and C in order to determine the output of the bridge resulting from any imbalance, as shown in Equation 1.9.

$$e_0 = e_A - e_C = E_{ex} \Big[\frac{R_4}{R_1 + R_4} - \frac{R_3}{R_2 + R_3} \Big] \tag{1.9}$$

If the bridge is initially balanced, points A and C are at equal potential and, consequently, $e_0 = 0$. It is worth to notice that the bridge can be balanced without all resistances being equal. The condition that has to be satisfied is expressed in Equation 1.10

$$\frac{R_1}{R_4} = \frac{R_2}{R_3} \tag{1.10}$$

1.1.6 Constant-voltage strain gage bridge output

The Wheatstone bridge allows multiple configurations of the strain gage, in particular it is possible to place one, two or four strain gages in the same circuit, called respectively quarter bridge, half bridge and full bridge. In the quarter bridge configuration, one of the resistors is replaced with a strain gage, whereas the other three arms employ high-precision resistors with a nominal value equal to the strain gage one, as depicted in Figure 1.6. If a half bridge is used, the gages are usually positioned in the bridge such that the greatest unbalance of the bridge is achieved when the gages are exposed to the strain of the part; the remaining arms receive high-precision resistors. If a full bridge is used, all four resistors in the bridge are replaced with active strain gages.

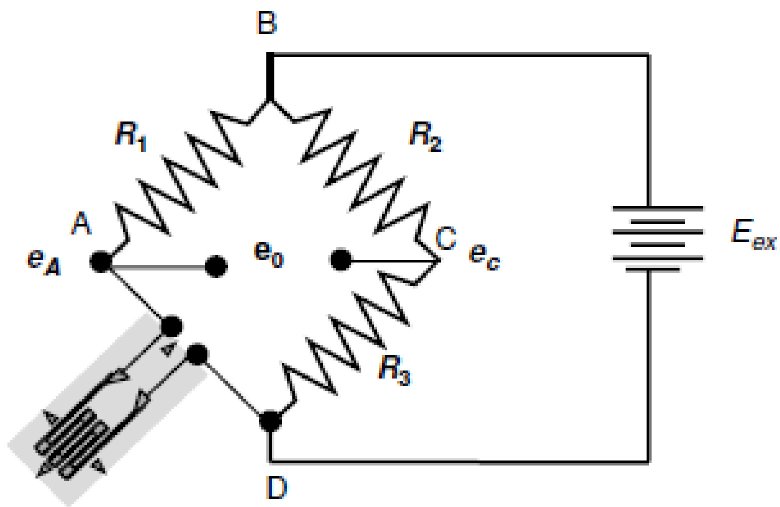

Figure 1.6: a constant-voltage quarter-bridge circuit

1.2 Telemetry system

In today's engineering, it is more and more important to know data from rotating or translatory moving part of a machinery. To optimise the efficiency, to reduce stresses and vibrations, to analyse the behaviour of new materials and to reduce cost and material it is necessary to get information from these parts. To get these information the sensor have to be provided with energy and the sensor's data have to be transmitted to the stationary evaluation unit.

1.2.1 Fundamentals

Telemetry systems are contactless method of transmitting data from the rotating assembly to the stationary data acquisition system: it transmits sensor data bidirectionally while simultaneous transmits power from the evaluation unit to the sensor signal amplifier. Basic telemetry systems consist of a modulator, a voltage-controlled oscillator (VCO) and a power supply for the strain gage bridge. In Figure 1.7 is reported a generic telemetry system equipment for torque measurement purposes.

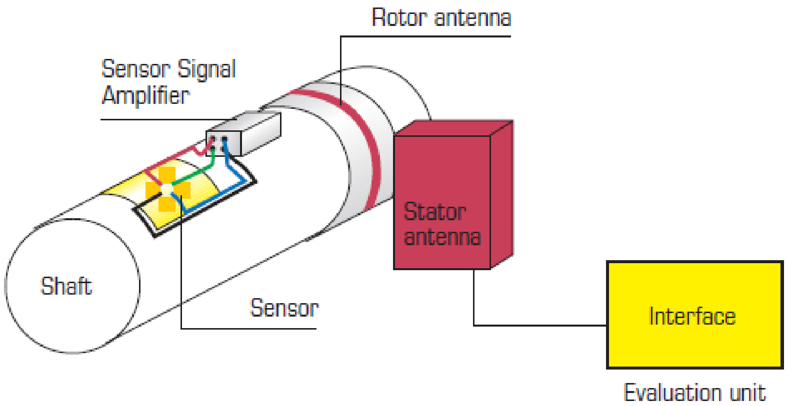

Figure 1.7: Schematic of the sensor telemetry

The signal from the strain gage bridge is used to pulse modulate a constant-amplitude square wave. The output pulse width is proportional to the voltage provided by the strain gag bridge. This square wave serves to vary the frequency of the voltage-controlled oscillator (VCO), which has a centre frequency f_c. The VCO signal is transmitted by the an antenna mounted on the rotating shaft and it is received by a stationary loop antenna. After the signal is received, it is demodulated, filtered and amplified before recording. Most of the transmitting unit of a telemetry system is completely self-contained: the power supply to the components on the rotating shaft is obtained by inductively coupling the power supply through the stationary loop antenna. The sensor signal is mapped to the range of -10÷10 V at the output of the evaluation unit. Consequently, having a resolution of 12 bit, a theoretical resolution of 4.8828125 mV/digit is achieved. The data and power transmission working principles are represented respectively in Figure 1.8 and in Figure 1.9.

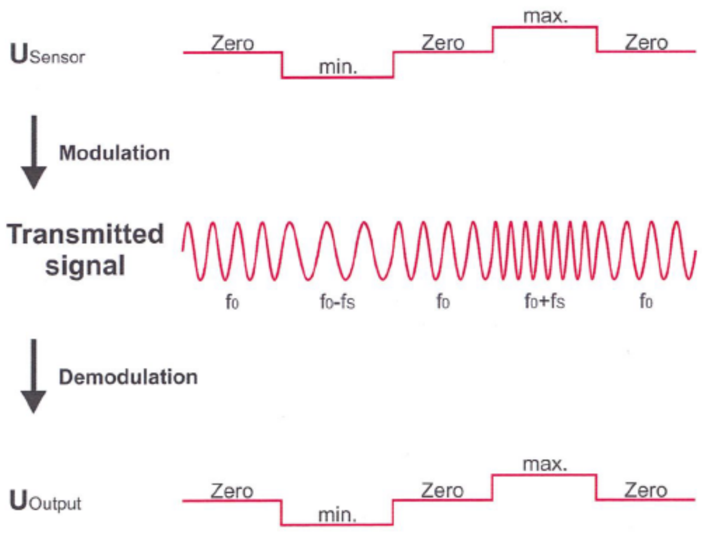

Figure 1.8: Data transmission working principle

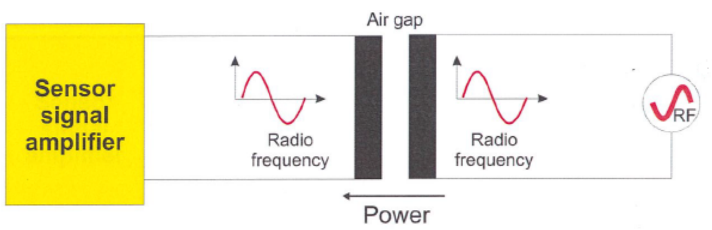

Figure 1.9: Power transmission working principle

1.3 Hall effect sensor

The speed/rpm sensors rely on various operating principles - Hall effect, magnetoresistive, inductive - to detect without contact the rotary movement of phonic or toothed wheels and generally of any rotative device built in a ferrous material and provided with slots or prominent parts. They provide a frequency output signal of digital type - for the Hall effect or magnetoresistive versions - or a sinusoidal wave - for the inductive versions - that follow exactly the alternating sequence of presence and absence of ferrous material presented by the rotative device. In this section only the Hall effect based rpm sensors are investigated.

1.3.1 Fundamentals

When a current-carrying conductor is placed into a magnetic field, a voltage will be generated perpendicular to both the current and the field. Figure 1.10 shows a thin sheet of semiconducting material through which a current is passed. The output connections are perpendicular to the direction of current. When a perpendicular magnetic field is present, a Lorentz force is exerted on the current. This force disturbs the current distribution, resulting in a potential difference across the output. This voltage is the Hall voltage (V_H). The interaction of the magnetic field B and the current I is shown in Equation 1.11.

$$V_H \propto I \times B \tag{1.11}$$

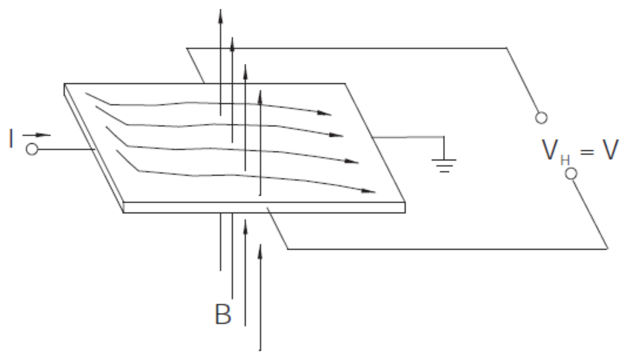

Figure 1.10: Hall effect principle

The Hall voltage is a low-level signal and therefore it requires an amplifier with low noise, high input impedance and moderate gain. Usually a differential amplifier with these characteristics is integrated directly in the Hall effect based sensor.

1.3.2 Hall effect RPM sensor

The Hall effect principle can be exploited in order to compute the rotational speed of a shaft. In particular, the RPM sensor include a wheel that is integral to the considered shaft with several rods, that are angularly equally spaced. Across such a wheel, the sensing element is placed radially at a distance lower than 2 mm and it generates a a different output voltage depending on the fact that a rod is passing in front of the sensor or not. As a consequence, when the wheel rotates, the sensor output is a square wave that is the input for a counter. It is good to notice that the output voltage of the Hall effect sensor is not affected by the wheel angular speed and therefore it is well suited to the measure of very low angular speeds.

1.4 Thermocouple

Typical gear failures like wear, scuffing, micropitting and pitting are influenced by the oil temperature in the lubrication system. High temperatures lead to low viscosities and thus thin lubricant films in the gear mesh with generally detrimental influence on failure performance. On the other hand, for gear oils with additives higher temperatures correspond with higher chemical activity and, at least in some cases, with better failure performance of the lubricant. Last, but not least, at very high temperatures even metallurgical changes have been found with a reduction in material endurance limits.

1.4.1 Fundamentals

When an electric conductor is subjected to a thermal gradient, a difference in the electric potential

$$\Delta v = k\Delta T \tag{1.12}$$

arises because of electric charge displacement induced by the temperature difference within the conductor body. This phenomenon, which is known as the *Seebeck effect* is exploited in thermocouples to measure a temperature gradient. To this purpose, two samples of different conductors are soldered together and are placed so that they are subjected to the same thermal gradient ΔT. In particular, one side of both conductors is put in contact with the body at the temperature to be measured T, whereas the other side is at known temperature T_{REF}, so that $\Delta T = T - T_{REF}$, as shown in Figure 1.11. In such a configuration, a voltage

$$v_{DIFF} = \Delta v_2 - \Delta v_1 = (k_2 - k_1)\Delta T = k(T - T_{REF}), \tag{1.13}$$

proportional to the thermal gradient ΔT, arises between the two conductors because of the Seebeck effect. The constant k ranges from 1 $\mu V/°C$ up to 50 $\mu V/°C$, therefore the output voltage signal is amplified by a differential amplifier. Materials for thermocouples are chosen for their temperature range and their sensitivity. The K-type thermocouple is one of the most commonly used in an engineering environment.

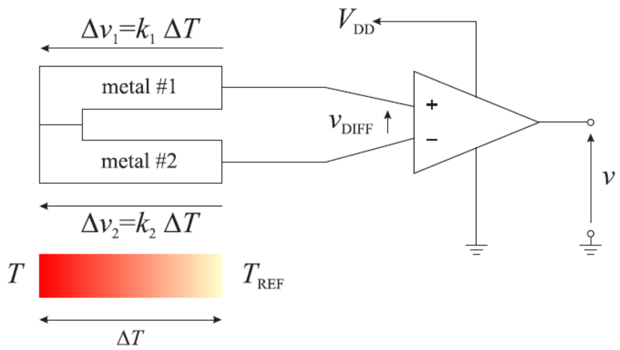

Figure 1.11: Operating principle of a thermocouple

Chapter 2

Transmission sensor module

The Transmission Sensor Module (TSM) has to guarantee functional and durability aspects of the original transmission and, furthermore, it has to provide an elevated measure quality with a maximum acceptable error of 1%. In the following are reported the main required characteristics of the TSM system:

- Performance: the transmission on which it is installed has to have the same performance of a normal production one, e.g. torque strain gage has to be designed to achieve the maximum level of torque/power of the original transmission;
- Overload: the transmission on which it is installed has to have the same resistance to dynamic overload of a normal production one, e.g. the sensors installation has not to interfere with the resistance to dynamic impulsive loads such as over-torque events due to gear shifting;
- Reliability: the transmission on which it is installed has to have the same reliability features of a normal production one, e.g. the installation of the different sensors has not to introduce any oil leakage in the system. The reliability characteristics of the wirings/connectors of the cables have to respect the best standard present on the market;
- Temperature: the maximum and minimum working temperatures of the sensors have to be inside the temperature working range of the transmission and the bonnet where it is installed;
- Accuracy: it has to guarantee a maximum measurement error of 1% in every working point of the transmission;
- Installability: it has to be installed on a specially designed transmission for

the location of the different sensors.

In general, TSM consists of three different typologies of sensors:

- Hall effect based sensor;
- Strain gage;
- Thermocouple.

The TSM is intended to operate with both manual transmission (MT) and dry dual clutch transmission (DDCT). Obviously, depending on the selected configuration, minor adjustments have to be accounted due to the different transmission layouts. For example, for what concerns the rotational speed measurements, the MT configuration expects to have three pick-ups: one positioned on the flywheel side, one on the primary shaft and the third on the final drive crown. Instead, in the case of DDCT configuration, the transmission has to be equipped with an additional pick-up (with respect to the MT layout) in order to acquire the rotational speed of the second primary shaft. In the current practices, it is used to equip the half-shafts with the strain gages for the torque measurement, while in this application the strain gage is putted only in the secondary shaft so that it is possible to have a unique system to be prepared.
For what concern the thermocouples, they register the time history of the air temperature on the flywheel side and the gearbox oil temperature. A synthetic representation of the above described layout for a transversal MT layout is reported in the Figure 2.1.

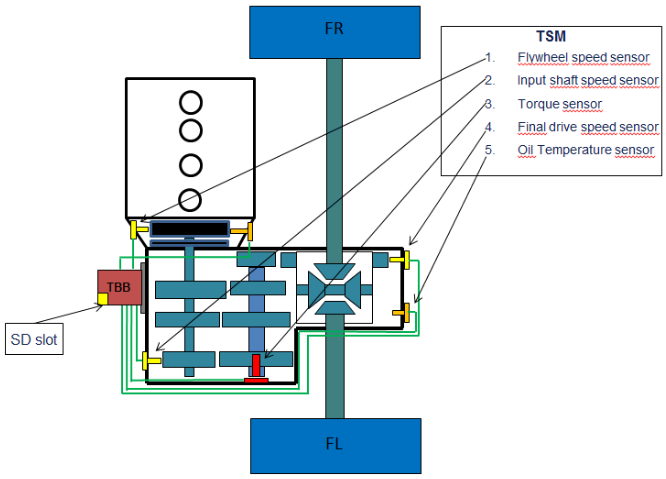

Figure 2.1: TSM layout

2.1 Torque measurement equipment

A shaft loaded with torsion is subject to a biaxial stress state. The principal normal stresses occur at an angle of ±45° to the cylindrical planes (lines running parallel to the longitudinal axis of the shaft). The strain produced by the normal stresses can be measured with strain gages by placing their measuring grid axes at ±45°. Suitable configurations are the half and full bridge circuits. In this application the latter configuration is adopted because the full-bridge configuration is the best in compensating interference signals from superimposed normal and bending loading. In Figure 2.2 is reported the full bridge configuration, obtained by means of two Chevron rosettes, applied on a shaft for torque measurement purposes.
Once the full bridge layout is selected, the specific strain gage has to be carefully chosen by looking at its data sheet. The most significant data related to the strain gages used are reported in Table 2.1.

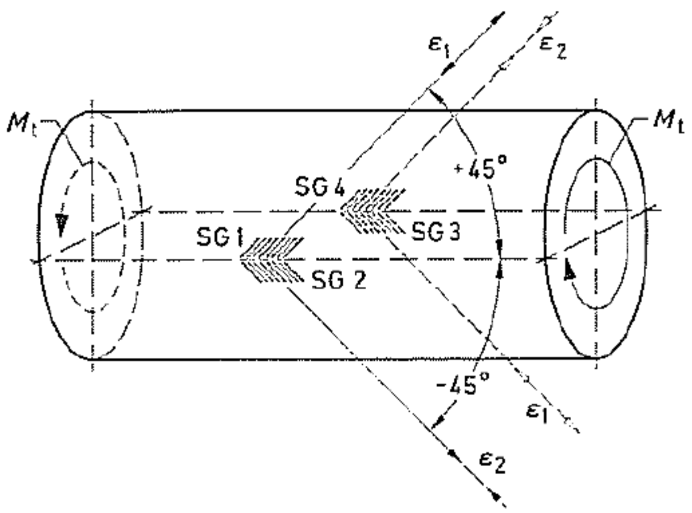

Figure 2.2: Torsion shaft with strain gages mounted in the principal directions ε_1 and ε_2

Gage factor (24°C)	Gage length	Gage resistance (24°C)	Temperature coefficient	Adoptable thermal expansion
2.09 ± 1%	2 mm	350.0 ± 2.4 Ω	+0.008%/°C	11.7 PPM/°C

Table 2.1: Strain gage technical specifications

The most used values for the gage resistance are 120 Ω and 350 Ω. In order to explain the reasons behind this decision, it is worth to underline that the two solutions are examined imposing an equal voltage difference across the strain gage. Considering that, the 350 Ω presents a list of advantages:

- Lower current flowing in the strain gage and, consequently, lower power dissipation due to Joule effect;
- Lower current flowing in the telemetry system;
- Higher excitation voltage allowed;
- Lower influence of wire resistance.

Therefore, the appropriate location for its positioning has to be determined by looking at the gearbox assembly drawing. In Figure 2.3 is reported the transmission assembly with the addition of the torque flows: in blue in case of 4th and in red in case of 1st gear.
Referring to this consideration, it has been decided to put the strain gages below the 4th and 6th gears respectively in the upper and lower secondary shaft, because in such a configuration it is possible to intercept the torque "flowing" in the shaft in every working conditions, i.e. which ever gear ratio is selected by the driver. In order to do so, the shafts have to be modified and below the bearings belonging to the above mentioned gears, a groove is machined. It is important to underline that the groove for the strain gage positioning has the same diameter of the shaft at the left extremity so that it is possible to not introduce any modifications and stress concentration points to the original geometry. In Figure 2.4 it is possible to see the application of a Chevron rosette and its related grid positioned in the designated groove of the secondary shaft.

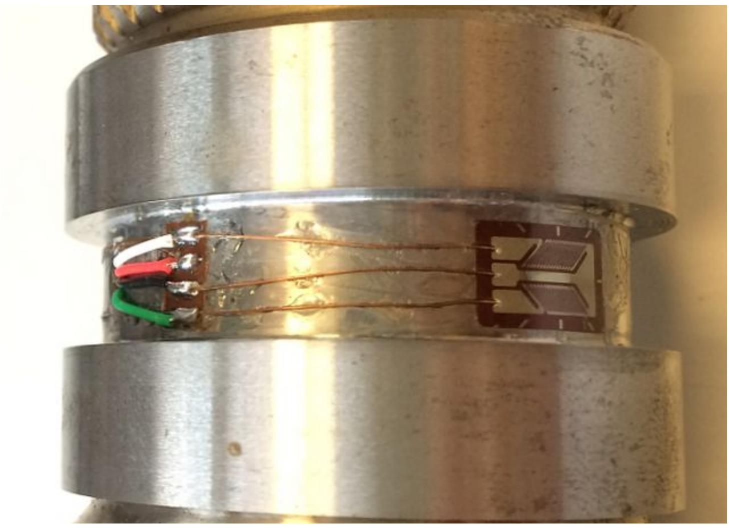

Figure 2.4: A Chevron rosette applied on the secondary shaft

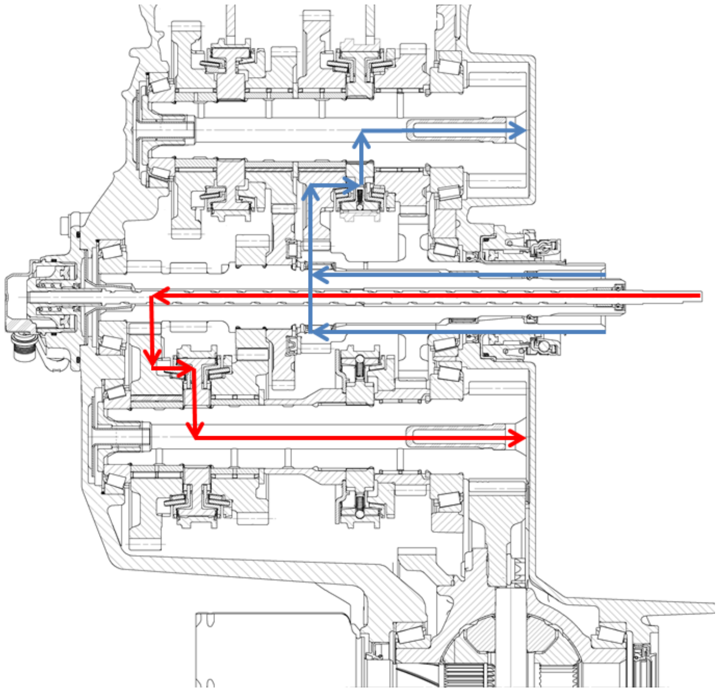

Figure 2.3: DDCT gearbox scheme

2.1.1 Strain gage application

Afterwards the application of the strain gage and the grid has been completed, the next step is to calibrate the output analog signal of the system in order to have an operating voltage range -10 ÷ 10 V. The tare procedure consists in applying a known torsional load to the secondary shaft while it is engaged with the differential ring. The latter is completely blocked by a dedicated clamp and so the product between the force exerted by the suspended weights and the arm results in the torque experienced by the secondary shaft. In particular, after a series of loading-unloading cycles to eliminate deviations coming from hysteresis phenomena, an increasing series of loads have been applied to the secondary shafts with a unitary length arm. Therefore, for each load and arm, the torque applied is computed and the corresponding output voltage is saved. After the load sequence is completed, the loads are removed and the verification that the output voltage drops to zero for null load is performed. In Figure 2.5 is reported the laboratory equipment for the calibration test in which can be seen the clamp used to block the differential crown and the lever mounted to the shaft with the applied weights. Once all the data are collected, the linear regression is computed in order to check the model linearity, as shown in Figure 2.6.

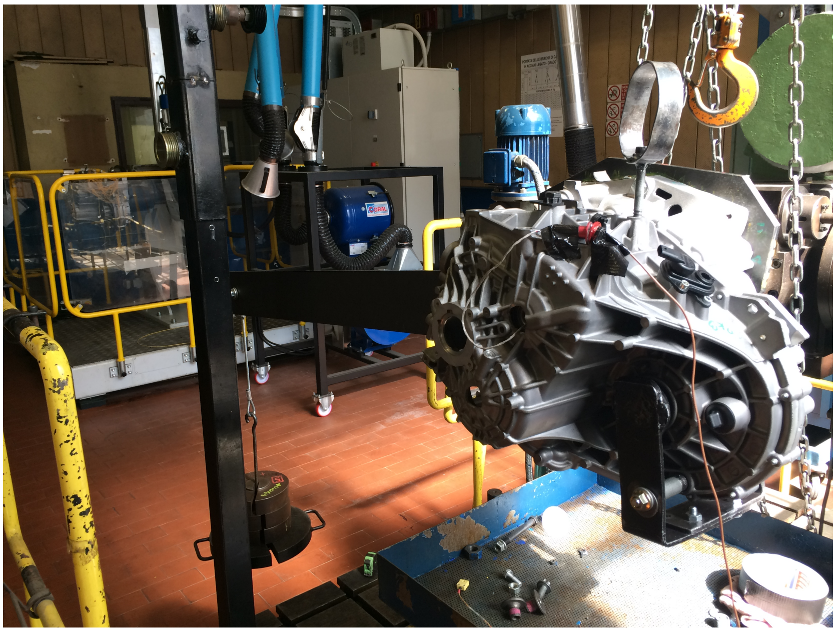

Figure 2.5: Calibration test equipment

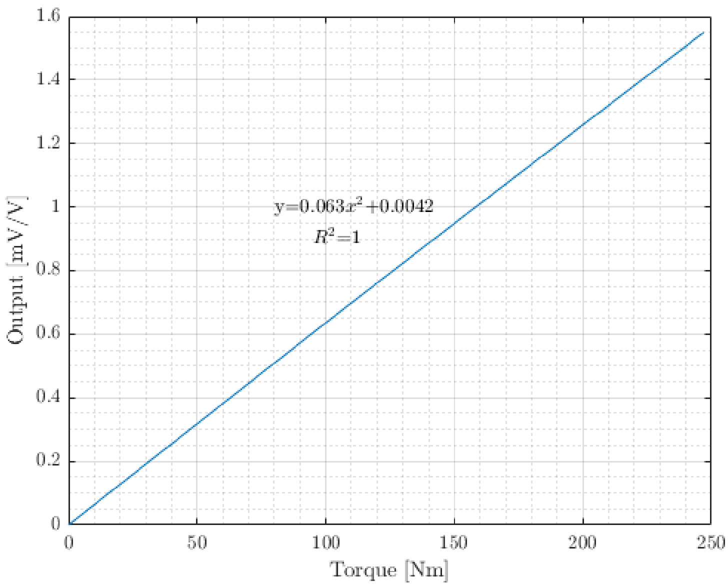

Figure 2.6: Characteristic output voltage-torque

Once the calibration is completed, the entire assembly has to be preserved from direct mechanical load and wear due to normal operating condition in the gearbox;

so a protective film of a bi-component material is spread over the surface to protect it and then cooked in an oven for 2 hours at 80°C. The final result can be seen in Figure 2.7.

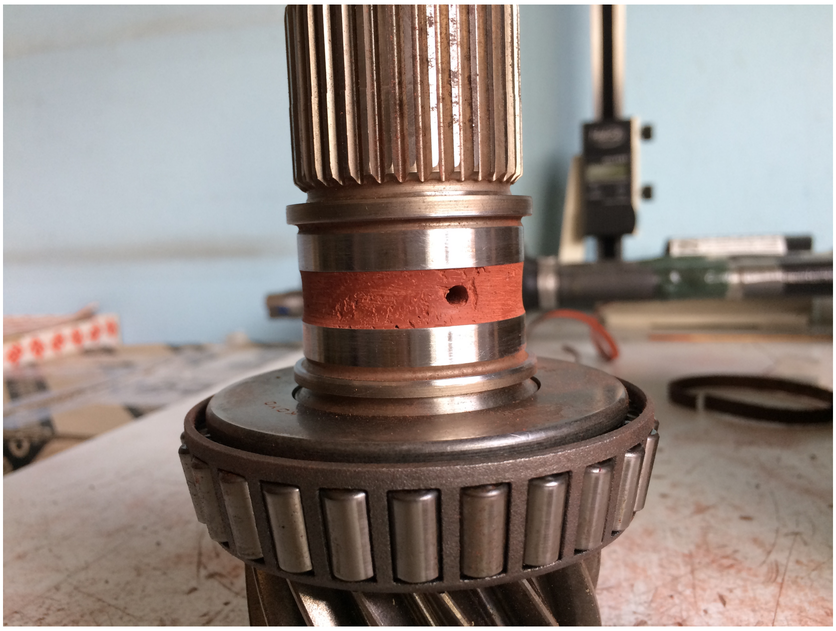

Figure 2.7: Application of protective film of bi-component over the strain gage

2.2 Telemetry equipment

The telemetry system, as described before, consists in three main components:

- a rotor antenna;
- a stator antenna;
- an evaluation unit.

The rotor antenna is mounted directly on the secondary shaft in a specially designed hole and fixed by means of four M2×6 screws directly to the same shaft. In Figure 2.8 is reported the application of the telemetry rotor antenna in the secondary shaft seat.

Figure 2.8: Application of the rotor antenna in the secondary shaft

The telemetry system used features a remote controlled programmable sensor (RMC) which is able to assure the following advantages with respect to a traditional evaluation unit without direct accessing to the electronic:

- Different bit resolutions available;
- Zero point compensation;
- Gain adjustment;
- Measurement range adjustment.

2.2.1 Telemetry system application

From the hardware point of view, the telemetry system has to be connected to the strain gage by means of a wire so that it is possible to amplify and transmit the output signal to the receiver, i.e. the stator antenna. In order to perform this connection, the shaft has been modified so that it can include the rotor antenna and the wire connecting the strain gage to the latter. In particular, the original shaft

was hollow and it featured a constant diameter along its longitudinal dimension. The modified version features a larger hole diameter in correspondence of the gear matching the differential crown and four additional threaded holes in order to fix the rotor antenna. The zone affected by these modification is sketched in Figure 2.9, where also the rotor and stator antennas are reported.
The secondary shafts are not the only components which has been modified in order to fit the telemetry system. In fact, the gearbox casing has been modified too so that it is possible to place the stator antenna. This second modification was necessary because, in order to operate properly, the rotor antenna, positioned on the secondary shaft, requires a maximum distance in axial direction not greater than 2 mm from the stator antenna. In Figure 2.10 is showed the positioning of the two stator antennas in the transmission case.

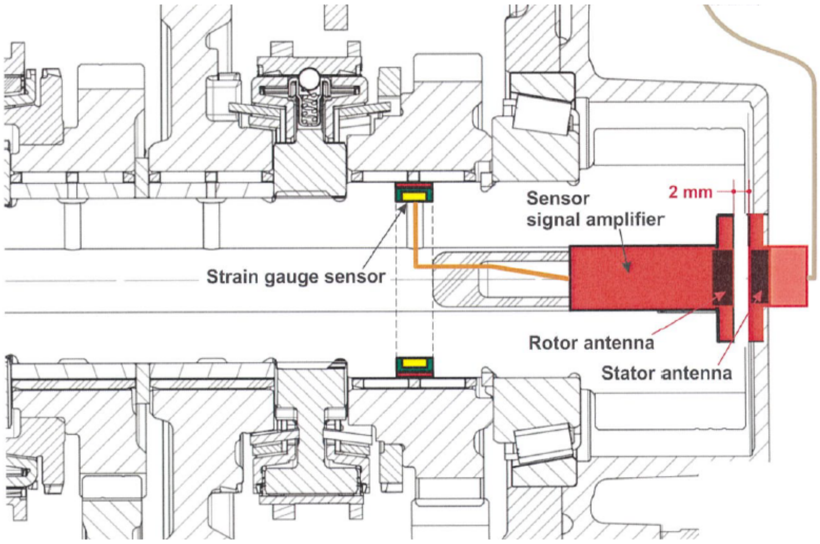

Figure 2.9: Secondary shaft modifications

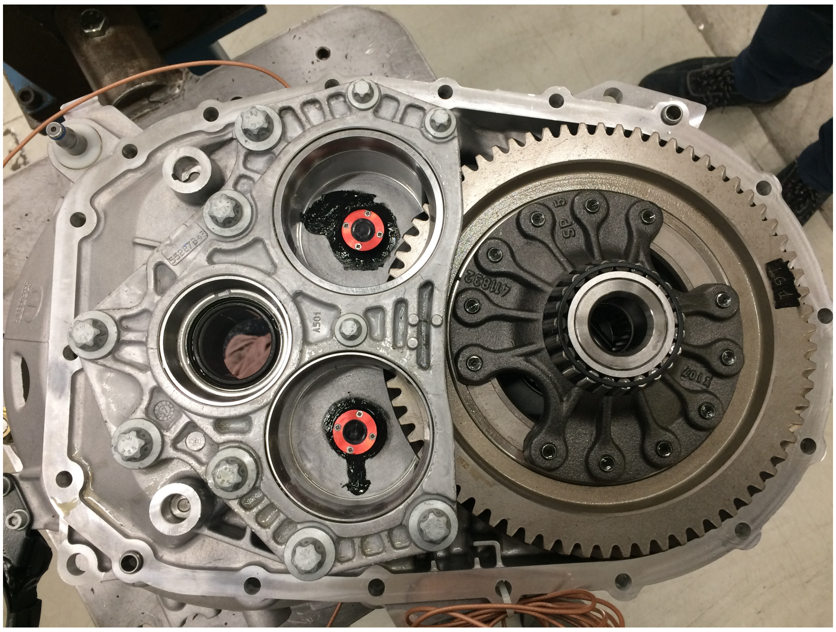

Figure 2.10: Application of the stator antennas in the gearbox case seats

The impact of these modifications have be proven to be minimal and not affecting the performance of the system with respect the original version. In particular, two different structural verifications have been performed: first the shafts deflection has been analysed and then the casing stress distribution have been recomputed in worst case condition. None of the two verifications showed particular criticisms with respect to the original version. In fact, in Figure 2.11 the case stress distribution is reported and can be noticed that the maximum stress is well below the yield strength of the material.

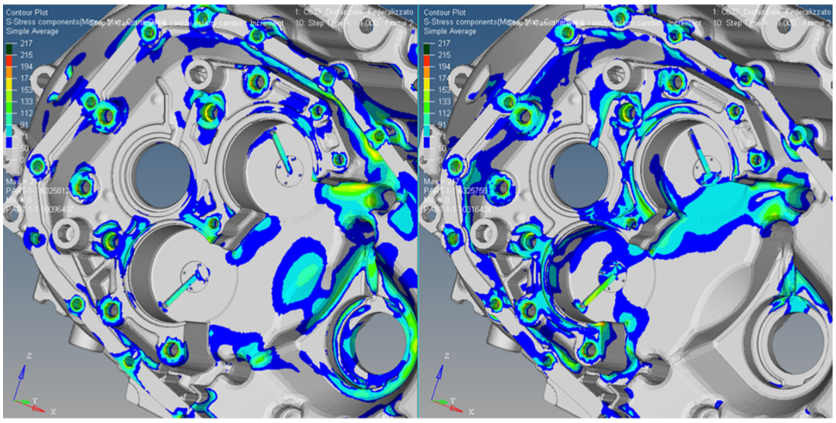

Figure 2.11: Case stress distribution after modifications

2.3 Speed sensor equipment

The Hall effect based sensor is used in the Transmission Sensor Module (TSM) in order to compute the rotational speed of the mechanical components such as:

- Flywheel;
- Primary shaft;
- Final drive.

In case of DDCT configuration, the TSM consists of an additional speed sensor in order to monitor the rotational speed of the second primary shaft too.
In the first implementation of the TDES it has been decided to use only one Hall effect gear tooth speed sensor in charge of the differential rotational speed measurement, since the differential gear RMH is the only output expected.
The pick-up designated to accomplish this task is a GS-1005. It is a Hall effect based gear tooth speed sensor which features an adjustable anodized aluminium housing. Its main features are:

- From near zero up to 15 kHz sensing capability;
- Compatibility with unregulated power supply;
- IP67 certification;
- Maximum sensing air gap of 2 mm.

Furthermore, the M12 threaded aluminium housing allows the pick-up to be easily positioned in the gearbox case in order to be as close as possible to the differential gear's tooth. In particular, it can be mounted by simply drilling a threaded hole in the gearbox case. In Figure 2.12 is reported the Hall effect based sensor technical drawing.

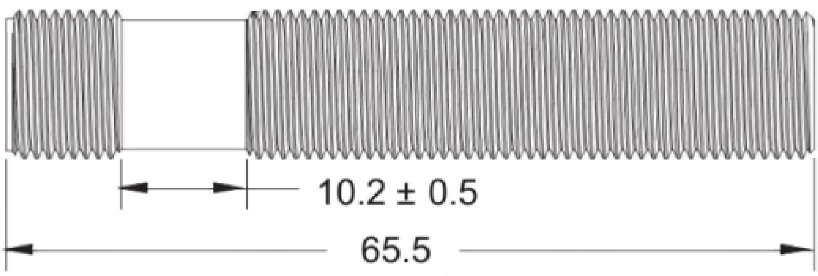

Figure 2.12: ZF GS-1005 technical drawing

In order to choose the best location for the sensor positioning, the CAD model of the gearbox assembly is used. In Figure 2.13 is reported the gearbox CAD assembly and the indication of the cross-section (A-A) used so that it is possible to compute the radial distance between the differential gear and the casing in which the speed sensor has to be placed.

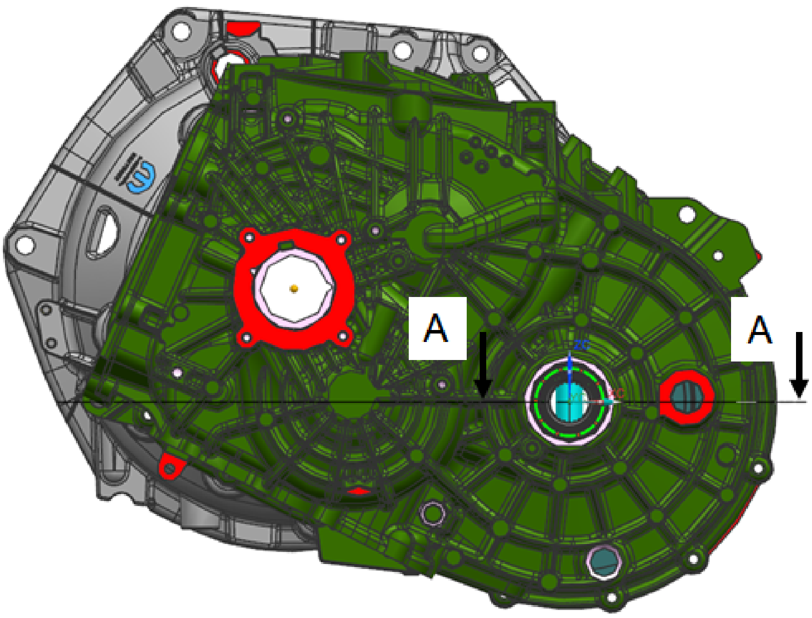

Figure 2.13: Gearbox CAD assembly

Once the positioning location for the speed sensor has been decided, a M12 threaded hole is drilled in the gearbox casing. Afterwards, the Hall effect based sensor is screwed up and its correct functioning in verified by means of an oscilloscope. In Figure 2.14 is reported the application of the pick-up sensor into the designated hole of the transmission case.

Figure 2.14: Differential gear pick up positioning

2.4 Thermocouple equipment

The last category of measuring devices featuring the TSM is the temperature sensor. In engineering and specially in automotive field, the most commonly employed thermocouples are of the K-type, since they are inexpensive, accurate and reliable. Furthermore they have a wide temperature range and can operate in oxidizing environments too. For all the above mentioned features, it has been choose to use a K-type thermocouple for the gearbox oil temperature measurements. In order to accomplish this task the thermocouple sensor has to be putted in direct contact with the oil. Therefore, it is integrated in the drain plug of the transmission oil and then it is screwed in the gearbox case. In Figure 2.15 is reported the assembly of the K-type thermocouple inside the oil plug together with its typical connector.

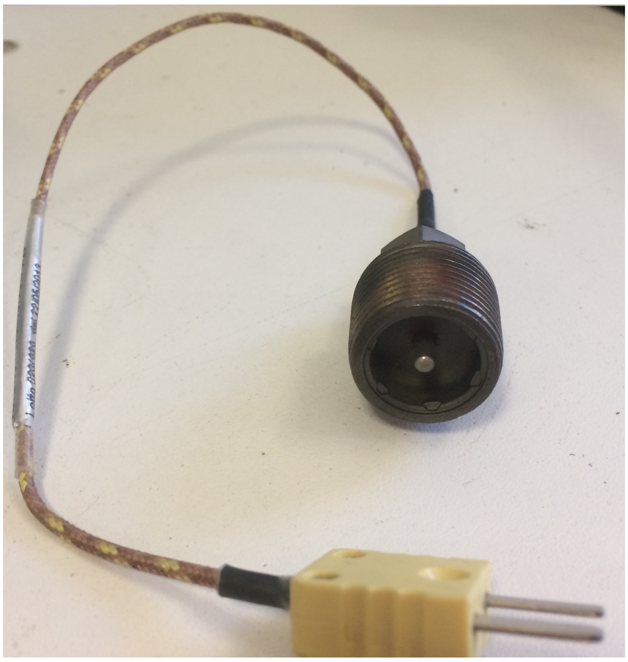

Figure 2.15: Thermocouple assembly

Chapter 3

Fundamentals of cycle counting method and damage cumulation

Real-life structures, such as airplanes, automobiles, ships, pressure vessels, bridges, offshore structure, etc. are often subject to cyclical loads that results in structural failure due to fatigue. To avoid any potential fatigue failure, the fatigue life of the structure must be known. The loads that cause structural failure are usually complex and random in form. Current approaches to random load fatigue analysis utilize experimentally obtained fatigue data, which are often acquired through constant amplitude testing methods, such as stress-life (S-N), strain-life (ε-N), or fatigue crack growth rate, to predict fatigue lives of actual structures subject to random loading. All these commonly used approaches require that stress ranges and cycles of the random load history be defined in order to perform fatigue-life calculation.
Except for constant-amplitude and narrow-band random loadings, where the precise definition of a cycle is clear, the determination of the stress ranges and cycles is a problem. Thus, before any fatigue analysis can be performed, some sort of cycle-counting method to reduce the random load history to proper stress ranges and cycles must be devised. Once a cycle counting method is established, and the stress ranges and cycles are defined, evaluation of the fatigue damage under random loading is possible.

3.1 Rotating moment histogram

Product validation tests are essential during the last design stages of product development. In automotive field, fatigue test are carried out as accelerated test, in order to reproduce the fatigue damage and failure modes from the real proving ground (PG) testing. Obviously, PG loading has to be represented as accurate as possible: the loading has to be measured by driving an instrumented vehicle over the same PG with various test drivers. In this case, instrumentation belongs only to the gearbox components: speed sensor, shaft torque measurement and thermocouples. In automotive field, a common way to represent the PG loading is the Rotating Moment Histogram (RMH), also called Torque Spectrum. The latter is an application of range-based counting methods to rotating components such as shafts, bearings and toothed gears. In order to construct it simultaneous measurements of both torque and rotational speed are needed. The goal of this representation is to count the revolutions (n_i) at a fixed torque level (Ti). Referring to torque and rotational speed time history reported in Figure 3.1, the number of revolutions at T_i can be computed as follows:

$$n_i = \sum_{i=1}^{n} \int_{\Delta t_i} rpm(t)dt \tag{3.1}$$

To make statistical analysis possible, the RMH for each driver must have the same bin limits, therefore resulting in identical bin size and torque value at each bin.

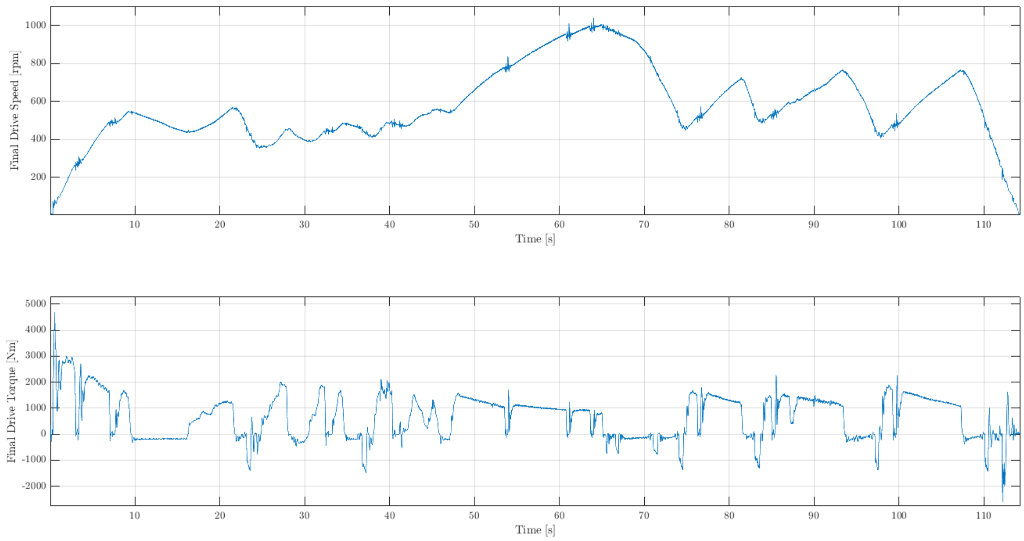

Figure 3.1: Example of a torque and speed time history

3.1.1 RMH cycle extrapolation

The purpose of the RMH cycle extrapolation is to estimate the RMH for a much longer time period based on the short-term load measurement. The method for including loading variability in longer times is to shift the RMH upward with an extrapolated factor for higher numbers of cycles and then to extrapolate the loading spectrum to the two extremes for higher torque estimates. The extrapolation technique for higher torques can be approximately done by graphically extending the data backward and intercepting with the torque axis at one revolution. A check needs to be performed to determine whether the estimated maximum torque value meets or exceeds the physical limit of the driveline system. In Figure 3.2, the x-axis is a log scale of the number of differential revolutions calculated and the y-axis is for the torque values. Any point on the histograms represents a certain number of revolutions found at the corresponding torque level.

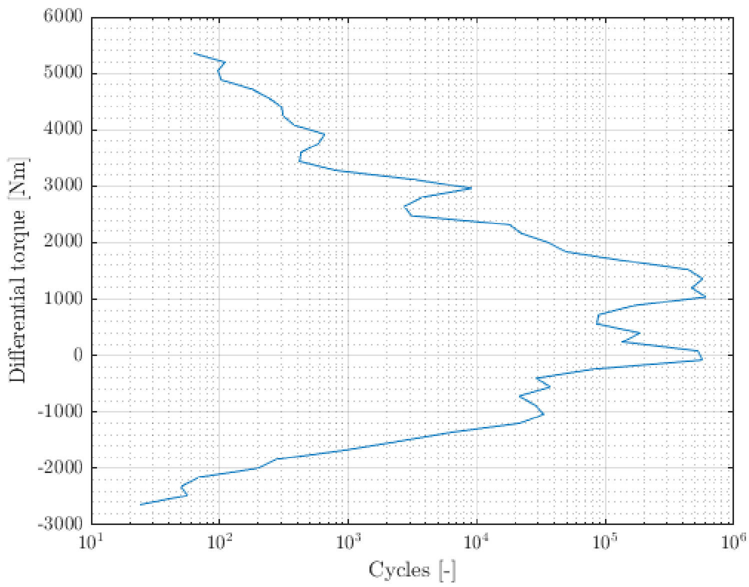

Figure 3.2: Example of a torque and speed time history

3.1.2 RMH quantile extrapolation

The purpose of the RMH quantile extrapolation is to predict the RMH of the single most damaging driver that would exist in a much larger set of data, based on a set of several time history measurements due to driver variability. It is assumed that the damage generated by a loading history. This implies that the most damaging RMH contains the most revolutions.

When the sample size is six or more, the median and the 95th percentile histograms can be generated with an assumption that the revolutions in each torque bin follow a certain statistical distribution. However, in the case where the sample size is limited to less than five, a special statistical method is required to calculate the median and the 95th percentile driver profile with reasonable accuracy. Therefore, a process is developed to make statistical analysis possible. First, the total number of revolutions for all drivers is calculated by adding the number of revolutions in each bin. Second, the average number of revolutions is calculated. Next, the number of revolutions for each driver in each bin is normalized, i.e. average revolutions divided by total revolutions. The average customer revolution profile is determined by the following equation:

$$\bar{x} = \frac{1}{n} \sum_{1}^{n} x_i \tag{3.2}$$

Next is the task to determining a 95th percentile customer revolution profile. To do so,the standard deviation (σ) from a given data range (range) and sample size (n_s) can be estimated as follows:

$$\sigma = range \times CF \tag{3.3}$$

where CF is the conversion factor, given for the various values of sample size (Table 3.1). The range is the difference between the largest and the smallest revolutions in each torque bin. The 95th percentile customer revolution profile ($x_{95\%}$) is derived by Equation 3.4:

$$x_{95\%} = \bar{x} + 1.645 \times \sigma \tag{3.4}$$

in which the constant of 1.645 is the 95$\%$ value of the standard distribution.

n_s	CF
2	0.886
3	0.591
4	0.486
5	0.430

Table 3.1: Factors for the standard deviation for a range

3.2 Rainflow counting method

The rainflow counting method is a two-parameters cycle counting method which can faithfully represent variable-amplitude cyclic loading. It is the most popular

algorithm used in the analysis of fatigue data in order to reduce a load spectrum of varying stress into a set of simple series of stress reversals. Its wide diffusion is due to the fact that it allows the application of the Miners rule so that it is possible to assess the fatigue life of a structure subject to complex loading. The algorithm was developed by Tatsuo Edo and M. Matsuishi in 1968.
The rainflow method gets its name from a metaphorical flow of rain-drops down many overlapping “pagoda” roofs, where the peaks and the valleys of a random load history are represented by the edge of each roof. The complete description of the method is left to the reader since it is just part of the technical literature.

3.3 Fatigue life analysis

3.3.1 Damage

Among the many mathematical models to describe progressive cumulated damage, a linear damage rule has been widely accepted by engineers because of its simplicity, despite its shortcomings of unpredictability and the exclusion of the load-sequence effect. The linear damage rule assumes damage (life-used up) is additive and defines failure to occur when Equation 3.5 is verified.

$$D = \sum d_i = \sum \frac{n_i}{N_i} \geq 1.0 \tag{3.5}$$

The summation notion represents the accumulation of each individual damage (d_i), which is defined by the ratio of n_i to N_i, where n_i is the counted number of revolutions at a driving shaft torque level T_i from PG testing and N_i is the fatigue life in revolutions to failure at the same torque level determined from a torque versus life (*T-N*) curve. For a gear train system with a rotating driving shaft subject ot a variable torque history with various speed in revolutions per minute (rpm), the range-pair counting algorithm is used to identify the number of revolutions (n_i) at each torque level with reference to the RMH. In order to compute the life portion consumed by each torque class T_i and, consequently, assess the total damage accumulated during a test, the Wöhler theory is used and Equation 3.6 is written.

$$N_f \sigma_f^k = B \tag{3.6}$$

Where B and k are constant values related to the material and N_f is the number of cycles completed to failure at the corresponding stress level σ_f. Then Equations 3.5 and 3.6 can be putted together in order to compute the total damage accumulated during the test, obtaining Equation 3.7

$$D = \sum \frac{n_i}{N_f} \left(\frac{\sigma_i}{\sigma_f} \right)^k \tag{3.7}$$

In FIgure 3.3 is reported the Wöhler curve used for the damage estimation analysis. It is worth to notice that the curve has two slopes depending on the fatigue life region considered. More in particular, for life included between 10^3 and 10^7 cycles it is assumed a slope k_1, instead for higher number of cycles a slope k_2 is considered. Values lower than 10^3 are not investigated since are not of interest for this type of analysis.

In Table 3.2 are reported the different slope values depending on the examined component, e.g. spur rather than bevel gears, and its characteristic N_{eq}.

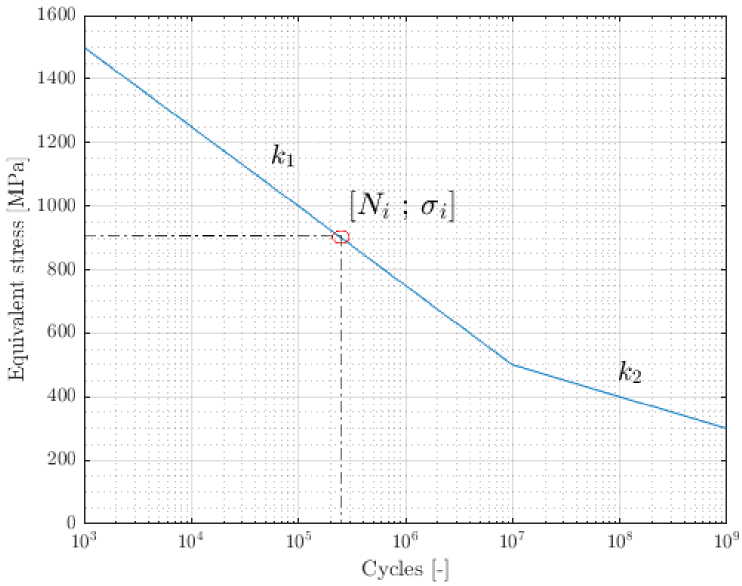

Figure 3.3: Example of Wöhler curve (S-N curve)

		Wöhler curve slopes		
		k_1	k_2	N_{eq}
Spur gears	Bending	8.9	16.8	10^7
	Pitting	4.875	9.275	2×10^8
Bevel gears	Bending	6.7	21	6×10^6
	Pitting	5.5	17.3	10^9
Bearings	Ball	3	3	-
	Roller	3.3	3.3	-

Table 3.2: Wöhler curve slopes

3.3.2 Equivalent torque

The damage computed using Equation 3.7 can be matched with the one produced by a constant amplitude stress (Equation 3.5) in order to compute the value of the latter and obtaining a unique parameter which represent the entire test conditions. From this equality the equivalent stress can be computed by writing Equation 3.8.

$$\sigma_{eq} = \sqrt[k]{\frac{\sum n_i \sigma_i^k}{N_{eq}}} \tag{3.8}$$

It is worth to notice that the stress σ_i is due to torque only and so it can be demonstrated that Equation 3.9 holds. Consequently, the equivalent torque T_{eq} can be computed as the torque level able to reproduce the same damage level provided by the RMH (Equation 3.10).

$$\frac{\sigma_i}{\sigma_{eq}} = \frac{T_i}{T_{eq}} \tag{3.9}$$

$$T_{eq} = \sqrt[k]{\frac{\sum n_i T_i^k}{N_{eq}}} \tag{3.10}$$

As reported in figure 3.3, the TDOA methodology prescribes to use a S-N curve featuring two different slope. Therefore, since the Equation 3.10 refers to a single slope curve, it has to be corrected. In particular, it has to be rewrite for each slope and then, the equivalent torque is obtained by an iterative calculation between the two equations reported in 3.11.

$$\begin{aligned} n_i \times T_i^{k_1} &= N_{eq} \times T_{eq}^{k_1} \\ n_i \times T_i^{k_2} &= N_{eq} \times T_{eq}^{k_2} \end{aligned} \tag{3.11}$$

3.4 Wear and seizure

In order to evaluate the damage of the components that are subject to wear, it is considered the wear as proportional to the work that acts on the component. Therefore, the quantity that estimate the level of wear is the total energy evaluated as follow for those components experiencing a relative displacement or rotation with respect another:

$$E_{differential} = \int_0^{\theta_{final}} T_{diff}(\Delta\theta) \times \Delta\theta d\theta \tag{3.12}$$

$$E_{clutch} = \int_{0}^{\theta_{final}} T_{primary}(\Delta\theta) \times \Delta\theta d\theta \tag{3.13}$$

Where $\Delta\theta$ is the relative angle between left and right half-shafts in the case of differential analysis, while it represents the relative angle between flywheel and primary shaft in case of clutch examination. Obviously, the higher the energy amount computed during the test the higher is supposed to be the requirement of the test in terms of wear.
Power estimation is necessary in order to rate the severity of an event in which a component is subject to a mechanical work in terms of seizure or thermal stress. The power diagram is a tool used to compare the thermal requirement of events in which mechanical components are slipping and to rate the level of energy that is dissipated in relationship with the slipping time. The construction of the power diagram starts with the computation of the instantaneous power at each sample time by using Equation 3.14.

$$P_k = T_k \times \Delta\omega_k \frac{2\pi}{60} \tag{3.14}$$

Where T_k is the torque measured at the k-th sample time and $\Delta\omega_k$ is the relative angular velocity measured at the same sample time k, expressed in rpm. Depending on the test time interval, the variable n is computed in order to respect the Equation 3.15.

$$2^{n-1} < N < 2^n \tag{3.15}$$

Where N is the number of samples registered during the test time interval. Then, the moving average vector (MAV) is constructed as:

$$MAV_{(j)} = [2^0, 2^1, 2^2, ..., 2^{n-1}, N] \tag{3.16}$$

Subsequently, the moving average is applied multiple times to the entire time history, depending on the length of the MAV. In particular, the numerical expression is reported in Equation 3.17.

$$P_{m_{k_j}} = \frac{1}{MAV_j} \times P_k + \frac{1}{MAV_j} \times P_{(k+1)} + ... + \frac{1}{MAV_j} \times P_{(k+MAV_j-1)} \tag{3.17}$$

Figure 3.4 represents a delta speed and differential torque time history used in order to compute the differential power diagram.

In Figure 3.5 is showed graphically what explained up to now by means of mathematical expressions. It is important to notice that only the maximum value among all the different moving averages is considered for each sample time. The power

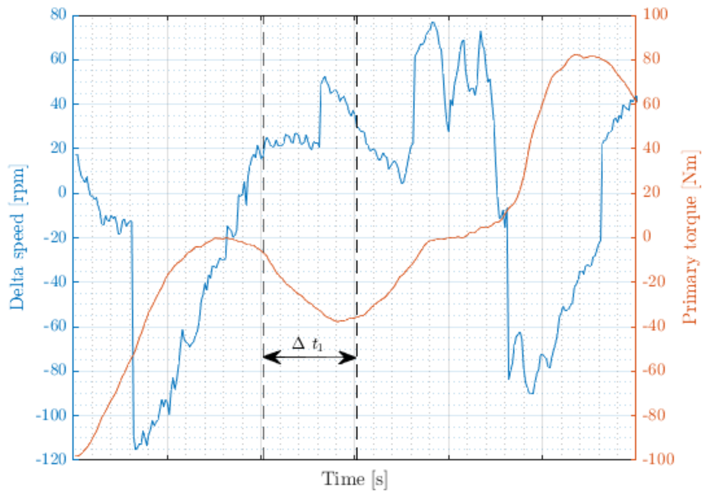

Figure 3.4: Example of a delta speed and primary torque time history

diagram shows the maximum value of average power for each time value starting from the sample time up to the entire test time: in particular, the first point, corresponding to the sample time, represents the maximum instantaneous power measured in one, i.e. 2^0, sample time during the test, while the last point of the curve corresponds to the mathematical average power measured in the entire test, i.e in 2^n sample times where n is computed according to Equation 3.16. Every point included between these two extremes represent the maximum instantaneous power registered for 2^n sample times. In Figure 3.6 is reported a front differential power diagram with the two characteristic points mentioned above.

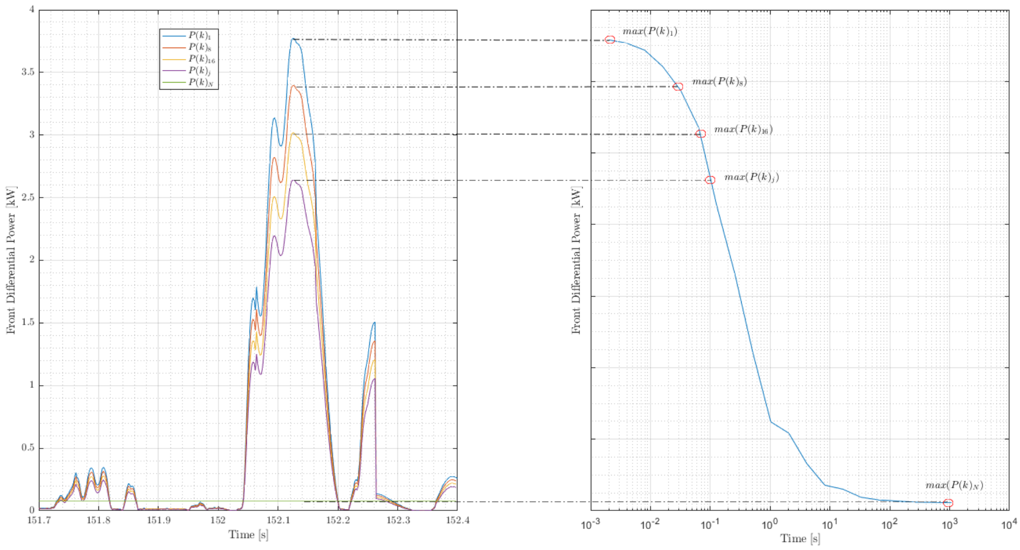

Figure 3.5: Construction of a front differential power diagram

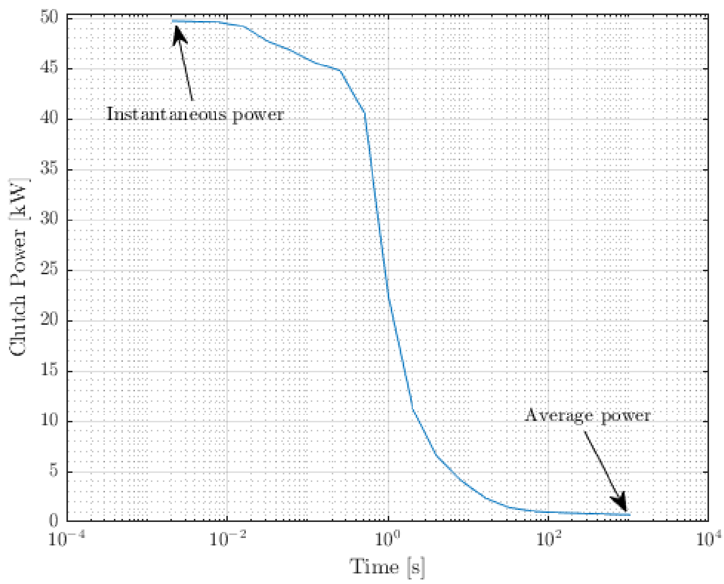

Figure 3.6: Front differential power diagram

Chapter 4

Cycle counting method numerical implementation

The rotating moment histogram and the other metrics prescribed by the TDOA methodology presented in Chapter 3 are addressed no more from a theoretical perspective, but from their numerical implementation by means of dedicated algorithms.

4.1 I/O definition

Before proceeding with the algorithm implementation of the various metrics, it is worth to summarize all the inputs/outputs expected to be acquired/obtained with the execution of such algorithms. Starting from the expected outputs, the inputs can be retrieved in a second stage of the analysis. The outputs are:

- Rotating moment histogram at the differential gear;
- Rotating moment histogram at the different gears;
- Temperature time-based rotating moment histogram.

Consequently, the inputs, i.e. the analog signals to be acquired, can be identified. They are:

- Differential gear speed, expressed in rpm;
- Differential gear torque, expressed in Nm;
- Primary shaft speed, expressed in rpm;

- Gearbox oil temperature, expressed in °C.

A very important consideration to be underlined concerns the differential torque. In fact, it cannot be directly measured since the differential gear is not provided with any strain gage able to measure the torque. Referring to Section 2.1 and to the gearbox scheme reported in Figure 2.3, the differential torque can be computed as

$$T_{diff} = T_{upper} + T_{lower} \tag{4.1}$$

where T_{upper} and T_{lower} are respectively the torque measured in the upper and lower secondary shafts.

4.2 Algorithm numerical implementation

Once all the inputs are defined, it is possible to proceed with the development of the algorithms, which will be deployed to the hardware in the last stage of the study.
The RMH algorithm has to count the revolutions of a mechanical component at a fixed torque level. In order to perform this task, the algorithm's input are:

- rotational speed ω_k, expressed in rpm;
- torque level T_k, expressed in Nm;
- differential torque bin array;
- sample time t_k, expressed in second.

The rotational speed and sample time inputs are used to compute the portion of a cycle (n_k) travelled during the sample time interval (t_k) as expressed in the Equation 4.2.

$$n_k = \frac{\omega_k \times t_k}{60} \tag{4.2}$$

Once the n_k has been computed, the differential torque bin array has to be defined. It is a vector identifying different classes for the differential torque between two extremes, i.e. a minimum and a maximum value. These two values are defined depending on the working condition of the component, e.g. in case of the differential gear they are assumed to be equal to -10000 and +10000 Nm, instead in the analysis of the primary shaft they are assumed to be one tenth of the differential case. Then this working range has to be divided in intervals with a constant amplitude. It has been chosen to divide the torque working range into 125 intervals independently from the component under investigation. This reflects in classes

having an amplitude of 160 Nm in case of the differential and 16 Nm in the primary shaft instance. So, the torque level acquired is compared with these interval and the result of Equation 4.2 is assigned to the designated torque class. As a final result the simulation outputs a vector with 125 rows (one for each torque class), each representing the number of cycles computed in that specific torque class during the test. In Figure 4.1 is reported graphically the algorithm explained so far.
In the Appendix A.1 the code used for the differential RMH elaboration is reported.

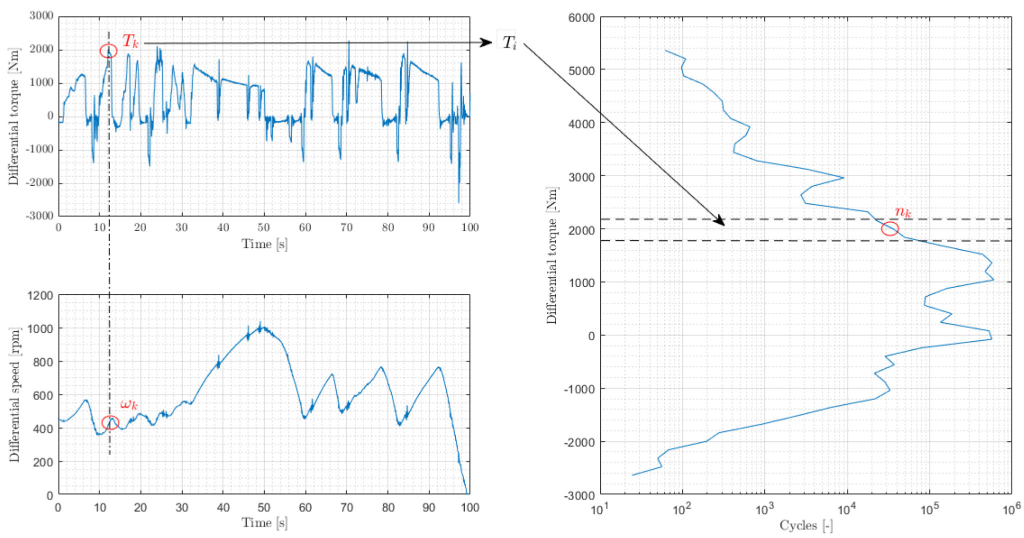

Figure 4.1: RMH algorithm graphical representation

4.3 Sampling frequency trade-off

In the development of the Transmission Durability Embedded System (TDES) one of the most crucial part is the design of the so called "Black Box" (BB): a box which is able to collect, store, elaborate and transmit data coming from different sensors placed in the transmission. The box is based on a low-cost prototyping platform, i.e. an Arduino board. Therefore, the first consideration which can be done concerns the computational power. In fact, it has not the same performance of a modern PC and, consequently, the computational power is surely smaller than the one of the latter. In order to assure the correct operation of the system and to exploit at most the available resources, one of the key parameters to be tuned in such acquisition tool is the sampling frequency. It is important to notice that this parameter affects the measurement accuracy, so its value has to be a good trade-off between resources optimization and measurement precision.

4.3.1 Methodology

Analogue signal discretization in the time domain is known as sampling and is performed by considering only the values of the analogue signal, i.e. the sample, at a given time instants, i.e. at sampling instants. Sampling instants are usually uniformly spaced in time (uniform sampling) and the time interval between two samples is known as the sampling time T_s. The quantity

$$f_s = \frac{1}{T_s} \tag{4.3}$$

which express the number of samples taken in the unit time, is known as the sampling frequency. It is worth to notice that the analogue input signal is continuous both in time and in amplitude, while the sampled signal is discrete in time but still continuous in amplitude. The sampling rate is a fundamental parameter in data acquisition process. High sampling frequencies allow more accurate capture of the peak values of the analog signal under investigation. This leads to a more precise analog-to-digital conversion and this provides not only a better time base resolution but also a good resolution in the frequency domain. The negative effect of high sampling frequencies is that the number of the data points is larger for a given data acquisition time interval. More data occupies more storage space and takes longer for subsequent processing. Furthermore, high sampling frequencies require more performant data acquisition system. An optimal sampling rate has to be chosen based on the maximum frequency of the expected input signal. In order to choose the most suitable one, a correlation between the sampling rate and the error in the measurements and in the elaborated data has to be investigated in more details. The aim of this document is the examination of such correlation in typical automotive transmission components such as gears, final drive and clutch. In order to carry out the comparison of the data at different sampling frequencies, a reference has to be identified: the nominal case is a 10 laps time history of a durability test sampled at 500 Hz. Then the data are filtered so that it is possible to reproduce the data sampled at the required frequency. The frequencies chosen for the study are: 250 Hz, 125 Hz, 100 Hz, 50 Hz, 25 Hz, 10 Hz and 1 Hz. Every output time history is then elaborated using the metrics defined in the TDOA and the comparison with the nominal case is carried out by using appropriate error indicator for each investigated parameter. The meter of judgement among the different sampling frequencies is the error with respect the nominal case of the different parameters that will be evaluated. In particular, the limit error accepted is the 1% in order to be in line to what is prescribed by the TDOA metrics.

4.3.2 RMH and equivalent torque

Once the data are acquired at 500 Hz and then resampled at lower frequencies, the first graphical verification that can be performed in this study is simply the speed and torque time history for both the final drive and the same with the addition of selected gear time history for the gear set. The results are reported from Figure 4.2 up to Figure 4.5.

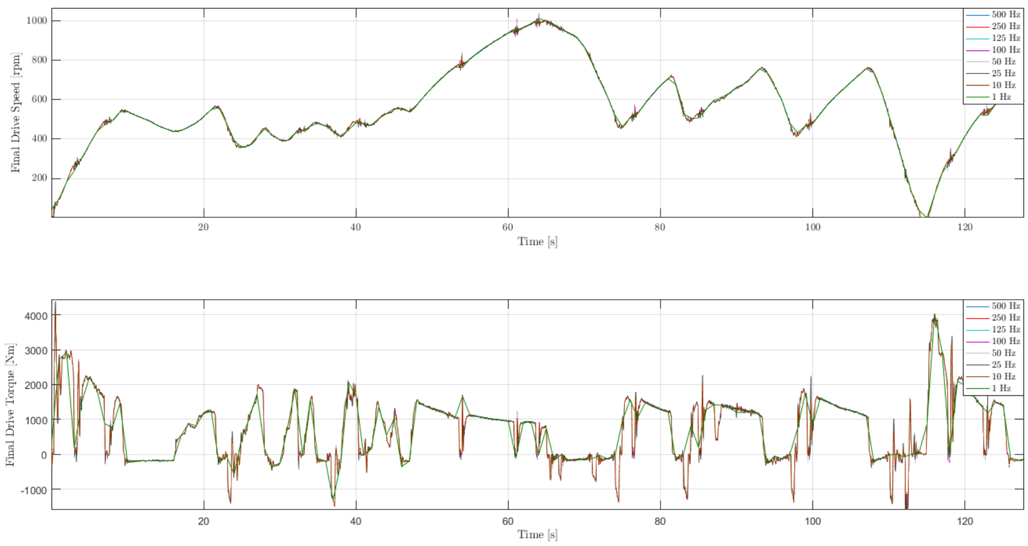

Figure 4.2: Final drive time history

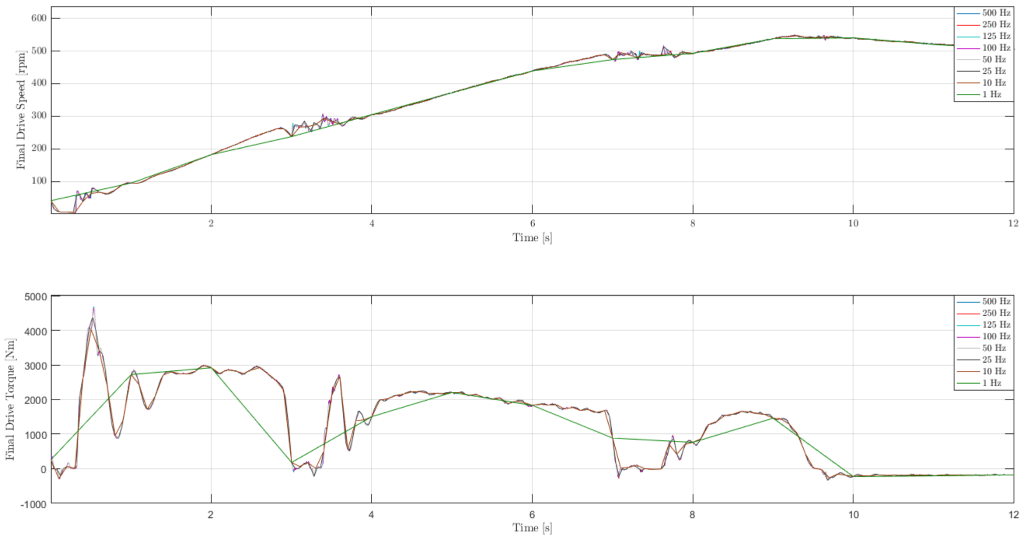

Figure 4.3: Detail of final drive time history (Figure 4.2)

In Figures 4.3 and 4.5 it is evident that lower frequencies are able to catch in a satisfactory way both the rotational speed and the torque time history over the entire durability test. The only frequency unable to evaluate correctly the time history is the lower one, i.e. the 1 Hz.

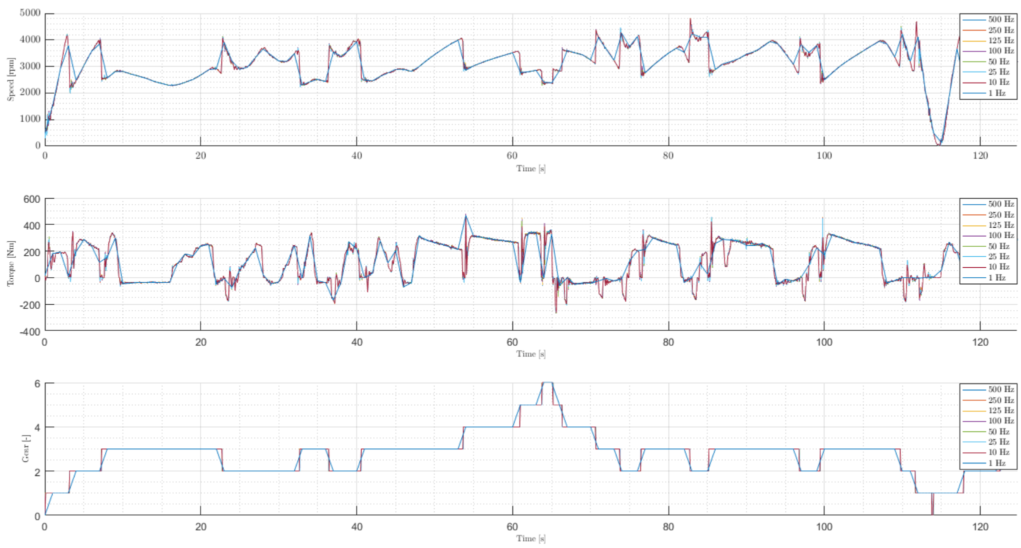

Figure 4.4: Gear set time history

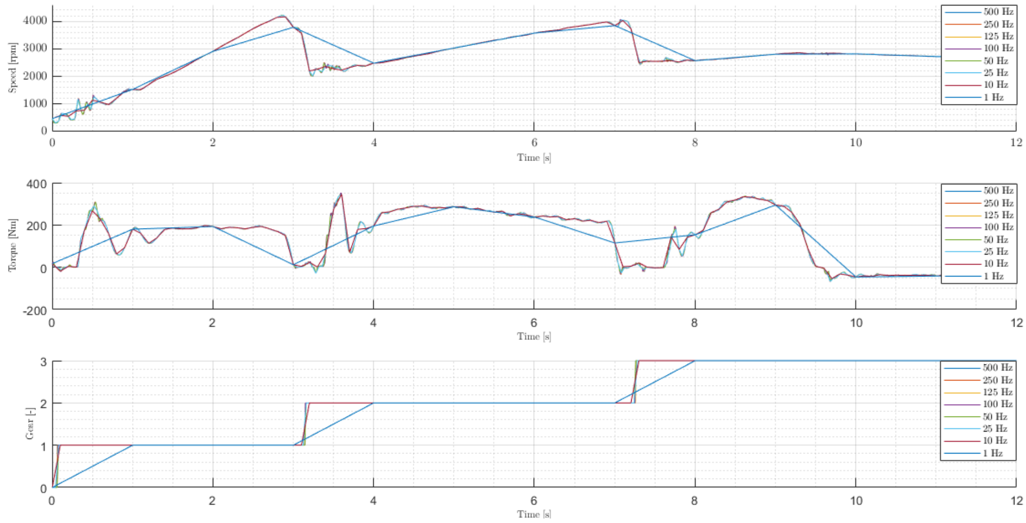

Figure 4.5: Detail of gear set time history (Figure 4.4)

In any case, before assess the correctness of a sampling frequency class, a more representative comparison has to be performed by analysing other different indicators. The starting point for the analysis of typical fatigue life parameters is the Rotating Moment Histogram (RMH) for both the final drive measurements, as reported in Figure 4.6, and for each gear. Looking at Figure 4.6 is clear that, in general, the cycle-counting for each torque class is not so affected despite the great change in the sampling frequency: errors expressed in absolute value greater than 1% are present but they are acceptable since one balance each other and this is confirmed by the fact that the mean error is well below the threshold and quite close to zero. In Figure 4.7 the percentage error (PE) is plotted versus the torque working range in order to have a more clear view of the approximations introduced by the sampling frequency reduction. The grater deviations in terms of RMH are present in the higher torque classes because they correspond to the most dynamic manoeuvres and so lower sampling frequencies are not able to catch these informations: it is worth to notice that even small deviations in high torque classes are responsible of great deviations in terms of equivalent torque, and consequently damage estimation due to the intrinsic nature of the Wöhler theory. The same results are reported in numerical form in Table 4.1. For the sake of brevity the RMH referring to each gear are omitted since the results are in line to what occurs in the case of the final drive analysis.

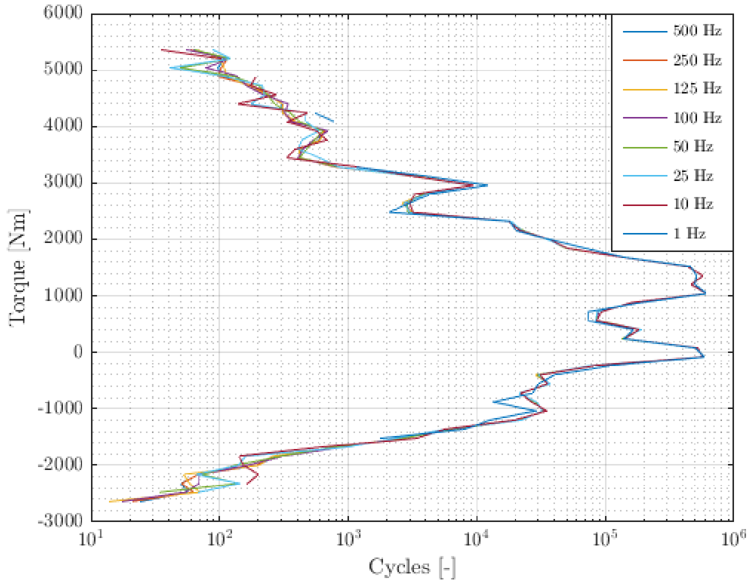

Figure 4.6: Final drive RMH for different f_s

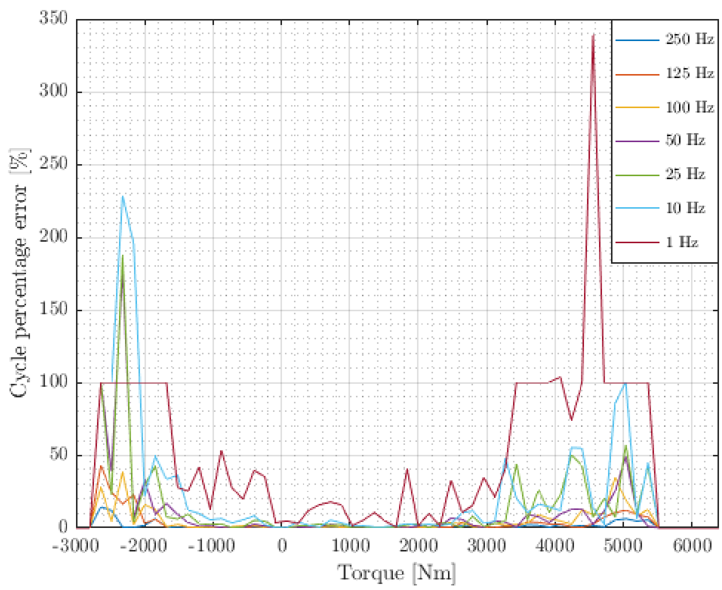

Figure 4.7: Cycle counting error distribution

f_s	250Hz	125Hz	100Hz	50Hz	25Hz	10Hz	1Hz
Cycle counting MPE [%]	0.25%	0.39%	-0.05%	-0.37%	-1.65%	-1.93%	21.85%
Cycle counting max PE [%]	14.28%	42.79%	-38.37%	-176.75%	-187.28%	-228.43%	-339.31%
Corresponding torque class [Nm]	-2640	-2640	-2320	-2320	-2320	-2320	4560

Table 4.1: Cycle counting errors

Once the RMH have been computed, a further significant fatigue life parameter to be computed is the equivalent torque, defined as the constant torque value which is able to generate the same level of damage as the one generated by the RMH. This kind of indicator can give a better feeling of the deviations introduced by lowering the sampling frequency value and its values are reported in the Table 4.2 for both final drive and gear set for each sampling frequency considered. The different values of the equivalent torque can be explained referring to the considerations done above while discussing about cycle-counting error distribution: lower sampling frequencies show a strong deviation in counted cycle estimation for high levels of torque due to the dynamic behaviour of the test and a quite good evaluation for lower ones. The equivalent torque is computed referring to the Wöhler law and so, since it is an exponential law, even small errors are amplified. The fact that the value of the equivalent torque T_{eq} are not so relevant is due to fact illustrated before when discussing about RMH: the deviations from the nominal case are both overestimations and underestimations and furthermore they balance each other, compensating the total error.

	Equivalent torque T_{eq} [Nm]							
	500 Hz	250 Hz	125 Hz	100 Hz	50 Hz	25 Hz	10 Hz	1 Hz
1st gear	161.2	161.1	161.2	161.2	161.0	161.2	159.2	159.2
2nd gear	215.4	215.4	215.2	215.4	215.8	214.8	216.8	218.2
3rd gear	263.9	264.0	263.9	264.1	264.1	264.8	263.7	259.3
4th gear	238.4	238.5	238.3	238.5	238.5	238.7	238.6	253.3
5th gear	239.3	239.3	239.2	239.2	239.4	239.2	240.6	234.1
6th gear	225.4	225.4	225.4	225.4	225.3	224.9	225.3	228.5
Final drive	1796.1	1796.5	1796.9	1795.9	1795.3	1799.1	1778.9	1742.7

Table 4.2: Equivalent torque

4.3.3 Cumulative damage

Once the equivalent torque has been computed, the computation of typical fatigue life parameters proceeds with the estimation of the damage as a function of the sampling frequency. The methodology used in this case is the Miner's rule since it is the simplest and the most widely used damage model. The damage estimation, as done before for the RMH and equivalent torque, is performed for both the final drive and the gear set. The damage is only the natural evolution of the data collected up to now, so the results are expected to be in line to what found at the beginning for the RMH and then for the equivalent torque, i.e. an overestimation or an underestimation of the latter will reflect respectively in a higher or lower damage estimation. The results for each sampling frequency are reported in Table 4.3, while in Figure 4.8 are reported the results for the final drive only both in terms of absolute value and in terms of percentage error referred to the nominal case (500 Hz). Also in this case the deviations introduced by lowering the sampling frequency are visible only at very low values of f_s, i.e. below 10 Hz. For higher values of the latter, the errors are not significant and well below the 1% error threshold.

	Damage [-]							
	500 Hz	250 Hz	125 Hz	100 Hz	50 Hz	25 Hz	10 Hz	1 Hz
1st gear	1.000	0.997	1.000	1.004	0.992	0.997	0.894	0.675
2nd gear	1.000	0.999	0.994	0.999	0.976	0.999	1.063	1.124
3rd gear	1.000	1.001	0.999	1.005	1.003	1.001	0.992	0.836
4th gear	1.000	1.000	0.996	1.001	1.000	1.000	1.007	1.717
5th gear	1.00	0.999	0.997	0.997	1.002	0.999	1.050	0.822
6th gear	1.000	0.998	0.999	1.000	0.992	0.998	0.995	1.126
Final drive	1.000	1.001	1.004	0.999	0.996	1.014	0.917	0.762

Table 4.3: Damage estimation

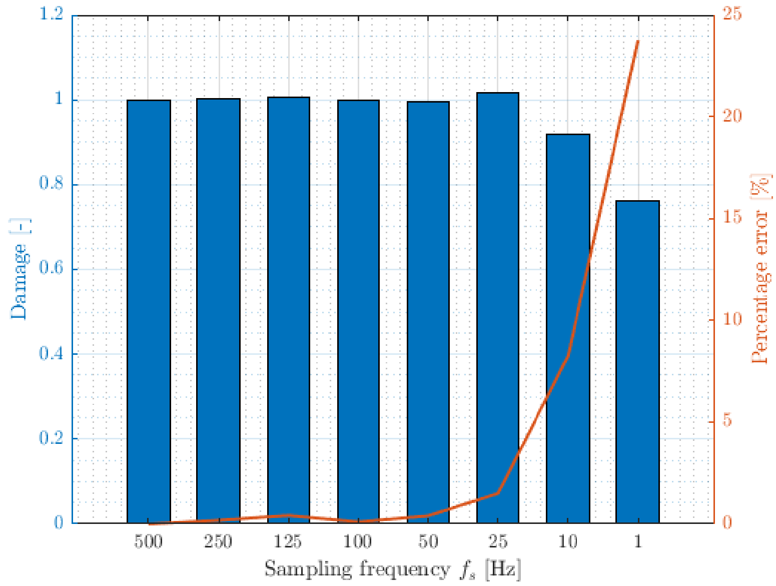

Figure 4.8: Differential cumulative damage

Considering the results just reported for both RMH, equivalent torque and damage estimation, a suitable sampling frequency f_s among the proposed one is the 100 Hz. In fact, despite it introduces, obviously, some simplifications, it is able to reproduce faithfully all the considered parameters examined so far and to guarantee an error lower to 1% with respect to the state of the art.

4.3.4 Cumulative energy and power diagram

After discussing typical fatigue life parameters and assessing the influence of the sampling frequency on such indicators, the analysis shifts to energy and power methodologies used in order to evaluate the wear and the thermal stress to which automotive transmission components are subject. The involved components in this study are the front and rear (only in case of 4WD) differential and the clutch. For each of them the cumulative energy and the power diagram will be computed as stated by the TDOA norm and explained in Section 3.4. The power analysis focuses mainly on the values of the maximum and mean power: the former is the instantaneous power at the sample time and it is represented by the first point of the curve; the latter is the test average power and it is represented by the last point of the same curve. In Figure 4.9 and in Figure 4.10 are reported the power diagrams for both clutch and front differential and in Table 4.5 are reported the numerical values for both the maximum and mean power for all the three studied components.

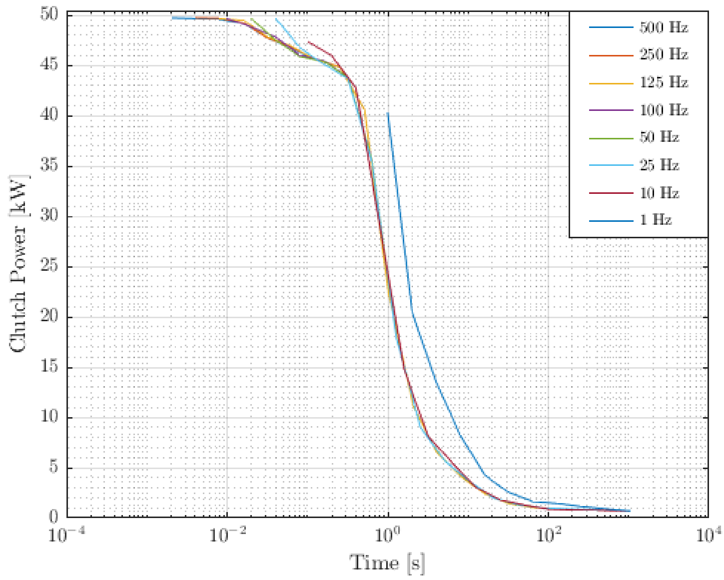

Figure 4.9: Clutch power diagram

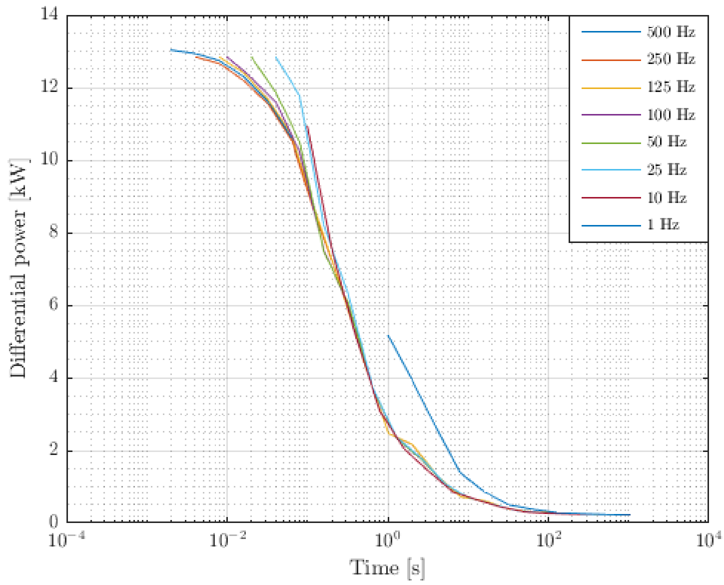

Figure 4.10: Front differential power diagram

In Table 4.4 are show the cumulative energy results for each investigated component. The cumulative energy analysis shows no significant deviations in any components from the nominal case for sampling frequencies up to 50 Hz. Particular attention has to be paid in the analysis of the front differential maximum power since the estimated value for sampling frequencies lower than the nominal one is quite different from the reference one, in fact the relative error is greater than 1% also for frequency value of 250 Hz.

	Cumulative energy [MJ]							
	500 Hz	250 Hz	125 Hz	100 Hz	50 Hz	25 Hz	10 Hz	1 Hz
Clutch	157.89	157.73	157.64	158.15	158.69	159.95	155.1	150.09
Front differential	96.27	96.27	96.29	96.27	96.25	96.37	95.78	92.78
Rear differential	66.78	66.78	66.76	66.81	66.81	66.72	66.70	63.00

Table 4.4: Cumulative energy

This behaviour can be explained referring to the meaning of P_{max}, i.e. it is the maximum power value registered during the entire test. So, it is possible that it appears only once in the test time interval and, consequently, it is not sure that lower sampling frequencies are able to catch this precise point. Further analysis are carried out on this particular case and it is demonstrated that this is the case and so the peak value registered at 500 Hz corresponds to a single value and so it lasts only for 0.002 s. Once considered this particularity, it is possible to state that also in the case of the front differential maximum power the influence of the sampling frequency is not so relevant.
Looking at the cumulative energies, the considerations don't change since the errors introduced by lowering the sampling frequency are below the 1% threshold for different class of the latter. So, as happened in case of the "mechanical" parameters analysis, it is possible to individuate the 100 Hz class as a suitable sampling frequency for the power and energy estimations.

4.3.5 Proposed sampling frequency

Once examined the results belonging to all the different indicators taken into consideration it is clear that values of the sampling frequency lower than 50 Hz are unacceptable for this kind of investigations. A good trade-off between measurement accuracy and hardware resources optimization among all the different sampling frequencies examined is represented by the 100 Hz class. In Figure 4.11 are reported the values of the equivalent torque and damage for this proposed frequency f_s.

	Clutch		Front differential		Rear differential	
f_s	$P_{max}[W]$	$P_{mean}[W]$	$P_{max}[W]$	$P_{mean}[W]$	$P_{max}[W]$	$P_{mean}[W]$
500 Hz	49774.3	704.51	13030.57	223.88	21743.43	181.48
250 Hz	49774.3	704.15	12846.38	223.87	21554.87	181.48
125 Hz	49699.73	703.26	12846.38	223.92	20824.78	181.42
100 Hz	49699.73	707.00	12846.38	223.87	21554.87	181.58
50 Hz	49699.73	708.35	12846.38	223.83	21554.87	181.64
25 Hz	49699.73	711.38	12846.38	224.11	17214.62	181.48
10 Hz	47408.34	687.13	10917.69	222.71	15625.65	181.22
1 Hz	40280.44	687.65	5184.63	215.57	6418.51	171.21

Table 4.5: Maximum and mean power

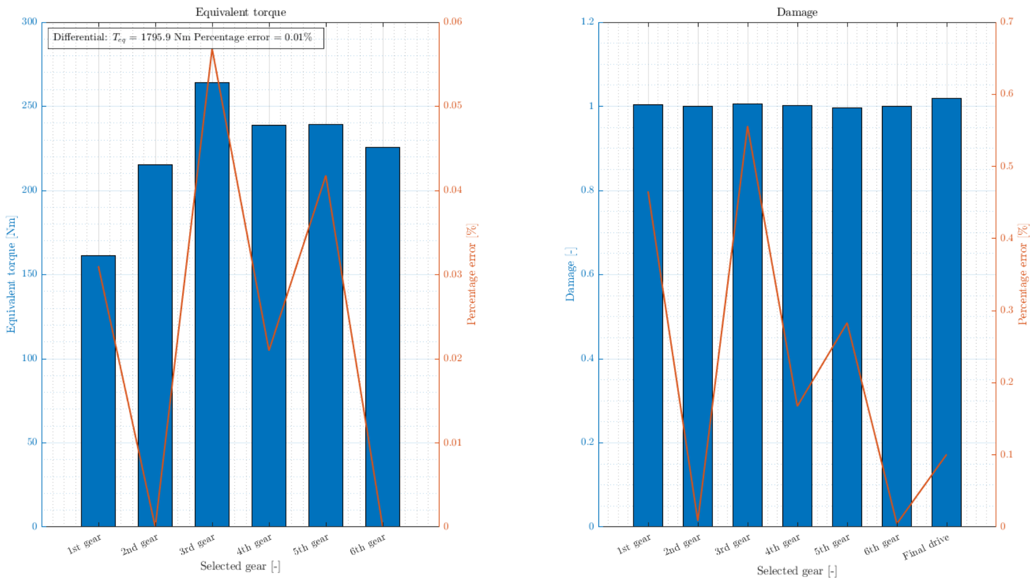

Figure 4.11: Equivalent torque and cumulative damage estimation for the proposed sampling frequency f_s

Chapter 5

Microcontroller programming

Arduino Uno is a low-cost microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs with an operating voltage of 5V and a 10 bit resolution. The Arduino Uno board can be powered via the USB connection or with an external power supply. The microcontroller features a 32 KB of flash memory of which 0.5 KB used by the boot-loader. It also has a 2 KB of SRAM and a 1 KB of EEPROM. The clock speed is 16 MHz. In Figure 5.1 is reported a scheme of the board described above.

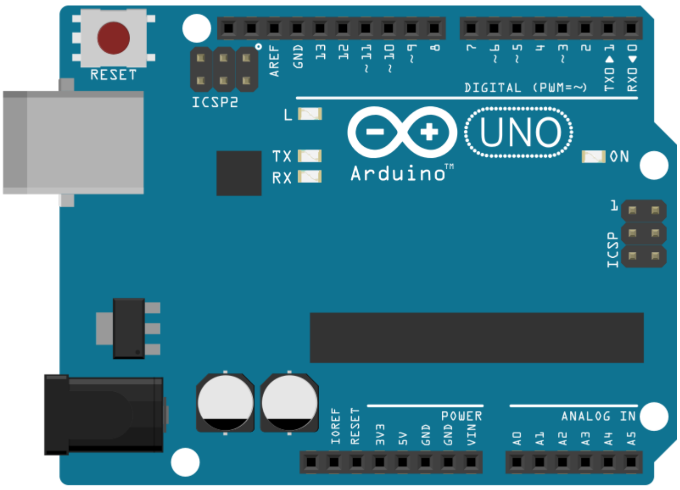

Figure 5.1: Arduino Uno board scheme

5.1 Arduino programming

The programming of the Arduino board is performed by using Simulink and in particular the Simulink Support Package for Arduino Hardware, since it allows to develop the Arduino code by using the Simulink block diagram environment. In particular, the package includes:

- UDP and TCP/IP blocks in the Ethernet shield library and Wi-Fi library to let Arduino hardware communicate with other devices over Ethernet or Wi-Fi;
- Simulink blocks for configuring and accessing Arduino sensors and actuators;
- Access to Arduino WiFi Shield, on-board Wi-Fi chip on Arduino MKR1000, ESP8266, and Ethernet Shield;
- External mode for interactive parameter tuning and signal monitoring as the algorithm runs on the board;
- Model deployment for stand-alone operation on the Arduino.

The programming activity through the Simulink environment allows to graphically represent, in a simple manner, the equivalent to hundreds or even thousands of lines written in Arduino language due to the already existing Simulink libraries and the blocks diagram interface. In Figure 5.2 is reported the configuration used to program the Arduino Uno board and to deploy to the hardware the Simulink block diagram.

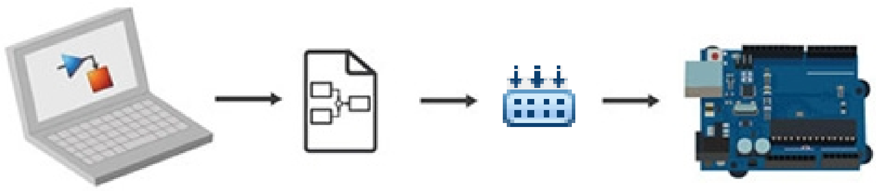

Figure 5.2: Arduino Uno board programming scheme

5.2 Arduino Uno analog signal acquisition

The Arduino Uno board is the core of what has been called the Transmission Black Box (TBB). It is in charge of acquiring and elaborating the analog signals coming from the equipped vehicle. It is important to remind that the Arduino board is a low-cost prototyping platform and therefore it cannot guarantee the same performance, in terms of both computational power and measurement accuracy, of a professional acquisition tool. Despite these limitations, the Arduino Uno board has to satisfy all the requirements imposed by the TDES and therefore, the programming code has to be optimized in order to make the most of the hardware specifications.

5.2.1 Arduino Uno analog pin tare

During the development of the embedded system, one aspect to be faced with is surely the tare of the acquisition tool, in this case an Arduino Uno board. In fact, during the first experiments and acquisitions has been noticed that the value measured by the board was not perfectly aligned with the expected one. In Figure 5.3 is reported a differential torque time history measured in parallel by both the Arduino board and a portable real-time acquisition tool representing the current state of the art of the validation equipment (IMC device). In this representation the deviations just mentioned are clearly visible.
It is important to remind that the Arduino Uno board has 6 analog inputs, labelled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). The operating voltage ranges from 0 up to 5V. This two data are useful in order to computed the resolution of the measurement tool: in fact, diving the working range by the number of possible outputs, it equals to 4.882 mV/bit. Referring to these considerations, the strain gages has to output a signal compatible with the board specifics: the zero torque value is positioned in correspondence of 2.5 V and the extremes (i.e. 0 and 5V) corresponds to the ends of the torque working range (minor than 2.5V for negative values of the torque and greater for positive ones).
For the calibration procedure of the analog pins, the hardware board is connected to a certified signal generator which guarantees a precision of ± 0.0002 V. The entire working range of the Arduino board is covered with steps of 0.1V and the corresponding output together with the expected one, expressed in bit, are stored. In Figure 5.4 it is reported the Arduino Uno characteristic compared with the reference one in terms of volts. Looking at the voltage characteristic can be noticed that the Arduino Uno board tends to overestimate the reference voltage more and more as the reference voltage increases. Furthermore, it can be noticed that the analog pin saturates for values lower than 5V, so reducing further the useful measurement range. The characteristic just described is clearly not suitable for torque measurement purposes, then another solution has to be found. This solution is represented by the linear regression. In particular, two different approaches were

studied:

- a unique linear regression representing the voltage characteristic;
- multiple linear regressions covering the entire working range.

The second options consists in subdividing the voltage range into a finite number of intervals and compute the linear regression for each of them, as they were completely independent. In Figure 5.5 is showed the comparison between the original characteristic and the ones obtained as described above. Looking at the calibrated characteristics, it is evident that the multi linear approach guarantees a more faithful representation of the reference characteristic, but it is worth to notice that this solution implies a higher computational power in order to be implemented. The simple linear regression represents a good trade-off between accuracy and computational power and, consequently, it will be implemented in the Simulink block diagram.
At this point, it has been decided to use the calibration data just obtained to correct, in post-processing, the differential torque time history represented in Figure 5.3. The result is reported in Figure 5.6. It is evident that the measurement of the Arduino board coincides quite perfectly to the one of the IMC acquisition tool. One fundamental point to be considered is that this first verification has been performed in post-processing, so it is not representing the actual accuracy performance of the embedded system.

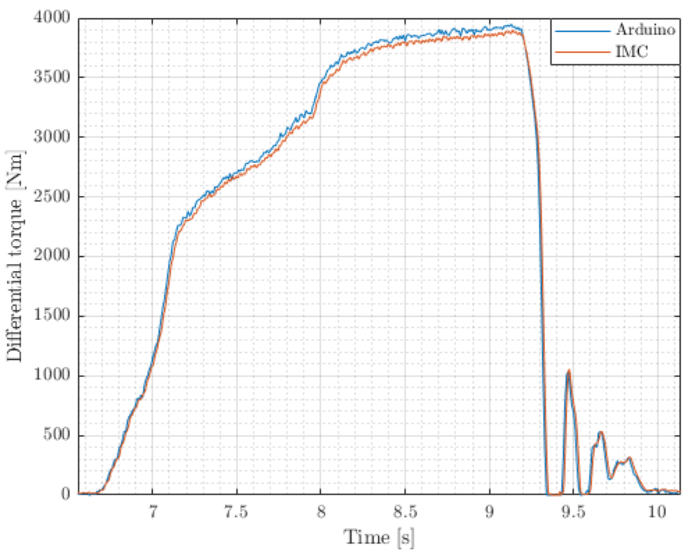

Figure 5.3: Differential torque time history comparison

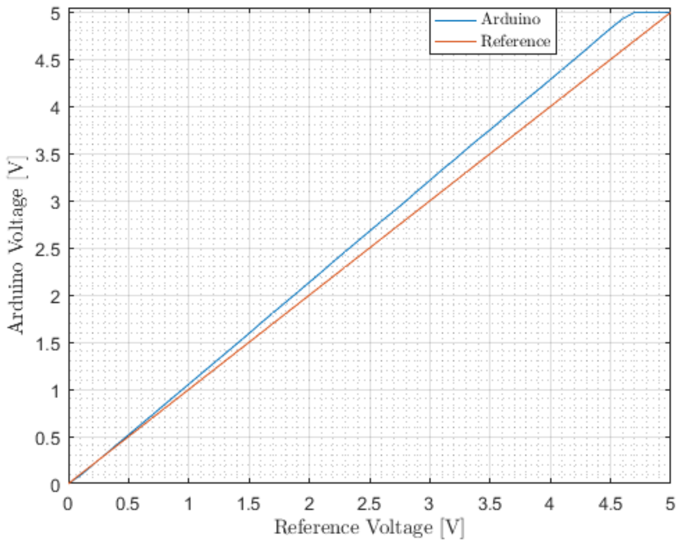

Figure 5.4: Arduino Uno voltage characteristic before calibration

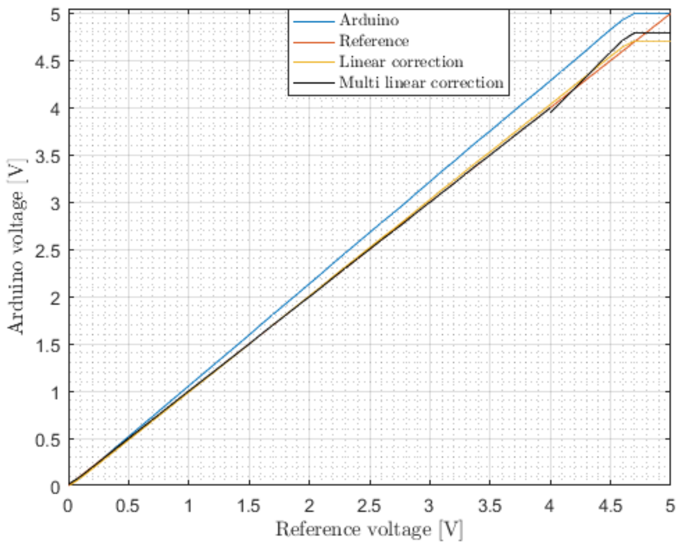

Figure 5.5: Arduino Uno voltage characteristic after calibration

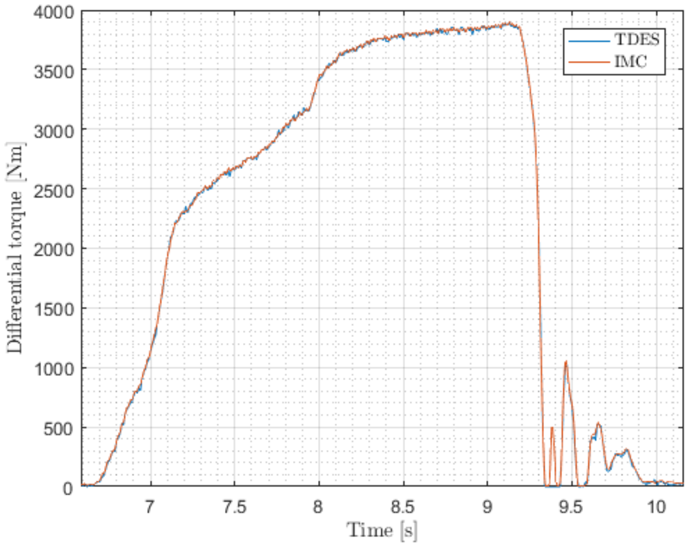

Figure 5.6: Differential torque time history comparison after calibration

5.2.2 Arduino Uno torque acquisition

The fundamental task of the TBB is to acquire the analog signals coming from the TSM, e.g. the one from strain gages. Since the TBB is constituted practically by only the Arduino board, the latter has to accomplish the duty. The Simulink block diagram adopted in order to fulfil this task is reported in Figure 5.7. It consists of a pre-existing “analog input” block in charge of measuring the voltage of a specified analog input pin, a gain and a subsystem identified as 1-D look-up table so that it is possible to correct the measurement errors, as explained in Subsection 5.2.1. It is worth to notice that the analog input block outputs a number ranging from 0 up to 1023 (10 bit) and so the gain is used in order to convert it into a voltage value ranging from 0 up to 5 volts. In Figure 5.8 are reported the parameters used to set-up the analog input block. In addition to these components, two additional blocks are included in the Simulink sketch:

- Scope, used to monitor in real-time the voltage;
- To Workspace, used to log the voltage time history once the simulation ends.

It is important to underline that the block diagram just reported represents a fundamental block because it can be copied and pasted every time an analog signals, whatever information is carried by the same, has to be acquired by means of the Arduino analog pin. The only modification will be the number of the analog pin to be read, depending on the configuration adopted.

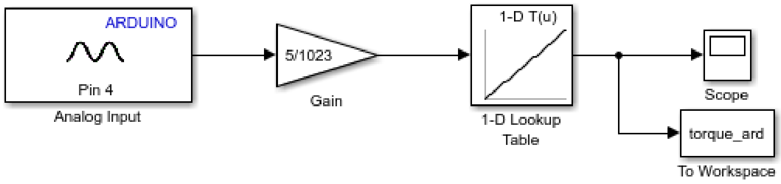

Figure 5.7: Analog signal acquisition Simulink block diagram

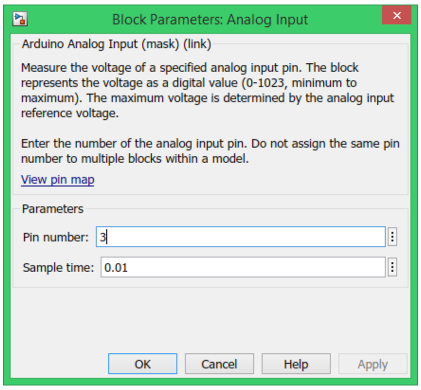

Figure 5.8: Analog input block set-up

5.2.3 Arduino Uno RPM measurement

The working principle of a Hall effect sensor was explained in Chapter 1.3. In this section, focus will be paid to the conversion of such output signal into a more meaningful form, i.e. the RPM value, and its acquisition. The output signal of a inductive field sensor is a constant amplitude and variable frequency square wave, in other words the information carried by the signal is stored in its frequency rather than in its amplitude content as happened in the case of strain gages. More in detail, the maximum frequency of the signal coming from an Hall effect sensor, in case of an automotive application, can be computed considering the engine maximum rotational speed and the number of teeth in the flywheel (usually a little bit more than 100). Using this first approximations the maximum frequency computed is about 10 kHz. Afterwards, the sampling frequency of the sensor has to be at least 2 times the one prescribed by the Nyquist theorem. Obviously, in order to have an accurate measurement, the sampling frequency used in real application is usually greater with respect to the one computed referring to the Nyquist theorem and so, the required sampling rate hovers around 40 kHz.

Referring from this considerations, the first question to be answered is: "Is an Arduino board able to sample analog signal at such high frequencies?". Referring to the hardware technical specifications the answer is clearly negative. So, an alternative way to acquire the RPM measurements have to be found. The solution consists in the adoption of an additional external components, i.e. a frequency to voltage converter (FCV). This instrument allows the Arduino board to read the signals coming from an Hall effect sensor as it was a strain gage, i.e. by measuring its amplitude and no more its frequency content. So, the same Simulink block diagram reported previously in the torque acquisition can be used in an analogue manner in order to measure the rotational speed.

5.2.4 Arduino Uno time history acquisition

Once the calibration of the analog pins is carried out and the fundamental block for analog signal measurement has been showed, the obvious continuation is the registration of an entire on-board vehicle time history. In Figure 5.9 is reported the comparison between the time history measured by the IMC acquisition tool, representing the current state of the art, and the one of the Arduino Uno by means of its dedicated analog pins. It is evident that, despite the tare activity presented in Subsection 5.2.1, the results are not acceptable since the differential torque "red" by the TDES is always an overestimation of the actual one.

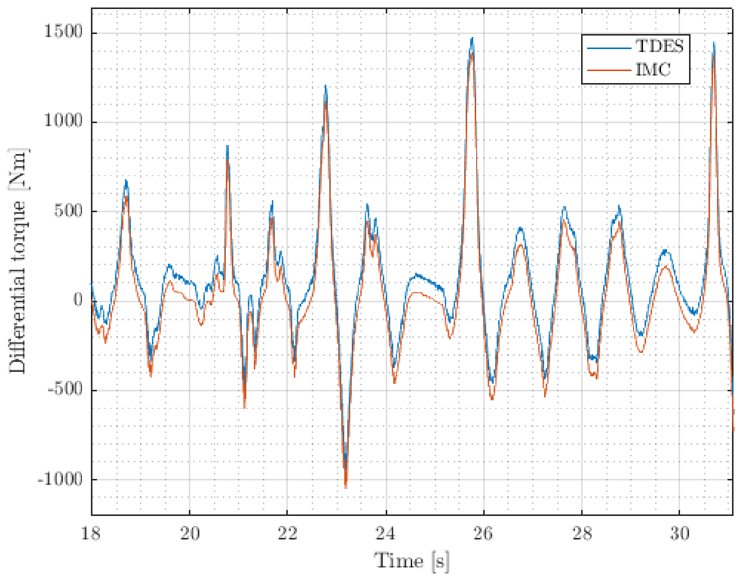

Figure 5.9: Differential torque time history comparison

5.3 Arduino I^{2}C communication

Up to this point were discussed about analog signal acquisition by means of the dedicated analog pin present in the Arduino board. This approach, as demonstrated in the previous sections, has its main limitation in the measurement accuracy and it is, consequently, not suitable for mission profile analysis purposes. The I^{2}C represents an alternative way to fulfil the same task, but avoiding the serious shortcomings of the analog pin approach. In fact, it guarantees a 16 bit resolution over the same voltage working range ($0\div5$V), despite the 10 bit provided by the Arduino itself.
The I^{2}C (Inter-Integrated Circuit) is a chip-to-chip protocol for communicating with low-speed peripherals. It is a multi-master, multi-slave, packet switched, single-ended, serial computer bus. I^{2}C uses only two bidirectional open-drain lines, i.e. Serial Data Line (SDA) and Serial Clock Line (SCL), pulled up with resistors. The I^{2}C reference design has a 7-bit address space.
Arduino devices have one or two I^{2}C buses. Each bus has an I^{2}C Master connected to the two bidirectional lines. These two lines are connected to a pair of pins on the hardware. So it is possible to connect multiple I^{2}C devices, such as ADCs, LCDs, and sensors, to the I^{2}C pins on the Arduino hardware. Each I^{2}C device on an I^{2}C bus must have an unique address. Most devices have a default address assigned by the manufacturer. If the address is not unique, it has to be reconfigured according to the device data sheet provided by the manufacturer. In Figure 5.10 it is showed a typical I^{2}C communication between the different players.

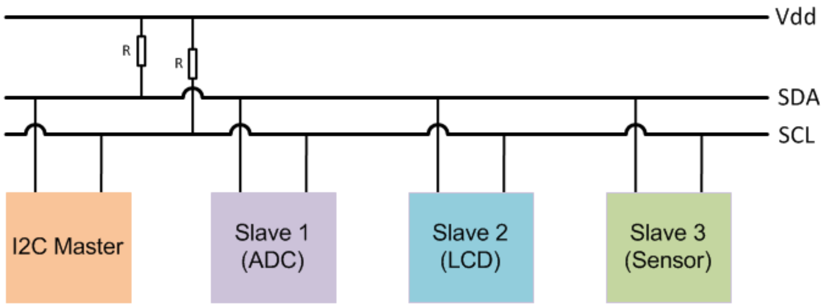

Figure 5.10: General I^{2}C scheme

5.3.1 ADS1115

The ADS1115 is a precision, low-power, 16-bit (N_{bit}), I^{2}C compatible, analog-to-digital converter (ADC). It features a programmable gain amplifier (PGA) and a digital comparator. These feature, along with a wide operating supply range (2÷5.5V), makes the ADS1115 well suited for power and space constrained sensor measurement applications. Furthermore, it provides an I^2C interface consisting in 4-pin-selectable addresses and it allows to perform up to four single-ended or two differential acquisition at the same time. In Figure 5.11 is reported the wiring connection of the ADC to the Arduino hardware: in particular, the ADC power supply is provided by the Arduino hardware itself, the SCL and SDA lines are connected with the dedicated board pin and the devoted address pin of the ADC is grounded in order to set the address to 0x48 (1001000). The remaining pins labelled from A0 to A3 are the pin dedicated to the input analog signals.

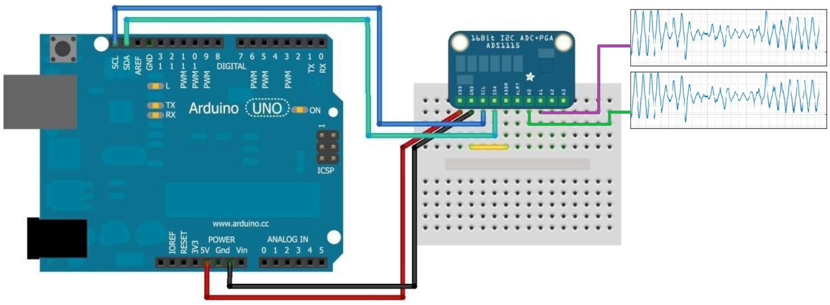

Figure 5.11: ADC wiring connection

The value of the PGA is setted to the default case, i.e. ± 6.144V. Consequently, dividing 6.144 volts by $2^{N_{bit}-1}$, the scale factor of 0.1875 mV per bit can be obtained. It is worth to notice that the standard Arduino ADC had a resolution of about 5 mV, so by using the ADS1115 it is possible to have 26 times the standard resolution of the Arduino Uno board. Unfortunately, the PGA setting is misleading because referring to the reported voltage ranges, it seems that is possible to read voltages up to 6.144V in absolute value. Instead, the maximum measurable voltage is established by the supply voltage to the chip. Specifically, as reported in the data sheet, the maximum measurable voltage is 0.3V more than the supply one and the minimum one is 0.3V smaller than the ground. The ADS1115 supports up to 4 single ended or 2 differential inputs. The question which may arise now is “Which should be used?”. Single ended inputs gives twice as many inputs, but they can, by definition, measure only positive voltages and so, without the sign bit, it has a 15 bit resolution. In addition to providing the full 16 bits of resolution and the ability to measure negative voltages, differential measurements offer more immunity from electromagnetic noise. Despite, the higher measurement quality assured by the differential measurement, the limited hardware specifications of the Arduino Uno board imposes to use the single ended approach. Consequently, in order to acquire correctly the measurements, it proceeded so that the ADS1115 and the different sensors were all grounded to the vehicle.

5.4 Arduino data logger

By design specifications, the TDES must give the opportunity to the user to log the simulation results at the end of the same, without registering the entire time history and avoiding post-processing activities. In order to perform this task, the Arduino Uno board alone is no more enough, but it has to be integrated with an additional component: the Arduino Ethernet Shield.
The Arduino Ethernet Shield is based on the Wiznet W5100 ethernet chip providing a network (IP) stack capable of both TCP and UDP. The Arduino Ethernet Shield supports up to four simultaneous socket connections. It uses the Ethernet library to run sketches which connect to the internet via a standard RJ45 Ethernet jack. It is compatible with the Arduino Uno board and it features a SD slot which enable the writing/reading of files present in the SD card. Arduino communicates with both the W5100 and SD card using the SPI bus (through the ICSP header). This communication takes place on digital pins 11, 12, and 13, while pin 10 and pin 4 are used respectively to select the W5100 and for the SD card. Therefore, these pins cannot be used for general I/O. Note that because the W5100 and SD card share the SPI bus, only one can be active at a time by setting the corresponding pin as an output pin. In Figure 5.12 is reported the assembly of the Arduino Uno board together with the Arduino Ethernet Shield.

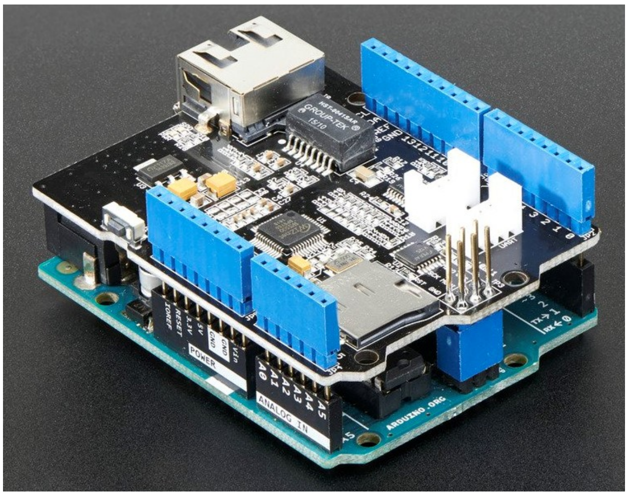

Figure 5.12: Assembly of the Arduino Uno board together with the Arduino Ethernet Shield

The only drawback of this configuration is the lack of the SD library in the Simulink Support Package for Arduino. So, it has to be inserted manually by using a specially designed block in the Simulink environment: the S-Function Builder block. It creates a C MEX S-Function from specifications and C source code provided by the user. In this way it is possible to include whatever Arduino library to the Simulink environment and to use the Arduino programming language in the development of such a block.
The S-Function Builder block integrates, from the hardware point of view, two buttons for two different purposes:

- One button is dedicated to the start of the simulation, i.e. once it is pressed the model starts to run;
- The second button is dedicated to the results saving, i.e. every time it is pressed the Arduino board saves the results to the SD card, each time in a different text file identified with a progressive number.

In order to work properly, the first thing that the S-Function Builder block has to do is to establish a connection between the Arduino hardware and the SD card through the dedicated pin, i.e. pin 4, from the hardware point of view and, from the software standpoint, through the dedicated Arduino libraries included in the block, i.e. the SPI and the SD. This first preliminary action is conceived to run only once and only when the simulation start button has been pressed. So, this part of the programming code will be placed in the discrete update tab of the block. Instead, the updates tab present in the S-Function builder block is dedicated to the part of the programming code in charge of the writing operation onto the SD card. In particular, once the start button has been pressed, it checks if already exist a file in the SD card with the name equal to the default one, if the outcome is negative, it proceeds by generating a file named “RMH00” in which writing the simulation results, otherwise it creates an different file identified with a progressive number in order to not be confused with the already existing one, e.g. if the file “RMH00” and “RMH01” already exist, a new file “RMH02”, containing the simulation results, is created.
All the functionalities made available by the I^2C requires, as in the case of the data logger, an additional S-Function Builder block so that it is possible to be fully exploited. It has to establish a communication between the ADC and the Arduino hardware and to enable the acquisition of the signals connected to the pins of the ADC itself. But before doing that, three parameters have to be setted for the correct operation of the embedded system:

- Slave address;
- Amplifier gain;

- Samples per second (SPS).

All these parameters are defined in the discrete update tab of the S-Function Builder since they have to be defined only once at the start of the simulation. The slave address is left as default option (0x48) and so the dedicated address pin has to be grounded. The amplifier gain is setted to the greater value possible in order to cover the entire working range, i.e. ±6.144V. The SPS, instead, is setted to 860 SPS, which is the first value which enables the ADC to acquire three different signals at a sampling frequency of 100 Hz. Therefore, the outputs of the S-Function Builder block are three analog signals, which share the same ground as all the sensors present in the equipped vehicle. The outputs are provided in terms of voltages and so they have to be translated in physical values, i.e. torques and speed in this particular application. This task is accomplished by the different gains present between the S-Function Builder block and the MATLAB Function in charge of the RMH computation. In Figure 5.13, for the sake of simplicity, the S-Function Builder has been represented in a generic subsystem identified by the ADC picture in order to be immediately recognized.

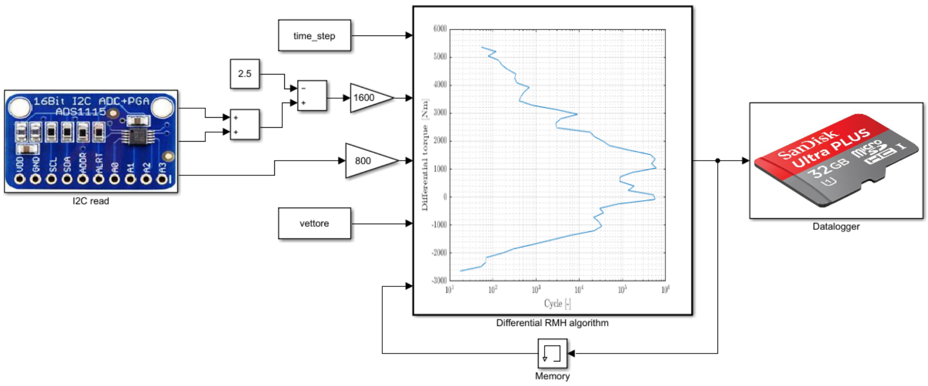

Figure 5.13: Data logger sketch in Simulink environment

It is important to underline that the torque signal has to be shifted in amplitude since it is intended to measure both positive and negative torque values. So, once the voltages indicating the torque “flowing” in the upper and lower secondary shafts have been summed, the shift is performed by subtracting a constant value, i.e. 2.5 volts. By using this shrewdness, the zero torque value corresponds to 2.5 volts, while the minimum and maximum values corresponds respectively to 0 and 5 volts. Once the differential torque and rotational speed have been converted in the proper units of measure, they have to be elaborated by the rotating moment histogram implemented by means of a MATLAB Function block. The outputs of such a block is a 125 rows vector, one for each differential torque class. In turn,

it is the input for the data logger S-Function Builder presented at the beginning of the Section.
In Figure 5.14 is reported the entire TBB work flow, starting from the embedded system programming and measurement up to the results saving in the SD card.

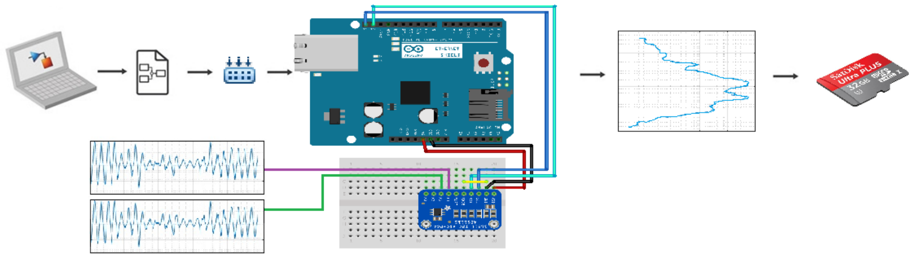

Figure 5.14: Complete microcontroller programming scheme

5.5 Hardware testing

Testing an embedded system directly in the field, i.e. on a real plant (e.g. an automotive transmission) can be impractical and also dangerous, since residual mistakes in the system hardware (HW) and software (SW) could impair the operation of the plant and could damage it. Moreover, it could be impractical to verify in the field the embedded system specifications. For this reason, a different test methodology is employed.
In Figure 5.15 the hardware validation set-up is showed. It consists of a signal generator in charge of providing fictitious inputs to the embedded system under test. The outputs of such device are not directed immediately to the Arduino board, but are first received by the IMC device. This shrewdness allows to log the signals by means of the IMC and to redirect them to the embedded system without any modification, i.e. it is like the Arduino board were directly connected to the signal generator output.
The first verification performed using this set-up consisted in the acquisition of a time history. In Figure 5.16 are plotted in the same chart the time histories registered by using the IMC device and the Arduino board. It can be noticed that, as it was expected, the IMC is able to provide to the user a more “clean” signal, anyway the embedded system succeeds to reproduce faithfully the time history too. In Figure 5.17 are reported the same curves of Figure 5.16, but showed with a smaller time scale in order to better visualize the measurement deviations introduced by the embedded system.

Once the consistency of the measurements have been verified, the hardware testing

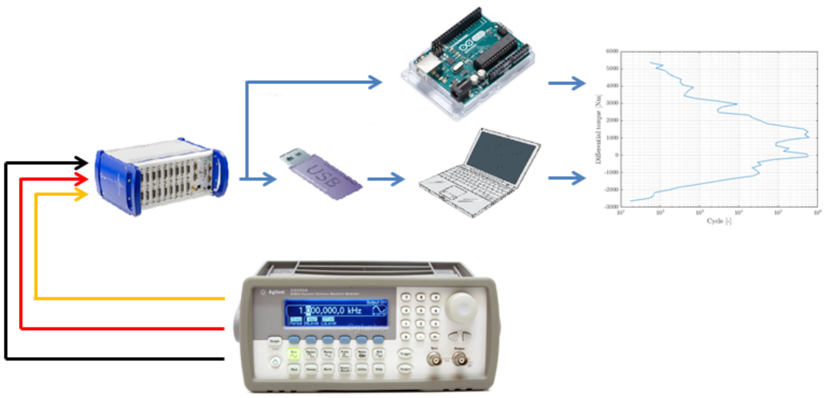

Figure 5.15: Hardware functional verification scheme

proceed by elaborating the rotating moment histogram (RMH) of a time-history acquired in real-time, i.e. the same conditions of a on-board vehicle test are established. In this case, the input analog signals are not expressed as volts, but they are elaborated as they were acquired from the equipped vehicle. So, the gains, computed during the strain gage tare in subsection 2.1.1, are used to multiply the voltage value across the strain gages themself and then to compute the instantaneous torque level. Same reasoning is applied for the measurement of the differential rotational speed.
In Figure 5.18 are reported the RMHs computed respectively by the TDES, in blue, and, in post-processing using the TDOA methodology, by the IMC device, in red. Looking at the results, it is evident that the TDES does not introduce any significant deviation with respect to the current state of the art. In fact, the number of counted cycles in each torque class is essentially equal to the one obtained by the elaboration of the IMC data. If it was not enough, the results' comparison is enriched by the equivalent torque value in order to have a unique engineering value summarizing the entire picture. The equivalent torque value computed by TDES is equal to 431.2 Nm, but the most meaningful data is the relative percentage error with respect to the one elaborated starting from the IMC time history: it is only 0.01%. It is evident that this value is well below the 1% error imposed as acceptable threshold.

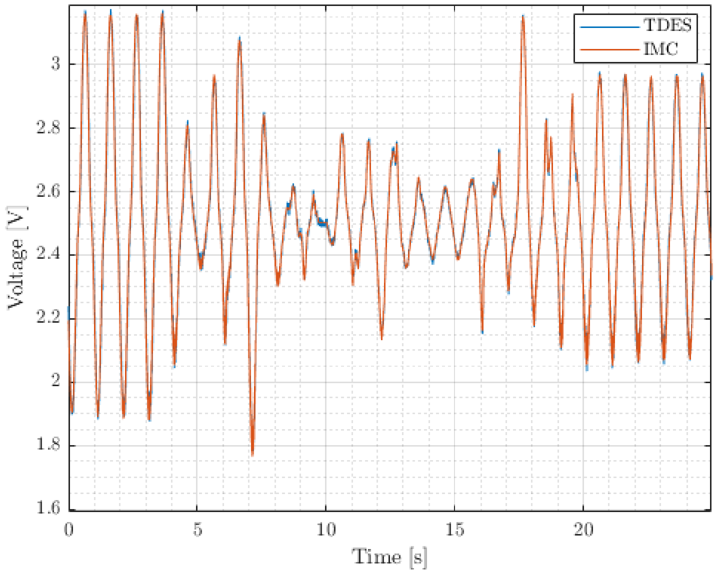

Figure 5.16: Time histories comparison

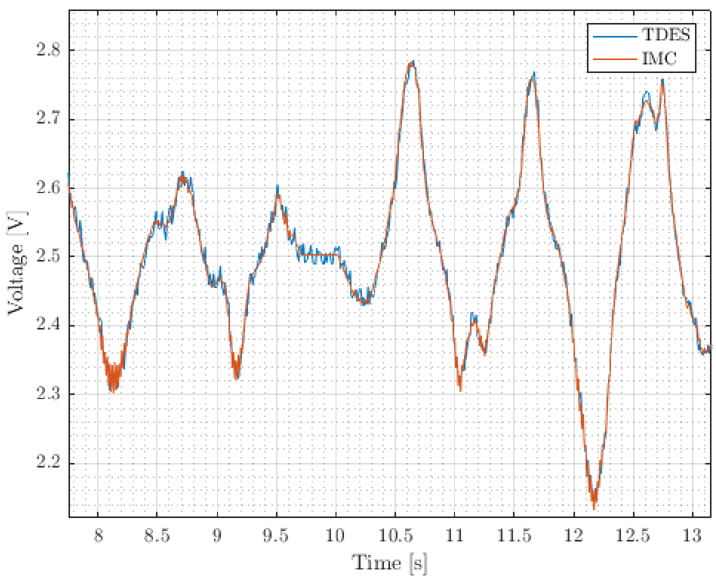

Figure 5.17: Time histories comparison

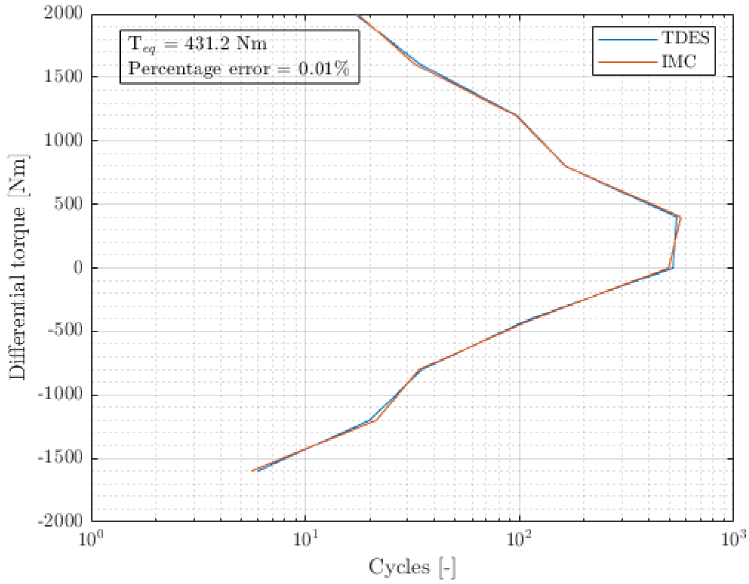

Figure 5.18: Hardware testing RMH

Chapter 6

TDES validation

So far, the development of the TDES proceeded validating one aspect of the same at the time and testing the hardware by using fictitious inputs. In this chapter all the aspects described in the previous chapters are putted together and the final validation with an on-board vehicle test is carried out.

6.1 Layout

The final validation is performed with an on-road vehicle test. The acquired channels are:

- Upper secondary shaft torque;
- Lower secondary shaft torque;
- Differential gear speed.

In Figure 6.1 it is showed the vehicle layout used for the validation test. It is worth to notice that, differently from Figure 5.15, now the embedded system is receiving, always through the use of the IMC device, the original signals coming from the strain gages and the pick-up with which the vehicle has been equipped. A question which may arise is “Why is not the Arduino board connected directly to the vehicle sensors?”. Part of the answer was explained in Section 5.5, but during the on-board vehicle test special needs have to be fulfilled. The complete justification for this configuration can be found considering what discussed in Subsection 2.1.1 and in Subsection 5.2.1. In fact, the output of the strain gages is an analog signal -10 $\pm$ 10V, while the operating voltage of the Arduino Uno board is just

$0 \div 5V$. Referring to these considerations, the strain gage output has to be modulated: in particular it has to be resized and shifted. The IMC acquisition tool is used in order to accomplish this task, i.e. it applies a gain and an offset to the original strain gage analog signal and then it outputs the conditioned measurement. A shrewdness important to underline is that the start and save buttons implemented for the embedded system operation, in the validation test are exploited by the IMC device too so that it is possible to start and to stop the parallel measurement together with the TDES. In this way a synchronous registration between the two systems is guaranteed.
In Figure 6.2 is showed in more details the hardware layout and its connection used for the validation test. In particular, it is worth to underline that the ADC converter is powered directly by the Arduino board through the dedicated 5V and GND pins. The button used for the start and saving operations are powered by the Arduino board too and their outputs are connected again to the hardware digital pins. Instead, the analog signals connections are not showed in the picture.

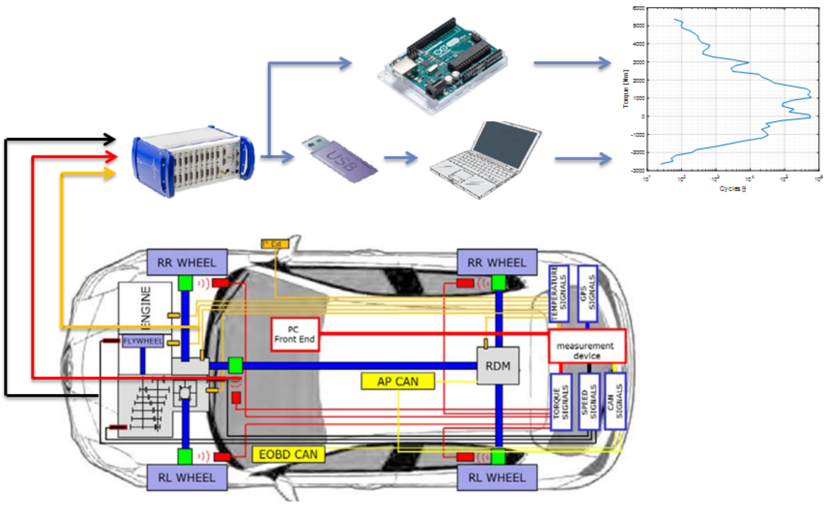

Figure 6.1: Functional validation scheme

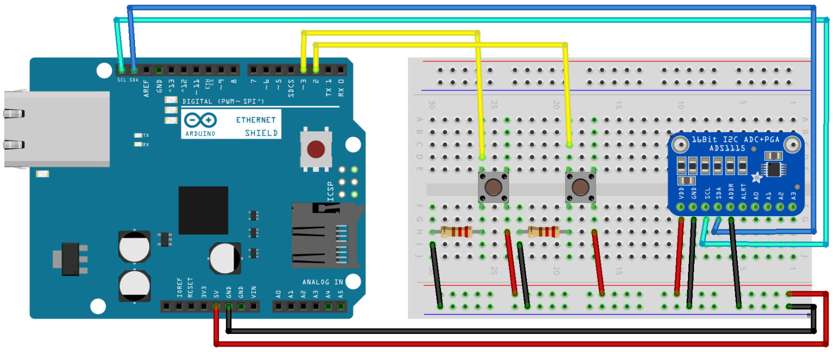

Figure 6.2: Arduino hardware validation layout

6.2 Methodology

The validation process consists in two distinct phases: a preliminary one and, then, the validation core. In the preliminary test, it has been decided to use the embedded system to register a time history of the physical quantities mentioned in Section 6.1 in parallel with the IMC acquisition tool. Once the acquisition is completed, the two time histories are first over imposed each other to identify any macroscopic errors and then elaborate by the same RMH algorithm provided by the TDOA methodology. The results are compared and discussed before proceeding with the actual validation. Afterwards, the second phase of the validation process is carried out, but this time the TDES will operate performing the tasks for which it is intended, i.e. it will elaborate just the data acquired in real time, giving as output a .CSV file including the RMH and no more registering the analog signals time history. In parallel, the IMC device registers the time history which will be elaborated once the test ended up. Therefore, the RMH computed by the TDES is compared with the one generated by the IMC data.
In the following sections, the same methodology is applied only to the configuration reported in Figure 6.2, i.e. the one exploiting the ADC for the analog signal measurements, since, as demonstrated in Subsection 5.2.4, the dedicated Arduino analog pins are not suitable for this kind of applications.

6.3 Validation

In this Section the validation test procedure explained in Section 6.2 is applied. In Figure 6.3 are reported the time histories measured by the TDES, in blue, and the IMC acquisition tool, in red, respectively for the differential torque, on the left, and rotational speed, on the right. In Figure 6.4 are presented the enlargement of the figures just mentioned. From just a visual inspection, it is possible to check that the measurements performed by the embedded system are in line to the one of the IMC device along the entire acquisition time interval, e.g. also during the most dynamic manoeuvres the TDES is able to log correctly the input signals coming from the equipped vehicle, introducing practically any deviation. So, unlike what was reported for the Arduino analog pin measurement, the acquisition accuracy is suitable for general automotive applications and in particular this one.

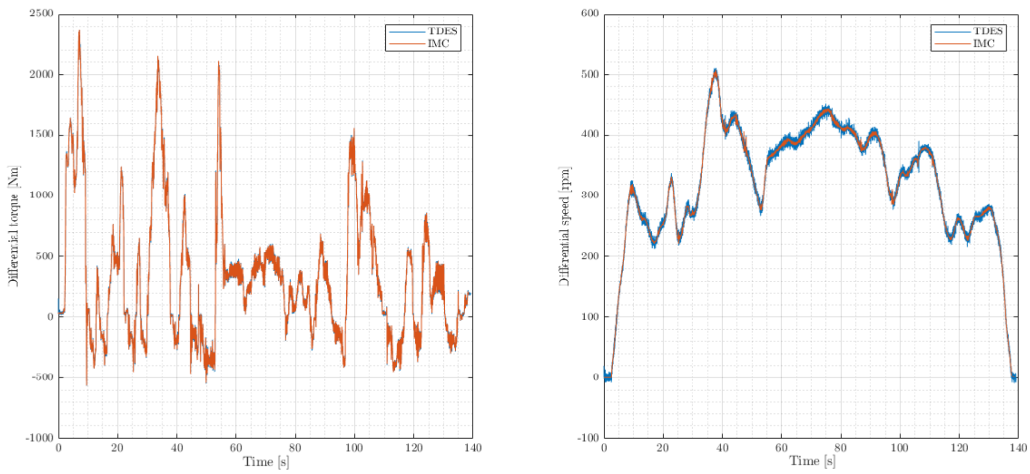

Figure 6.3: Differential torque and rotational speed time histories

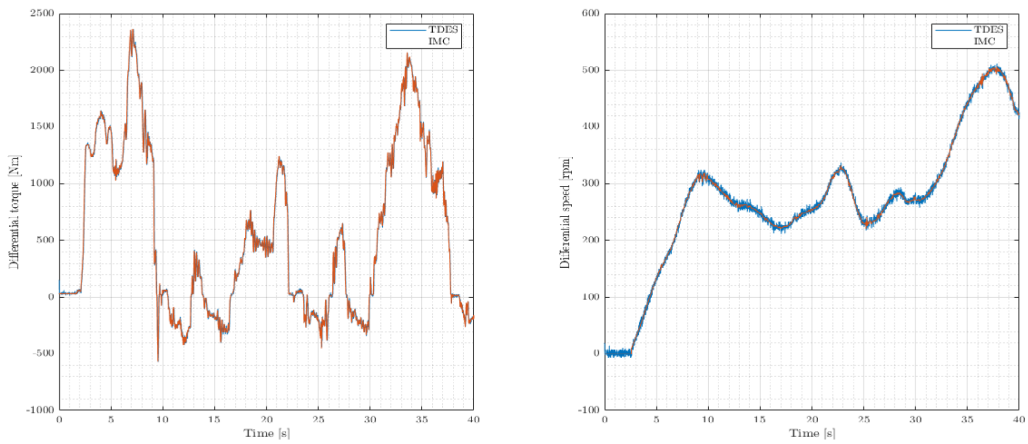

Figure 6.4: Detail of differential torque and rotational speed time histories

Once this first preliminary measurement has been carried out, the validation core test have to be performed. Now, the TDES is used to directly elaborate, in real-time, the rotating moment histogram of the differential gear, without registering any time history. In order to verify the results accuracy, the IMC device, as done for all the previous step, is configured so that it registers the differential torque and rotational speed time history so that it is possible to elaborate them in a second stage of the analysis. This validation phase has been repeated multiple times in order to have a more robust experimental data set with a larger variety both in terms of different dynamic manoeuvres and of time interval duration. In Figures 6.5 and 6.6 are reported only two rotating moment histograms computed by the TDES, in blue, and the one elaborated starting from the IMC data, computed in post-processing using the TDOA methodology, in red. In both the validation tests, the two curves over impose perfectly each other, indicating that the two RMHs are practically identical one to the other and no errors, if not insignificant, are present. To strengthen this affirmation, the equivalent torque belonging to the two RMHs is computed and reported, together with its percentage error (PE) with respect to the reference case in Table 6.1. In the same Table is also reported the total counted cycles PE, in order to give a feeling of the difference between the two approaches in terms of cycle counting accuracy.

	T_{eq} [Nm]	T_{eq} PE	Cycles PE
Test #1	380.2	0.02%	0.007%
Test #2	631.35	0.06%	0.03%

Table 6.1: Equivalent torque validation results

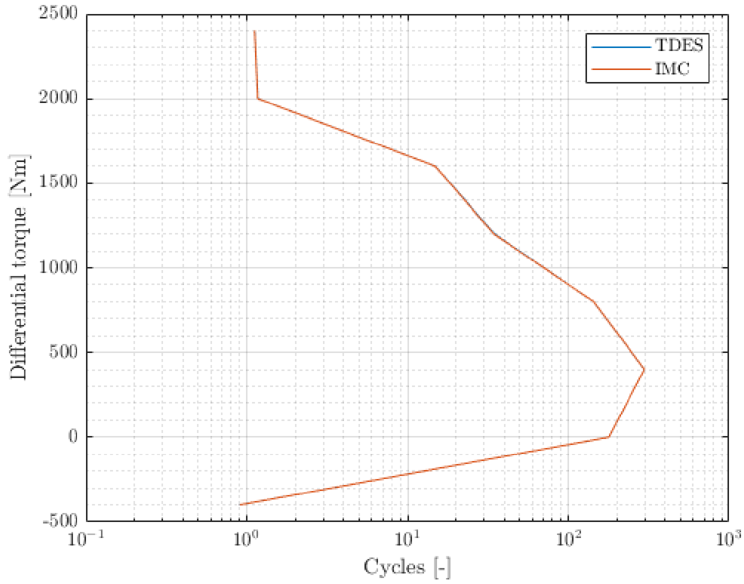

Figure 6.5: Validation test RMH

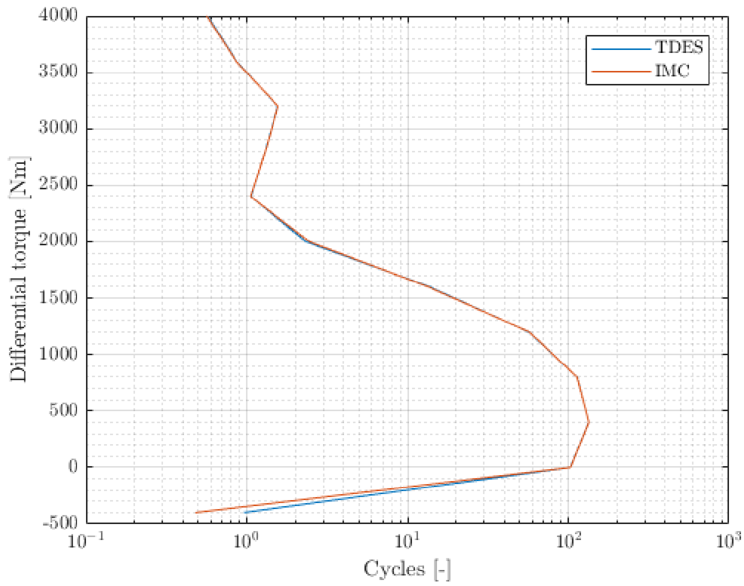

Figure 6.6: Validation test RMH

Chapter 7

Conclusions

The Transmission Durability Embedded System is intended to optimize the current testing methodology both in terms of time and economical resources. In fact, it has to unify in an unique step the data acquisition and - the most time demanding activity - the elaboration process. It is based only a low-cost prototyping platform and an additional component, i.e. the analog to digital converter. Thank to its intrinsic cheapness, the fleet of equipped vehicles, the so called “golden vehicles”, can be enlarged having as a result no more few experimental data, but a more populated data set having a more meaningful population and without reducing the measurement and results accuracies. Furthermore, the TDES, by design specification, is a transmission integrated component, meaning that it is no more correct to talk about “golden vehicles”, but “golden transmission” is a more appropriate term. In fact, it can be mounted directly to the transmission under test and then delivered as a classical gearbox to the region of destination, avoiding the delivery of an entire equipped vehicle.
The work presented in this thesis project represents only the beginning of the TDES, it has to be considered as a feasibility investigation. In any case, during this first development phase the main features and limitations of the system arose already. In fact, the embedded system demonstrates that even a low-cost prototyping platform, as Arduino is, can represent a valid alternative to the current measurement device used for mission profile analysis in automotive field. The measurement accuracy loss is unavoidable but not at all relevant in this kind of investigation and the results deriving from the elaboration of the same are essentially equal to the one obtained by using the current testing methodology and equipment.

7.1 Future developments

The state of the art of the TDES has been exposed up to now. At this point, it is worth to underline that this is only a preliminary phase of the embedded system development and that its complete potentialities are far to be fully exploited.
In this early development stage, the main limitations raised up belongs mainly to the hardware technical specifications area. The first proof to support this affirmation is the number of bin array torque classes implemented in the TDES: the TDOA methodology prescribes a greater torque working range with a higher number of classes, while the TDES, in particular the Arduino Uno board, is not able to manage and elaborate the same amount of data. A second example is represented by the number of analog signals acquirable in parallel and the number of different metrics which can be deployed to the board simultaneously. The solution to these inconvenient is to a higher clock frequency of the board microcontroller and larger memories (flash memory, SRAM and EEPROM), which means that it is enough to upgrade the current equipment to a more “expensive” version of Arduino, e.g. Arduino DUE.
Despite these limitations, the TDES featuring the most economical Arduino board is able to compute the rotating moment histogram with a reasonable number of torque classes at the differential gear. Obviously, the methodology applied to the differential can be extended to other mechanical components with minor adjustments, e.g. primary and secondary shafts or other rotating component. The first update, in order of simplicity, to be implemented is surely the equivalent torque and damage theory. Doing in this way, it would be possible to summarize an entire time history simply with just two engineering values. This reasoning can be extended to all those mechanical components present in an automotive transmission which are subject to fatigue loading, e.g. the final drive, primary shaft(s), secondary shaft(s), propeller shaft and so on. Obviously, the implementation of such methodologies relies on a more complete mission profile equipment of the vehicle, e.g. it would be necessary to measure the torque transmitted by the primary shaft rather than its rotational speed, but the main concepts remain untouched.
Furthermore, the TDES functionalities can be expanded to mechanical components different from shafts or toothed gears: a clutch for example. In Section 3.4 was illustrated the methodologies used for the clutch dissipated power evaluation, the so called power diagram. This methodology is different from the ones discussed so far because it cannot be operated in parallel, i.e. in real-time, together with the ones discussed above. In fact, the current algorithm needs the entire time history registration to work properly, but is possible to modify it in order to allow the real-time operation.

Appendix A

Code

A.1 Differential rotating moment histogram algorithm

```
function results=RMH_diff(time_step,DIFF_torque,DIFF_speed,torque_bin_array,memory)

%% Initialization
y=0;
N=length(torque_bin_array(:,1));
T=length(DIFF_speed(:,1));
results=zeros(N,1);

%% Distribution vector
for i=1:1:T
   j=1;
   while j<=N
      if DIFF_torque(i,1) >= torque_bin_array(j,1) && DIFF_torque(i,1) < torque_bin_array(j,2)
         y=j;
         j=N;
      end
   j=j+1;
   end
   results(y,1)=time_step*DIFF_speed(i,1)/60;
   results=results+memory;
   y=0;
end
end
```

A.2 I^{2}C read S-Function Builder

A.2.1 Libraries

```
#ifndef MATLAB_MEX_FILE

#include <Arduino.h>
#define ARDUINO 100

#include <Wire.h>
#include <Wire.cpp>

#include <twi.h>
#include <twi.c>

#include <Adafruit_ADS1015.cpp>
Adafruit_ADS1115 ads;

#endif
```

A.2.2 Discrete update

```
if (xD[0]!=1){
  #ifndef MATLAB_MEX_FILE
  ads.begin();
  ads.setGain(GAIN_TWOTHIRDS);
  ads.setSPS(ADS1115_DR_860SPS);
  xD[0]=1;
  #endif
}
```

A.2.3 Outputs

```
if (xD[0]==1){
  #ifndef MATLAB_MEX_FILE
  A0[0]=ads.readADC_SingleEnded(0)*0.1875/1000;
  A1[0]=ads.readADC_SingleEnded(1)*0.1875/1000;
  A2[0]=ads.readADC_SingleEnded(2)*0.1875/1000;
  #endif
}
```

A.3 Data logger S-Function Builder

A.3.1 Libraries

```
#ifndef MATLAB_MEX_FILE

#include <Arduino.h>
#include <SD.h>
#include <SD.cpp>

#include <SdFat.h>
#include <SdFatUtil.h>

#include <Sd2Card.h>
#include <Sd2Card.cpp>

#include <Sd2PinMap.h>
#include <FatStructs.h>
#include <SdFatmainpage.h>
#include <SdVolume.cpp>
#include <SdFile.cpp>
#include <SdInfo.h>

#include <File.cpp>

#include <SPI.h>
#include <SPI.cpp>

File dataFile;

int indicator = 0;

#endif
```

A.3.2 Discrete update

```
if (xD[0]!=1){
  #ifndef MATLAB_MEX_FILE

  while (indicator == 0){
    if (digitalRead(startPin[0])==HIGH){
      indicator = 1;
    }
  }
  if(!SD.begin(chipSelect[0])){
    return;
  }
  xD[0]=1;
  #endif
}
```

A.3.3 Outputs

```
if (xD[0]==1){
  #ifndef MATLAB_MEX_FILE

  if (digitalRead(savePin[0])==HIGH){
    char filename[] = "RMH00.CSV";
    for (uint8_t i = 0; i < 100; i++) {
      filename[3] = i/10 + '0';
      filename[4] = i%10 + '0';
        if (!SD.exists(filename)) {
          dataFile = SD.open(filename, FILE_WRITE);
          break;
          }
        }
    if (dataFile){
      for (uint8_t j = 0; j < 20; j++){
      dataFile.println(input[j]);
      dataFile.flush();
      }
    }
  delay(2000);
  }
  #endif
}
```

Bibliography

[1] Yung Li-Lee, Jwo Pan, Richard Hathaway, Mark Barkey, *Fatigue testing and analysis theory and practice*

[2] Robert C. Juvinal, Kurt M. Marshek, *Fundamentals of machine component design*

[3] Karl Hoffmann, *An introduction to measurements using strain gages*

[4] Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Naraig Manjikian, *Computer organization and embedded systems*

[5] Massimo Banzi, Michael Shiloh, *Getting started with Arduino*