



POLYTECHNIC UNIVERSITY OF BARI

DEPARTMENT OF ELECTRICAL AND INFORMATION ENGINEERING

AUTOMATION ENGINEERING MASTER'S DEGREE

---

Course of SENSORS AND TRASDUCERS

Prof. Eng. Francesco **ADAMO**

*Project work:*

**Theoretical and Practical Applications of Current  
Sensors and Rotary Incremental Encoders**

*Students:*

**CARAMIA** Donato  
**SOLETI** Giovanni  
**VENEZIA** Antonio

---

ACADEMIC YEAR 2019-2020



### Abstract

This project work inherent to the exam of "Sensors and Transducers" has the aim to gain in-depth knowledge of Hall effect rotary encoders and current sensors (Shunt Resistors, Hall-effect sensors and Fluxgate). As an application of these sensors, the experimental extraction of the static characteristic of a DC motor is shown. At the end of the work a Proportional controller has been developed to control the angular velocity of the motor shaft.

All code files developed in STM32CubeIDE and MATLAB relative to this project are available here: [https://github.com/giovanni-soleti/Sensors\\_And\\_Transducers\\_Proj19-20.git](https://github.com/giovanni-soleti/Sensors_And_Transducers_Proj19-20.git)



## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Current Sensors</b>	<b>4</b>
2.1	Shunt Resistor Current Sensor . . . . .	4
2.1.1	Power Module GM v1.0 . . . . .	5
2.1.2	The Data Acquisition System . . . . .	7
2.1.3	Transduction Coefficient . . . . .	8
2.2	Hall-Effect current transducers . . . . .	12
2.2.1	Working principle . . . . .	12
2.2.2	The LEM LTA 50p/SP1 Hall Effect Current Sensor . . . . .	13
2.2.3	Current measurements . . . . .	14
2.2.4	Test-bed . . . . .	14
2.2.5	Firmware development . . . . .	15
2.2.6	Results . . . . .	16
2.3	Fluxgate Current Sensors . . . . .	18
2.3.1	Measurement method . . . . .	18
<b>3</b>	<b>Rotary Encoders</b>	<b>20</b>
3.1	Hall Effect Rotary Encoder . . . . .	20
3.2	Pololu Rotary Incremental Encoder . . . . .	21
3.3	Firmware development . . . . .	22
<b>4</b>	<b>DC Motor Static Characteristic</b>	<b>24</b>
4.1	Test-bed . . . . .	24
4.2	Static characteristic of DC Motor . . . . .	25
4.2.1	Firmware development . . . . .	27
4.3	Application: Speed control . . . . .	30
4.3.1	Firmware development . . . . .	32
<b>5</b>	<b>Conclusions</b>	<b>33</b>
<b>6</b>	<b>Datasheets</b>	<b>34</b>
6.1	Pololu 70:1 Metal Gearmotor . . . . .	34
6.2	Cytron MDD3A DC Motor Driver . . . . .	39
6.3	LEM LTA50P/SP1 Current Sensor . . . . .	43
<b>References</b>		<b>45</b>
<b>List of Figures</b>		<b>46</b>



## 1 Introduction

This term paper has the objective to present current sensors analyzed during the course and mainly used in industrial application, either from theoretical and practical point of view. The simplest current sensor based on a shunt resistor will be analyzed first, setting up a data acquisition system in the OpenMultiLAB laboratory using proper instrumentation in order to calibrate the sensor. Then, due to Covid-19 outbreak, the term project has been completed in smart-working where a Hall effect current sensor and a rotary encoder have been tested and used to get the static characteristic and to implement a speed controller for a DC motor, in order to present possible applications of this kind of sensors. At the end, a brief overview of the flux gate current sensors is presented.



## 2 Current Sensors

A current sensor is a device that detects electric current in a circuit and generates a signal proportional to that current. The generated signal could be an analog voltage or current or a digital output, so that a transduction coefficient can be used to obtain the desired measurement. There are different kind of current sensors, based on different current sensing technologies and physical principles, for example:

- Shunt Resistors;
- Hall-effect sensors;
- Rogowski coils;
- Giant Magnetoresistance (GMR);
- Fluxgate.

In this project, the Shunt Resistor, Hall-effect and Fluxgate current sensors will be taken into account.

### 2.1 Shunt Resistor Current Sensor

The most common sensing element used to detect current flow is a low value, precision resistor placed in the current path. This resistor, usually called a *shunt*, develops a voltage across it that is proportional to the current passing through it. In order to limit at most the *load effect* in the circuit, the resistance value must be as small as possible when compared to the load impedance; generally speaking it should be on the order of *milliohms*. As a result, the voltage developed across the shunt resistor is also quite small, and often requires amplification before being converted by an ADC.

A typical shunt resistor current sensor is based on using IC like INA139-INA169[1], that is an high-side, unipolar current monitor, used in applications such as:

- Automotive, Telephone, Computers;
- Portable and Battery-Backup Systems;
- Battery Chargers;
- Power Management;
- Cell Phones;
- Precision Current Sources.

Power supply can range from 2.7 V to 40 V-60 V with low quiescent current ( $60\mu\text{A}$ ). The working principle can be deduced from the typical application circuit in Fig. 2.1: the load current  $I_S$  is drawn from supply  $V_S$  through the shunt resistor  $R_S$ . The voltage drop on the shunt resistor  $V_S$  is forced across RG1 by the internal operational amplifier, causing current to flow into the collector of Q1. The external resistor  $R_L$  converts the output current to a voltage  $V_O$ :

$$V_O = \frac{I_S R_S R_L}{1K\Omega} \quad (1)$$

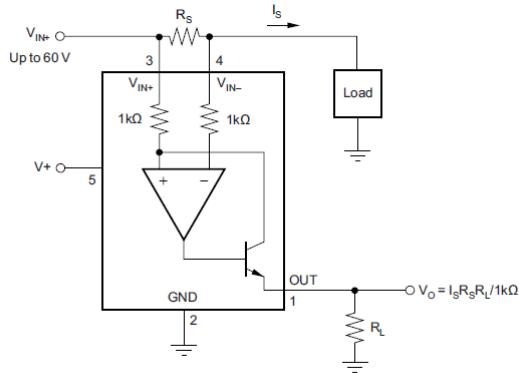


Figure 2.1: Typical Application Circuit INA139-INA169.

The maximum differential input voltage for accurate measurements is 0.5 V, which produces a 500  $\mu$ A output current. Differential measurements must be unipolar: if a more-negative voltage is applied to pin 3, the output current,  $I_O$ , is zero, but it will not cause damage.

The value selected for the shunt resistor  $R_S$  depends on the application and is a compromise between small-signal accuracy and maximum permissible voltage loss in the measurement line. High values of  $R_S$  provide better accuracy at lower currents by minimizing the effects of offset, while low values of  $R_S$  minimize voltage loss in the supply line. For most applications, best performance is attained with an  $R_S$  value that provides a full-scale shunt voltage of 50 mV to 100 mV.  $R_L$  is selected to provide the desired full-scale output voltage. The output impedance of the INA139 and INA169 is very high, which permits using values of  $R_L$  up to 100 K $\Omega$  with excellent accuracy. The input impedance of any additional circuitry at the output must be much higher than the value of  $R_L$  to avoid degrading accuracy.

### 2.1.1 Power Module GM v1.0

An example of using this type of current sensor could be the power module, embedded on small-/medium sized UAVs (Unmanned Aerial Vehicles) flight control systems, where it is important for monitoring the current consumption and the voltage of the onboard LiPo battery.

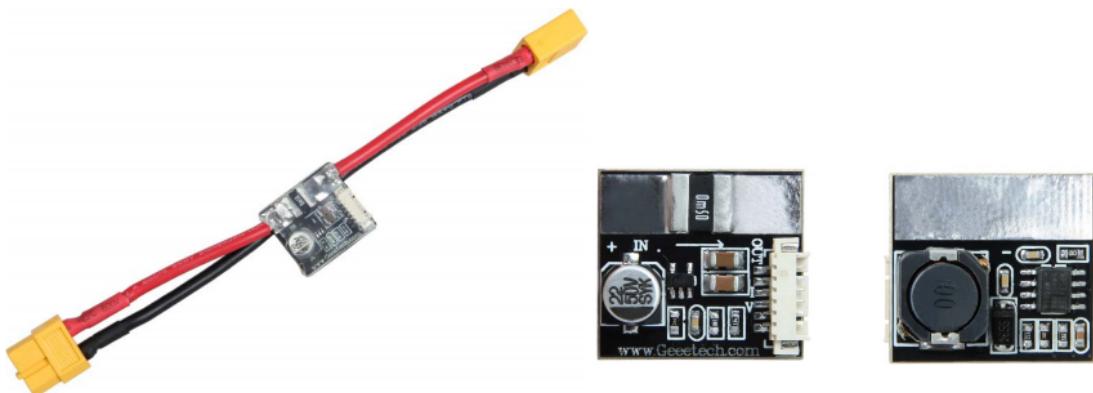


Figure 2.2: PowerModule GM v1.0.

In the case of study, the POWER MODULE GM v1.0 (Fig. 2.2 has been considered, and its schematic is shown in Fig. 2.3). The main characteristics of this kind of module are the following:

- based on INA169;
- max voltage sensing: 30 V;
- max current sensing: 90 A;
- shunt Resistor  $R_S$  : 0.50 mΩ;
- load Resistor  $R_L$  : 110 kΩ;

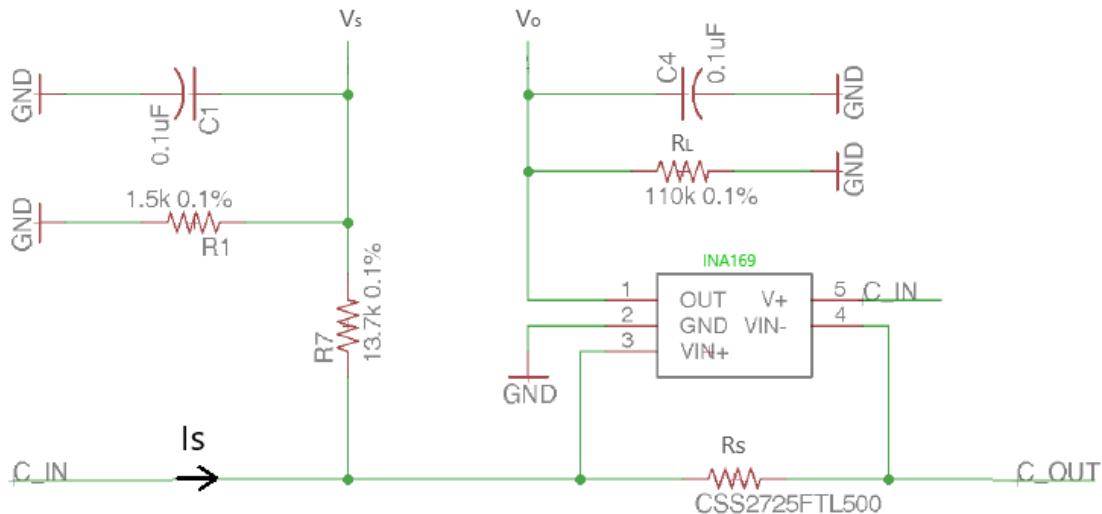


Figure 2.3: Schematic Power Module GM v1.0.

By considering Equ. 1, the current transduction coefficient for the Power Module is:

$$K_a = \frac{V_o}{I_s} = \frac{R_S R_L}{1k\Omega} = 0.055 \frac{V}{A}$$

while the voltage transduction coefficient can be derived by considering the voltage divider composed by  $R_1$  and  $R_7$  of the schematic in Fig. 2.3:

$$K_v = \frac{V_s}{V_{in}} = \frac{R_1}{R_7 + R_1} = 0.099 \frac{V}{V}$$

The aim of this test is to verify that the sensor transduces voltage and current properly, obtaining its calibration curve. Consequently, a test-bed has been set up in order to characterize the sensor, determining the unknown current and voltage transduction coefficient,  $K_a$  and  $K_v$  respectively.



### 2.1.2 The Data Acquisition System

The instrumentation used for the sensor characterization are the following:

- a laboratory power supply, Rigol DP831
- a digital multimeter, Agilent 34410A
- a programmable electronic load, Hochert&Hackl ZS506-4



Figure 2.4: Instrumentation.

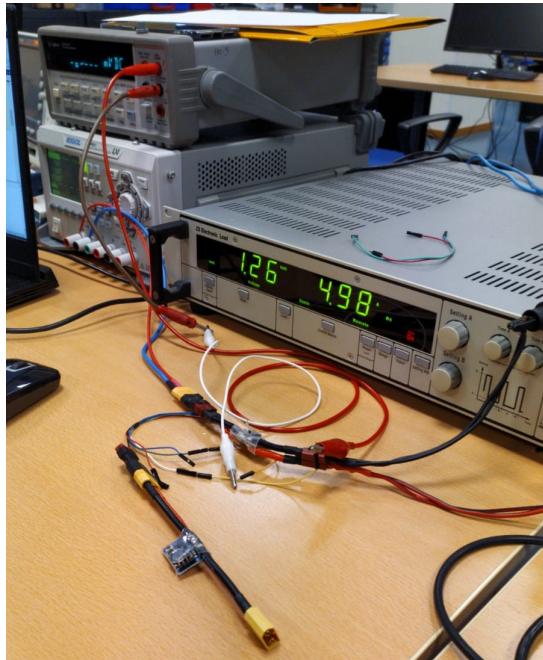


Figure 2.5: Building up the data acquisition system

Thanks to the programmability of the electronic load, different measurements can be done by directly varying the load current and the supply voltage, so its resistance. This can be done by sending appropriate text-based instructions using a standard communication protocol based on a software communication library called VISA (Virtual Instrument Software Architecture) defined by Keysight (formerly Agilent Technologies) and National Instruments for communicating with instruments regardless of the hardware interface. In the case of study, each instrument was connected on a bus by using the GPIB (General Purpose Interface Bus) standard. MATLAB running



the Instrument Control Toolbox is used to set up the instruments and acquire data, according to each test requirement. Fig. 2.6 shows how to set up the instruments.

```
%% Setting instrumentation

% Hochert&Hackl (ZS Electronic Load) Programmable Load
pl = visa('ni', 'GPIB0::7::INSTR');
% DMM Agilent 3441xA Multimeter
dmm = visa('ni', 'GPIB0::3::INSTR');
% Rigol DP831 Power Supply
rigol = visa('ni', 'USB0::0x1AB1::0x0E11::DP8F194600477::INSTR' );
% Clean up the instruments
cuf = onCleanup(@()cleanup(ps, pl, dmm, scope));
% Open communication to instruments
fopen(pl);
fopen(dmm);
fopen(rigol);
% Asking for instrument IDs
query(pl, '*idn?', '%s', '%s')
query(dmm, '*idn?', '%s', '%s')
query(rigol, '*idn?', '%s', '%s')
% Setting instrument operation mode
fwrite(pl, 'mode:curr');
fwrite(pl, 'output on');
fwrite(rigol, 'output on');
fwrite(rigol, ':OUTP CH1,ON')
fwrite(rigol, sprintf('volt %f V', 8));
```

Figure 2.6: Instrumentation set up.

### 2.1.3 Transduction Coefficient

Due to limited amount of current provided by the power supply, the characterization has not been done on the full scale of the sensor (recall up to 90A). For this reason, in the linearly spaced range of [0.25-4.9]A,  $N_p = 25$  characterization points have been considered, with a number of measurements for each point equal to  $N_m = 10$ , so that a point is obtained by the mean of its  $N_m$  measurement. Once the process is done, the current transduction coefficient has been determined as:

$$K_a = \frac{V_o}{I_s} \quad (2)$$

where  $V_o$  is the output voltage and  $I_s$  is the current that flows through the Shunt Resistor. Fig. 2.7 shows how to calculate the transduction coefficient for each measurement, with MATLAB.



```
%% Measurements

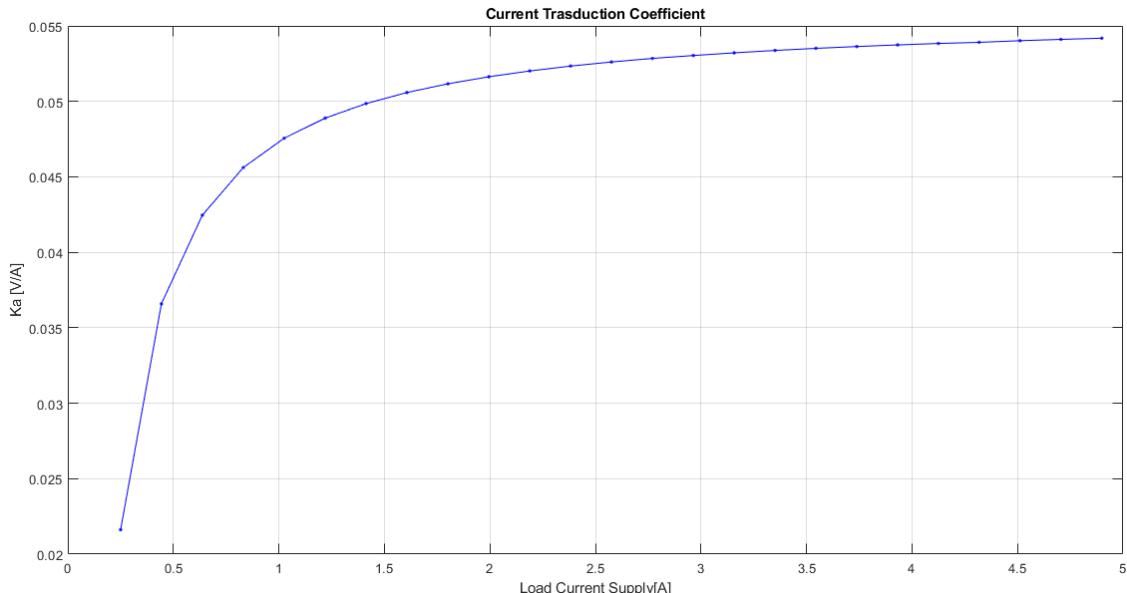
for k = 1:Np
    % Setting load current supply for each test
    fwrite(pl, sprintf('curr %f A', Iout0_c(k)));
    pause(0.5);
    for i = 1:Nm
        % Storing Nm load current supply and sensor output voltage
        ic(k, i) = query(pl, 'meas:curr?', '%s', '%f');
        vc(k, i) = query(dmm, 'meas:volt?', '%s', '%f');
    end
    % Calculating current trasduction coefficient
    Iout0_c(k) = mean(ic(k, :));
    Vout_c(k) = mean(vc(k, :));
    ka(k) = Vout_c(k)/Iout0_c(k);

end
```

Figure 2.7: Data Acquisition.

Fig. 2.8 shows the values of  $K_a$  obtained by varying the load current supply. As can be seen, due to a large full scale of the sensor considered, the transduction coefficient appear to be constant only at higher values of load current supply. Fig. 2.9 shows the linear regression of the experimental data and the point-to-point error between experimental data and the regression line. As can be noticed, the regression line fit the experimental data with low point-to-point error, consequently its slope can be considered as the transduction coefficient and equal to

$$K_a = 0.0559 \frac{V}{A}$$

Figure 2.8:  $K_a$  for each measurement

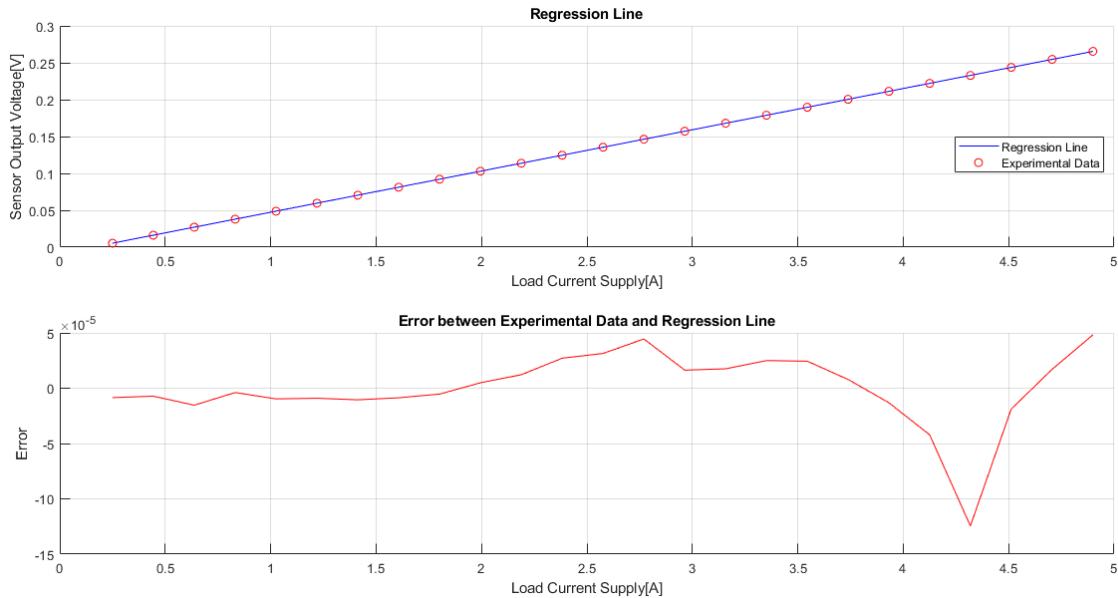
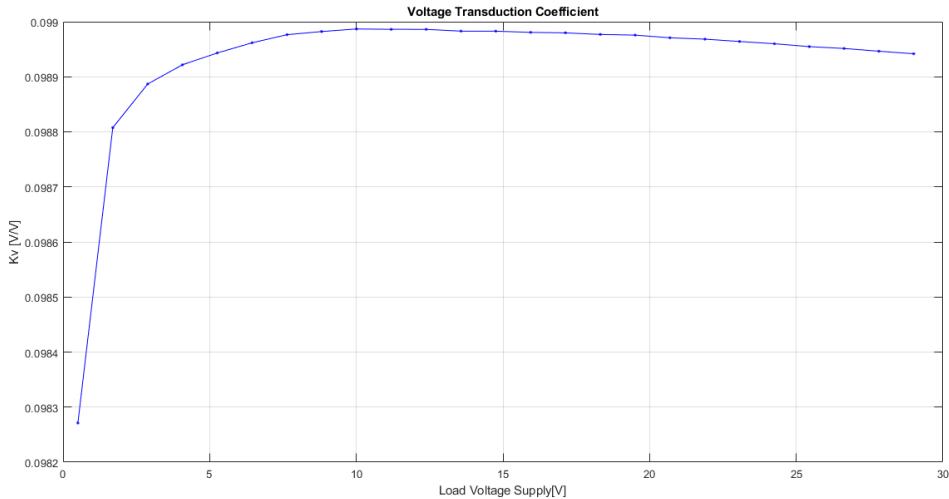


Figure 2.9: Linear Regression.

Same considerations can be done for the determination of the voltage transduction coefficient  $K_v$ , obtaining:

$$K_v = 0.099 \frac{V}{V}$$

Figure 2.10:  $K_v$  for each measurement.

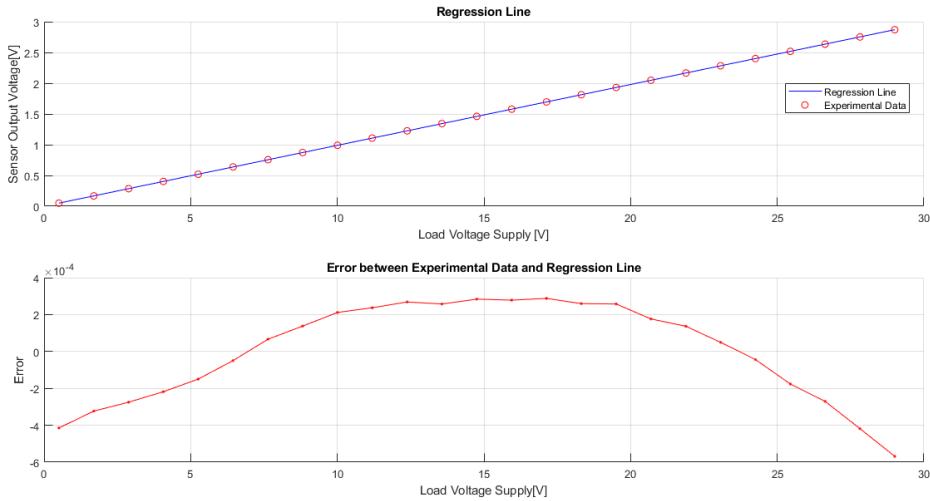


Figure 2.11: Linear Regression

As can be seen, the transduction coefficient  $K_v$  appear to be nearly constant only at the full scale of the measurement, noticing the same behavior as the current measurements.

## 2.2 Hall-Effect current transducers

Due to Covid-19 outbreak our laboratory project has been interrupted and, consequently, no more tests could be done on the POWER MODULE GM v1.0. In order to complete the project in smart-working, a LEM LTA 50P/SP1 current sensor has been used instead. The sensor employing the Hall effect to measure DC and complex waveform AC current providing a galvanic isolation between the control signal and the measured current.

### 2.2.1 Working principle

The LEM LTA 50P/SP1 is a sensors based on Hall-Effect so, if a current  $I_c$  flows through a sheet of conducting material, a voltage  $V_h$  which depends on  $I_p$  is generated, in particular:

$$V_h = \frac{K}{d} \cdot I_c \cdot B + V_{0H} \quad (3)$$

where:

- K is the Hall constant,
- d is the thickness of the sheet,
- B is the external magnetic flux,
- $V_{0H}$  is the offset voltage of the Hall Generator.

Therefore the output of the Hall Generator is proportional to the primary current, with the offset  $V_{0H}$ .

In the closed loop Hall Effect current transducers,  $V_h$  creates a compensation current  $I_s$  which generate a flux equal in amplitude but opposite in direction to the primary flux

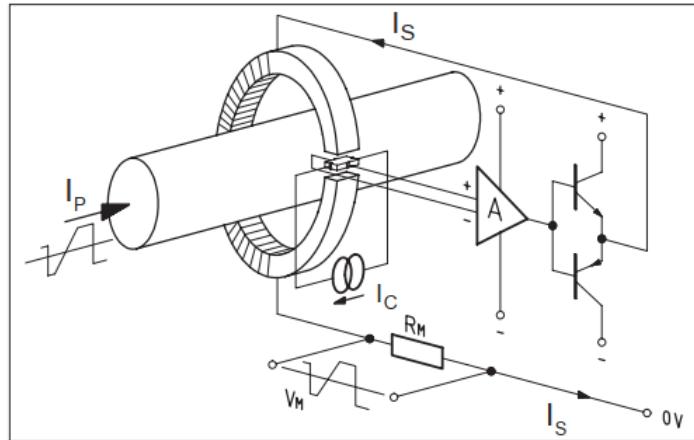


Figure 2.12: Hall effect electronic circuit [2].

When the magnetic flux is fully compensated the magnetic potential of the two coils are identical:

$$N_s \cdot I_s = N_p \cdot I_p \quad (4)$$

as shown in Fig. 2.12. Operating the Hall generator in a zero flux condition eliminates the drift of gain with temperature. An additional advantage of this configuration is that the secondary winding will act as a current transformer at higher frequencies, significantly extending the bandwidth and reducing the response time of the transducer [2].



### 2.2.2 The LEM LTA 50p/SP1 Hall Effect Current Sensor

The LEM LTA 50P/SP1 is characterized by:

- $\frac{N_s}{N_p} = 1000$ ;
- Primary RMS nominal current  $I_p = 50$  A;
- measure resistance  $R_M = 100 \Omega$ .

In our study case the measured current is in the order of few amperes, therefore to use the sensor near the full scale, as shown in Fig. 2.13, five primary turns are added in order to reduce the full scale of the sensor as described later.

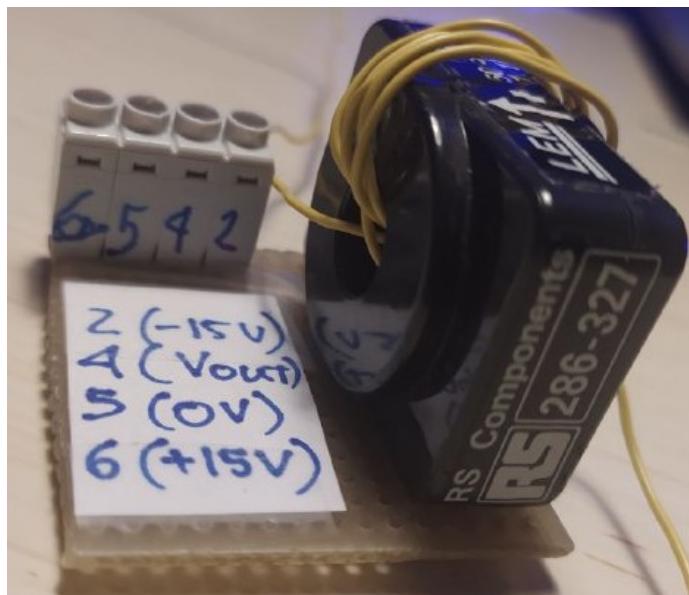


Figure 2.13: LEM LTA/50p/SP1



### 2.2.3 Current measurements

The current  $I_s$  is measured using the voltage drop  $V_{out}$  across  $R_M$ . The transduction coefficient  $k_T$ , can be evaluated considering the parameters shown in Fig. 2.14.

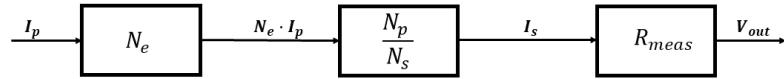


Figure 2.14: How to calculate  $k_T$ .

$$k_T = \frac{V_{out}}{I_p} = \frac{N_e \cdot N_p \cdot R_{meas}}{N_s} = 500 \frac{mV}{A} \quad (5)$$

where:

- $N_e$  equal to 5, is the number of the external turns;
- $\frac{N_p}{N_s}$  equal to 1 over 1000, is the turns ratio;
- $R_{meas}$  equal to 100  $\Omega$ , is the measure resistance.

In order to reduce measurement noises,  $V_{out}$  is obtained as the mean of a given number of samples coming from an ADC, embedded on the STM32-F446RE Nucleo Board, in a fixed time window. Considering a time window  $T_w$  of 100 ms and a 12 bit ADC with a clock frequency  $f_s$  of 21 MHz, the number of samples computed is given by:

$$N_{sample} = \frac{f_s \cdot T_w}{nBits + 3} = \frac{21 \cdot 10^6 \cdot 0.1}{15} \simeq 140\,000 \quad (6)$$

### 2.2.4 Test-bed

In this section the test-bed is shown (Fig.2.15), used to test the sensor. It is composed by:

- PC power supply;
- Hall effect current sensor, LEM LTA 50p/SP1;
- DC motor;

The Hall effect sensor measures the current required by the DC motor running in no load condition.

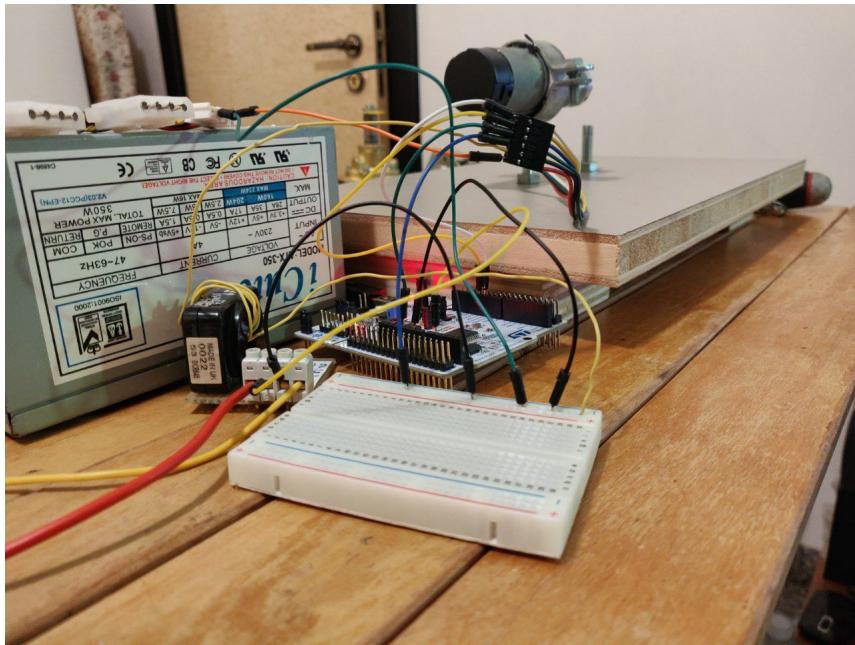


Figure 2.15: Test-bed.

### 2.2.5 Firmware development

Due to the amount of samples collected during the measurements, an ADC in DMA mode is used. It has been set-up in Circular mode, so that when the DMA Buffer fills-up, a dynamic average is computed as shown in Fig. 2.16.

```
void HAL_ADC_ConvCpltCallback(ADC_HandleTypeDef* hadc)
{
    counter++;
    float buffer_mean = get_mean();
    raw_mean = buffer_mean/counter + ((float)(counter-1)/counter)*raw_mean;
}

float get_mean()
{
    uint32_t sum = 0;
    for (int i = 0; i < n_samples; i++)
    {
        sum += buffer_raw[i];
    }
    return (sum/n_sample);
}
```

Figure 2.16: Callback that computes the dynamic average.

Furthermore, a base timer in Interrupt Mode has been used to create a time window. As shown in Fig. 2.17, when the interrupt is fired, the ADC output average is trasduced in current and sent on the UART.

```

void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
{
    char msg[30];
    float inputV = raw_mean * Vref / 4095;
    double Current = (double)inputV */ kT;
    sprintf(msg, "Current[mA] : %1.2f\r\n", Current);
    counter = 0;
    raw_mean = 0;
    HAL_UART_Transmit(&huart2, (uint8_t*)msg, strlen(msg), 200);
}
    
```

Figure 2.17: Callback implemented to trasduce the average ADC Value in current and send it on UART.

### 2.2.6 Results

Once the test-bed has been set-up, very noisy data have been collected from the sensor. In order to find the causes of the noise, a series of simulations have been done by varying the time window  $T_w$ , setting it up to (non)multiple of the suspected current ripple running at 50Hz (T = 20ms) and provided by the test-bed because of a non perfect isolation of the circuit respect to the ground. Fig. 2.18 shows measurements with a time window  $T_w$  multiple of the ripple current period while Fig. 2.19 shows measurements with a time window  $T_w$  non-multiple of the ripple current period.

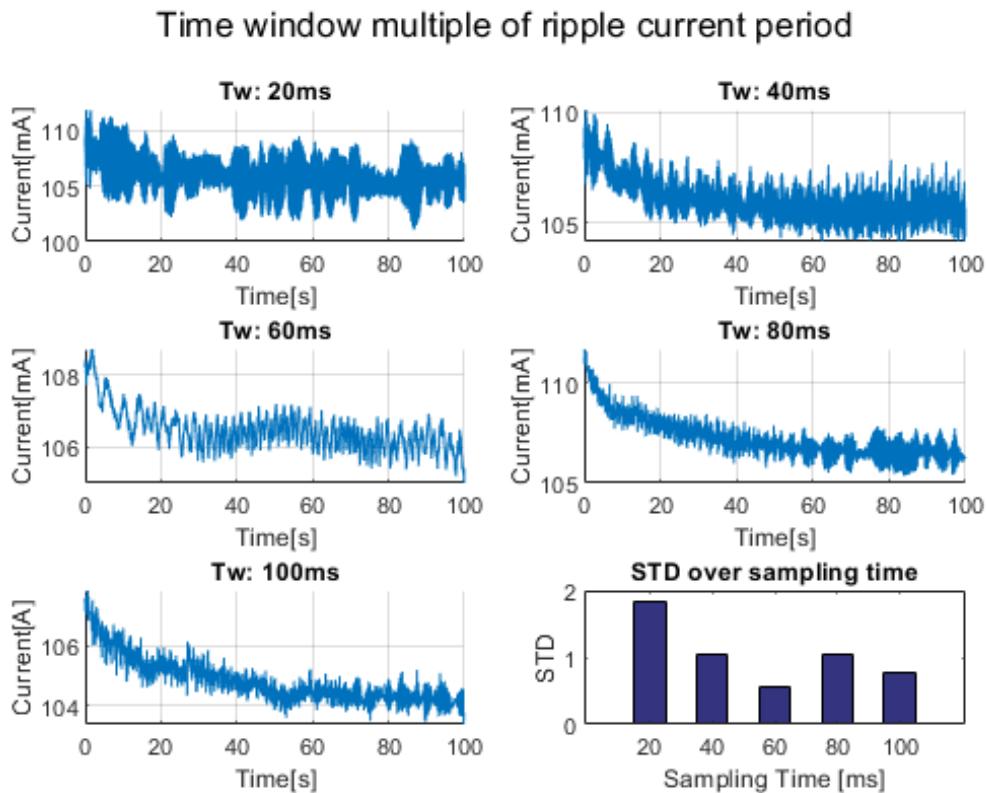


Figure 2.18: Current measured changing the time window with a multiple of 20 ms.

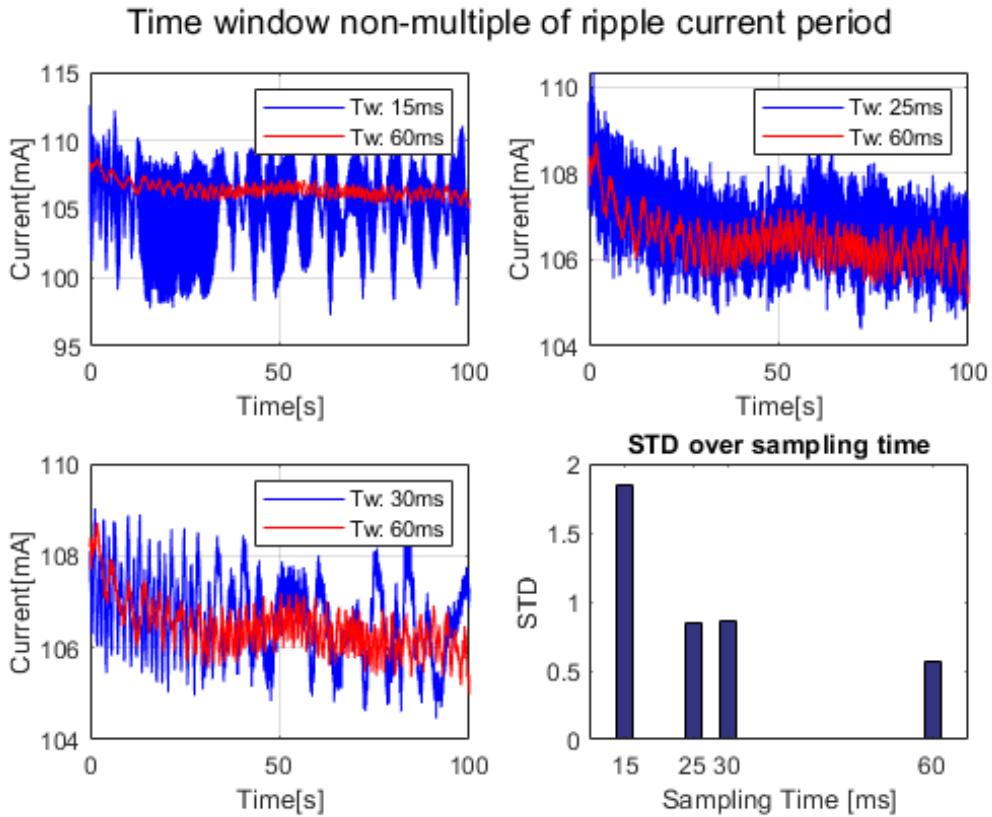


Figure 2.19: Current measured changing the time window with a non-multiple of 20 ms.

At the end, Fig.2.20 shows the offset current when the sensor is not inserted in a closed circuit. It is caused by the residual magnetization of the core.

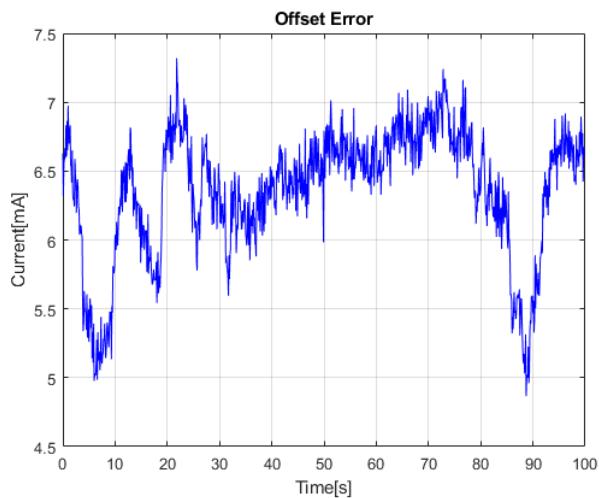


Figure 2.20: Offset error over time considering and time window of 100 ms.

### 2.3 Fluxgate Current Sensors

Fluxgate technology is one of the most used method for accurate measurement of current. This method consists in compensating the current linkage  $\theta_p$ , which is the linkage generated by measured current, with an opposing current linkage  $\theta_s$ .

As shown in Fig. 2.21, the compensation is done when:

$$N_p \cdot I_p = N_s \cdot I_s \quad (7)$$

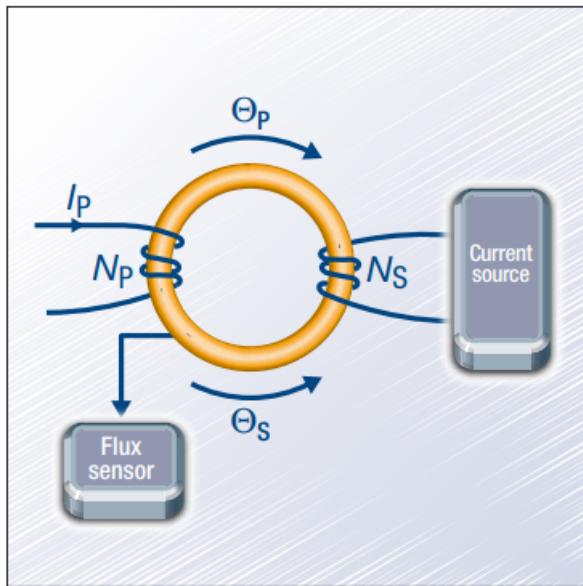


Figure 2.21: Fluxgate working principle.

#### 2.3.1 Measurement method

When applying a square wave voltage (Fig. 2.22a) to a saturable inductor until its magnetic core starts to saturate, a current (Fig. 2.22b) is created. This current flowing through a measuring resistor will provide a symmetric voltage relative to zero with peak values  $+V = -V$ . When the measured current flows through the aperture core, the curve is shifted (Fig. 2.22c), so  $+V \neq -V$ , so the error voltage is supplied to a power amplifier, shown in Fig. 2.23, that drive a current  $I_s$  until  $+V \neq -V$  [3]. At the end, known  $I_s$ , the measured current  $I_p$  is given by:

$$I_p = \frac{I_s \cdot N_s}{N_p} \quad (8)$$

In order to avoid error of flux detector, a second D' winding is mounting on. In this way the residual flux will create a very small voltage, so the remaining signal is correlated to the fluxgate excitation as shown in Fig. 2.24. Main problem of this technologies is the bandwidth which is low, and to increase it, a fourth winding W is wound to extend frequency range.

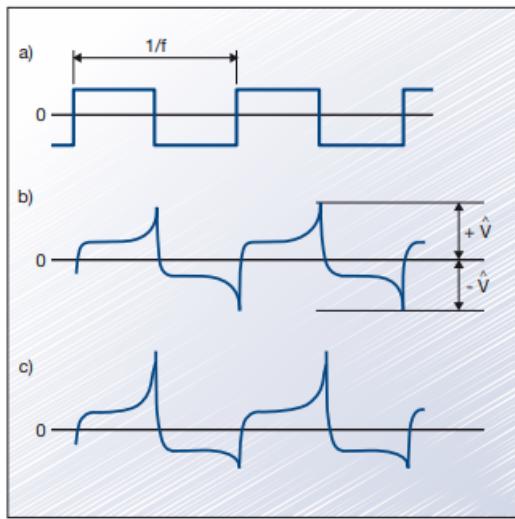


Figure 2.22: Square wave voltage (a), Current created (b), Asymmetry of the created current (c)

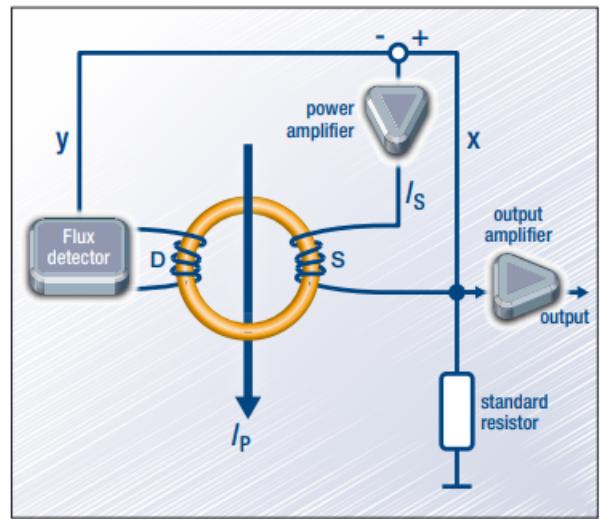


Figure 2.23: Simplified base circuit for DC current compensation.

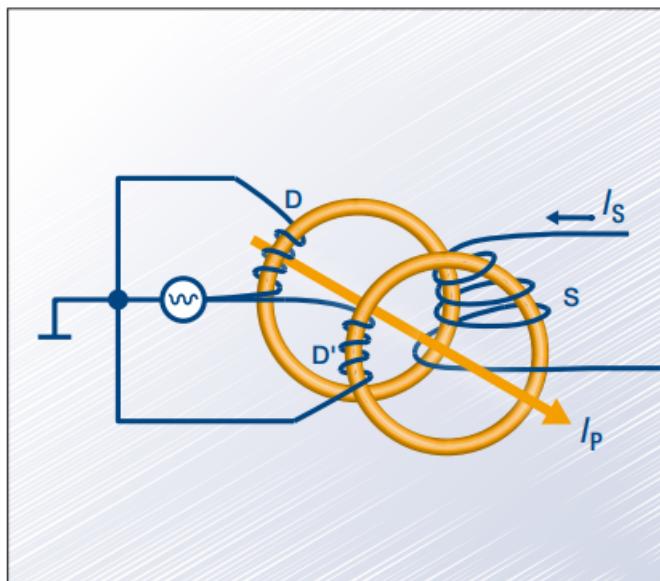


Figure 2.24: Solution against voltage peaks re-injection.



### 3 Rotary Encoders

A rotary encoder is a type of sensor used to detect the angular position or movement of a rotating shaft. The goal of this kind of sensor is to convert the rotational motion of the motor shaft into analog or digital signals. There are three major sensor types used in rotary encoders:

- **Mechanical** are common mainly because they are so cheap and easy to implement. However, the necessity of metal components coming into contact with each other at rapid speeds will inevitably result in mechanical wear;
- **Optical** rotary encoder relies on a photosensor panel that generates electrical signals when it gets hit by light (Laser). The laser are mounted on the rotating element which generate signals through photosensor. Typically the outputs are 2 PWM signals which allow to determine speed and direction. However, the components of optical rotary encoders are a bit fragile, making them unsuitable for highly rugged applications;
- **Magnetic** encoder uses the same principle to determine a position as an optical encoder, but it does it using magnetic fields rather than light. There are 3 major types of magnetic encoders: Magnetic Gear Tooth Sensor or Pickup, Magneto-Resistive Encoder and Hall-Effect Magnetic Encoders. Since the method uses non-contact technology, it remains robust without the danger of early mechanical wear and tear.

There is another type of classification about rotary encoders:

- **Absolute** rotary encoder, the data reported by the encoder indicates the absolute position of the rotating element. In more concrete terms, absolute encoders can provide data on speed, position, and direction by outputting discrete bits of data instead of a continuous data stream. Generally, the benefits of absolute encoders are that they are capable of higher data resolution and that they have better start up performance since they do not need to be calibrated periodically. However, they are also more complex to implement, modify, or troubleshoot;
- **Incremental** rotary encoder typically produce a continuous data stream. A classic example is a sinusoidal wave generated by magnetic rotary encoders which are used as a means of determining rotational speed. They cannot record absolute position. However, they are excellent choices for providing speed and distance feedback. They use fewer sensors than absolute encoders and thus are cheaper and easier to implement. The simplicity of incremental sensors also gives them a higher degree of versatility.

#### 3.1 Hall Effect Rotary Encoder

The Hall effect magnetic encoders use a magnetic phased array sensor to average the signal over multiple detectors. This magnetic phased array technology is available in an IC which integrates both the sensor and the processor on the same chip which considerably decreases the complexity for a robust, compact, easily manufacturable component. As showed in Fig. 3.1

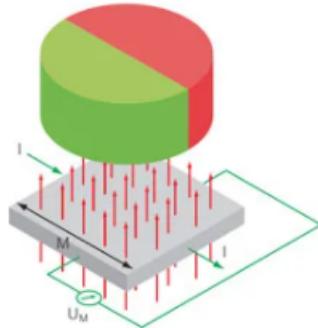


Figure 3.1: Hall Effect Sensor.

this sensor consists of a layer of semiconductor material, connected to a power supply. When a magnetic field applied to wheel or ring passes through a Hall-effect sensor, the interaction generates a voltage signal. The amplitude and frequency of the magnetic perturbation can be used to determine speed and displacement.

### 3.2 Pololu Rotary Incremental Encoder

In our study case an incremental magnetic rotary encoder has been used and shown in Fig. 3.2.

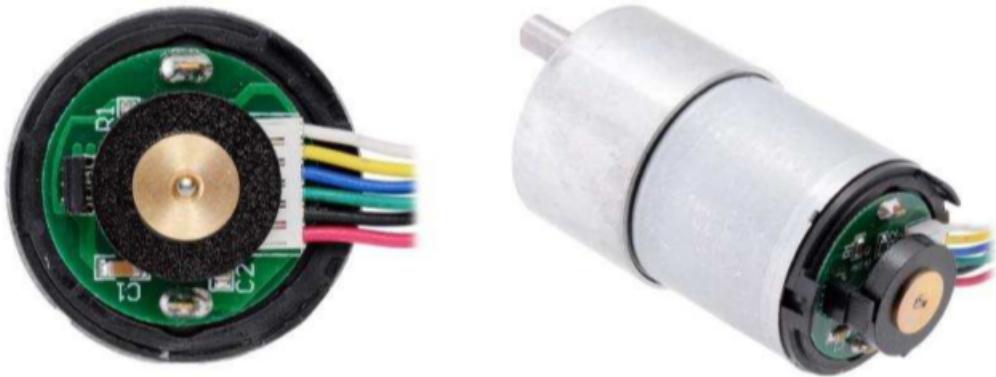


Figure 3.2: Pololu Rotary Encoder.

Two Hall-effect sensors are used to sense the rotation of the motor shaft. The encoder electronics and magnetic disc are enclosed by a removable plastic end cap.

The quadrature encoder provides a resolution of 64 counts per revolution (CPR) of the motor shaft when counting both edges of both channels. To compute the counts per revolution of the gearbox output, multiply the gear ratio by 64. The Hall sensors require an input voltage,  $V_{cc}$ , between 3.5 V and 20 V and draw a maximum of 10 mA. The A and B outputs are square waves from 0 V to  $V_{cc}$  approximately 90° out of phase. The speed of the motor can be determined from the frequency, and the direction of rotation can be determined from the order of the transitions. Counting both the rising and falling edges of both the A and B outputs results in 64 counts per revolution of the motor shaft, as showed in Fig. 3.3:

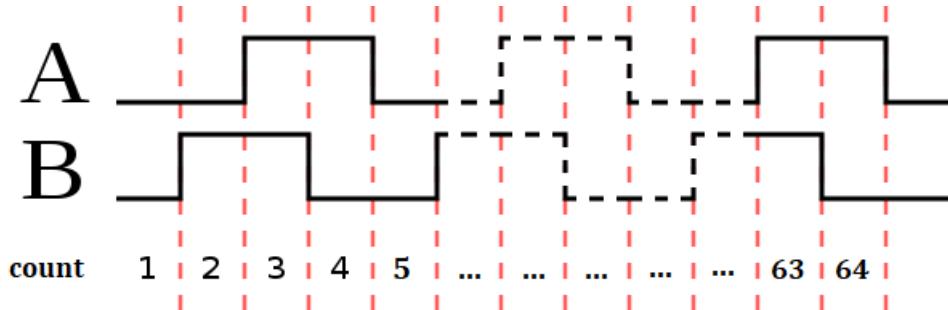


Figure 3.3: Output square waves, A and B.

Using only a single edge of one channel results in 16 counts per revolution of the motor shaft, so the frequency of the A output in the above oscilloscope capture is 16 times the motor rotation frequency.

### 3.3 Firmware development

In order to update the value of speed, the general purpose timer (TIM3) have been used. It senses the data in output from the motor rotary encoder. Timer are used in Encoder Mode, so the Counter Register (CNT) increases by one when the rotational speed is positive and decreases by one otherwise.

Rotational speed  $w$  (rpm) is given by `get_speed` function, as showed in Fig. 3.4:

```
double get_speed()
{
    /* Calculate system states */
    count2_encoder = _HAL_TIM_GET_COUNTER(&htim3);
    double difference_count_motor = (double)difference_count(count2_encoder, count1_encoder, htim3);
    count1_encoder = count2_encoder;
    return (((double)difference_count_motor * 60)/(gear_ratio_motor * time_elapsed * CPR_encoder_motor)); // rpm
}
```

Figure 3.4: `get_speed` function.

where:

- `time_elapsed` is the sampling time, 0.1s;
- `gear_ratio_motor` is the gear ratio of the motor, 70;
- `CPR_encoder_motor` is the count per rotation of the encoder, 64;
- `count2` and `count1` are two consecutive value of CNT Register;
- `difference_count` used to avoid the counter overflow when the difference between `count2` and `count1` is computed, the function showed in Fig. 3.5 has been implemented.



```
int32_t difference_count(uint32_t cnt2,uint32_t cnt1,TIM_HandleTypeDef htim)
{
    /* Function used to avoid count overflow */
    int32_t difference = cnt2 - cnt1;
    if (__HAL_TIM_IS_TIM_COUNTING_DOWN(&htim) && (cnt2 > cnt1))
    {
        difference = cnt2 - __HAL_TIM_GET_AUTORELOAD(&htim) - cnt1;
    }
    else if ((__HAL_TIM_IS_TIM_COUNTING_DOWN(&htim) == 0) && (cnt1 > cnt2))
    {
        difference = -cnt1 + __HAL_TIM_GET_AUTORELOAD(&htim) + cnt2;
    }
    return difference;
}
```

Figure 3.5: Function implemented to avoid overflow when the difference is computed.



## 4 DC Motor Static Characteristic

One of the most used actuator in industrial automation is the DC motor, having a static characteristic as shown in Fig. 4.1, where the torque provided on the motor shaft is related to its angular velocity, useful to get the working point of the motor when a load torque is applied. This term paper has the aim to show examples of using sensors in industrial applications so that the static characteristic of a DC motor has been determined using the LEM LTA50PSP1 as current sensor and the incremental rotary encoder as angular velocity sensor, each of them discussed in the previous sections.

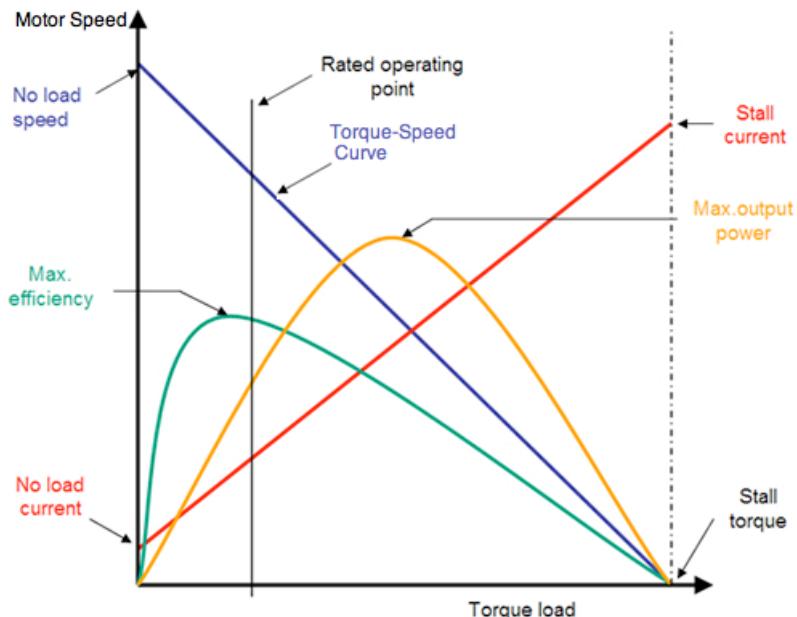


Figure 4.1: Example of Static characteristic of DC Motor.

### 4.1 Test-bed

The test-bed is shown in Fig.4.2a and Fig.4.2b. As can be seen, it is mainly composed by:

- a power supply;
- a current sensor, LEM LTA50PSP1;
- a DC motor, the Pololu 70:1 Metal Gearmotor 37D with rotary incremental encoder;
- a motor driver, the Cytron MDD3A;
- STM32F446RE Nucleo Board.

The datasheets of each components are reported in the Datasheet section.

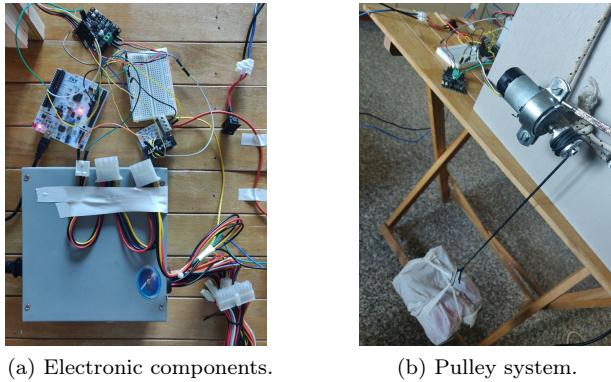


Figure 4.2: Test-bed.

To obtain the static characteristic, load torque will be altered in order to measure the armature current and angular velocity.

A pulley system with a variable weight is built-up in order to change the load torque as shown in Fig. 4.2b.

For each weight, N measures of armature current and angular velocity are collected to avoid measure error. In the Tab. 4.1, for each weight raised, the mean of armature current and angular velocity are shown respectively.

Weight	Current	Speed
No load	0.20 A	64 RPM
280 g	0.29 A	60 RPM
400 g	0.35 A	57 RPM
720 g	0.45 A	54 RPM
1040 g	0.56 A	50 RPM
1390 g	0.70 A	46 RPM
1800 g	0.86 A	42 RPM
2600 g	1.08 A	35 RPM

Table 4.1: Armature current and angular velocity for each weight raised.

## 4.2 Static characteristic of DC Motor

After an appropriate data acquisition, the current-speed characteristic is obtained by a linear regression of the data shown in Tab. 4.1.

In Fig. 4.3 the linear regression obtained is shown in red while blue markers represent each measure for each test done. Torque-Speed characteristic is given by recalling the so known relationship:

$$\tau = k_m I_m$$

where  $k_m$  is the constant torque provided by the data-sheet and  $I_m$  is the armature current. Moreover, mechanical power for each point is reported in cyan.

By comparing the static characteristic obtained at 5V and the one given by the data-sheet at 6V in Fig. 4.3 and Fig. 4.4 are similar, in particular:

- max mechanical power point is at 34 RPM and 36 RPM at 5 V and 6 V, respectively;

- stall current is 2.3 A and 3.1 A at 5 V and 6 V, respectively;
- no load speed is 65 RPM and 73 RPM, at 5 V and 6 V, respectively.

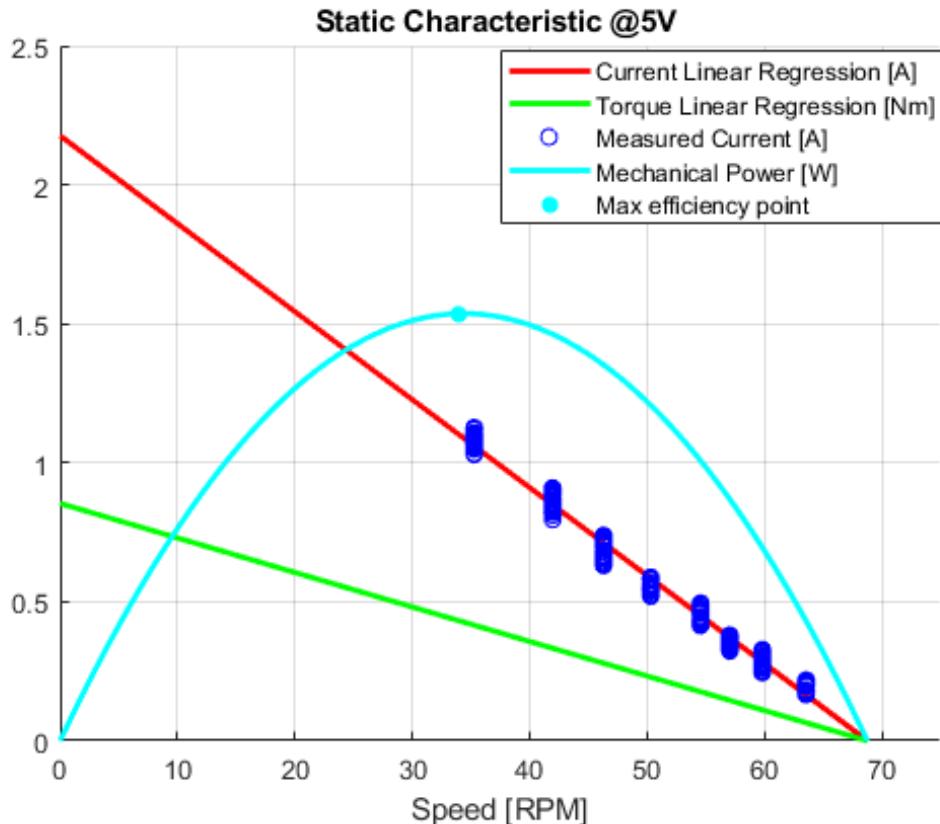
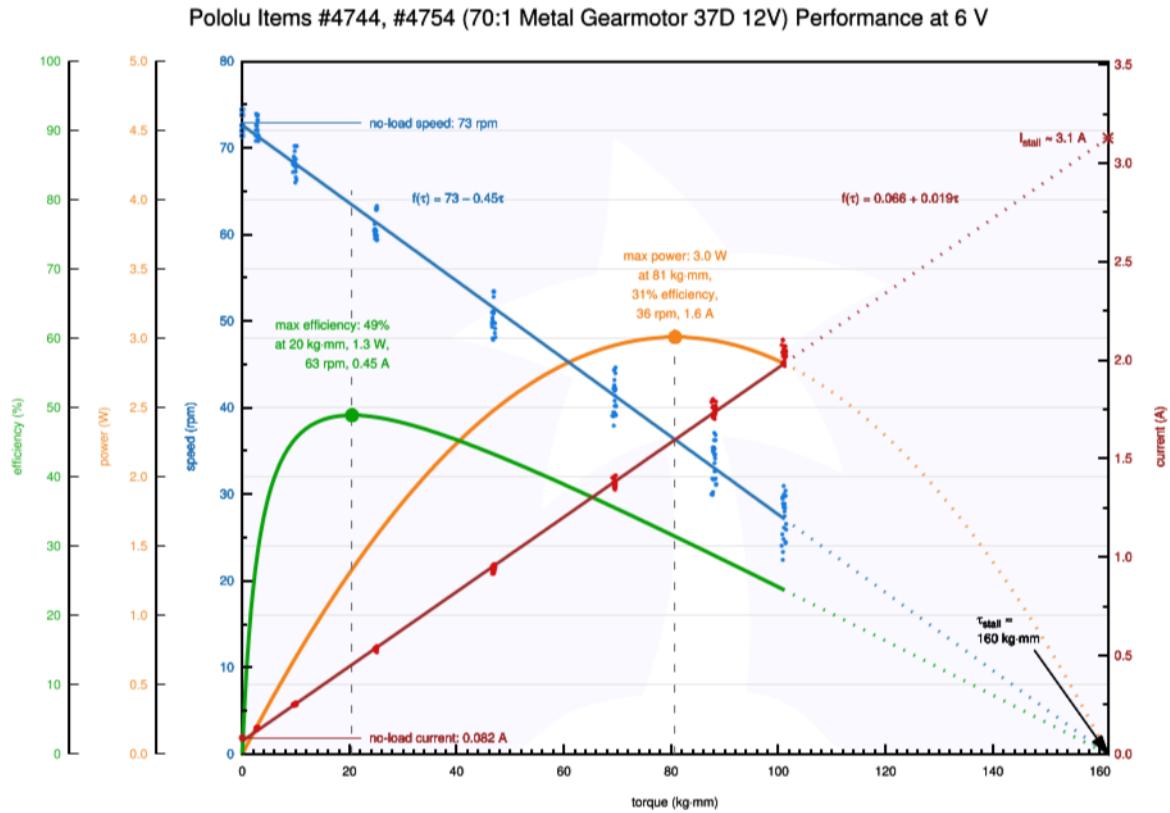


Figure 4.3: Static characteristic @5 V obtained in this study work.



January 2020 – Rev 1.2

© Pololu Corporation | www.pololu.com | 920 Pilot Rd., Las Vegas, NV 89119, USA

16

Figure 4.4: Static characteristic @6 V from data-sheet.

#### 4.2.1 Firmware development

To collect N measure for each weighted raised, a TIMER-BASE in Interrupt Mode has been used, with an event generated every 100 ms. In Fig. 4.5 the callback in which data are sent to MATLAB, using UART communication protocol, is shown. The data acquisition in MATLAB is shown in Fig. 4.6. Acquisition lasts:

$$T_{acq} = \frac{\text{count}}{2 \cdot T_c} \quad (9)$$

where:

- count is equal to 200;
- $T_c$  is equal to 0.1 s.



```
void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
{
    /*Read data from Sensors*/
    float inputV = raw_mean * 3300 / 4095;
    double current = (double)inputV * 2;
    double speed = get_speed();
    /*Send data to MATLAB*/
    HAL_UART_Transmit(&huart2, (uint8_t*)&current,sizeof(current),200);
    HAL_UART_Transmit(&huart2, (uint8_t*)&speed,sizeof(speed),200);
    counter = 0;
    raw_mean = 0;
}
```

Figure 4.5: TIMER-BASE Callback.

```
Prove_LEM_LIA.m > + | %
1 %> Initialized workspace
2 - clc
3 - clear all
4 - close all
5
6 %> Parameters definition
7 - count = 200;
8 - device = serialport("COM6",115200);
9 - data_vector = read(device,count,"double");
10
11 %% 
12 - current = zeros(1,count/2);
13 - speed = zeros(1,count/2);
14 - for i = 1 : count/2
15 -     current(i) = data_vector(2*i-1);
16 -     speed(i) = data_vector(2*i);
17 - end
```

Figure 4.6: MATLAB script for data acquisition.

After data collection for each weight raised, data are processed to obtain results showed previously. In Fig. 4.7 the MATLAB script used to get each characteristic is shown.



```
Static_Characteristic_POLULUSV.m  × + |
```

```
1 - close all
2 - clear all
3 - clc
4 - load('WorkspaceCompleto.mat')
5 -
6 - %% c_mean = 1e-3*[c_0 c_280 c_400 c_720 c_1040 c_1390 c_1800 c_2600];
7 - s_mean = [s_0 s_280 s_400 s_720 s_1040 s_1390 s_1800 s_2600];
8 -
9 - %from motor datasheet
10 - kphi = 0.3922;
11 - figure(1)
12 - hold on,grid on
13 - title('Static Characteristic @5V ')
14 - [m,q,r] = reglin(s_mean,c_mean);
15 - y = m*s_mean + q;
16 - speed = 0:1:75
17 - C = kphi*m*speed + q;
18 - I = m*speed + q;
19 - h = plot(speed, I , 'r','Linewidth',2);
20 - h1 = plot(speed,C, 'green','Linewidth',2);
21 - ylim([0 2.5]), xlim([0 75])
22 - xlabel('Speed [RPM]');
23 -
24 - h2 = plot(s_0*ones(1,length(current_f_0)),1e-3.*current_f_0,'bo')
25 - plot(s_280*ones(1,length(current_f_280)),1e-3.*current_f_280,'bo')
26 - plot(s_400*ones(1,length(current_f_400)),1e-3.*current_f_400,'bo')
27 - plot(s_720*ones(1,length(current_f_720)),1e-3.*current_f_720,'bo')
28 - plot(s_1040*ones(1,length(current_f_1040)),1e-3.*current_f_1040,'bo')
29 - plot(s_1390*ones(1,length(current_f_1390)),1e-3.*current_f_1390,'bo')
30 - plot(s_1800*ones(1,length(current_f_1800)),1e-3.*current_f_1800,'bo')
31 - plot(s_2600*ones(1,length(current_f_2600)),1e-3.*current_f_2600,'bo')
32 - w = speed/9.55;%rad/sec
33 - P = C.*w;
34 - h3 = plot(speed,P, 'cyan','Linewidth',2);
35 - legend([h1 h2 h3],'Current Linear Regression [A]', 'Torque Linear Regression [Nm]', 'Measured Current [A]', 'Mechanical Power [W]')
36 -
37 - %% Max efficiency point
38 - max_eff_ind = find(abs(diff(P)) < 0.001);
39 - max_eff_speed = speed(max_eff_ind);
```

Figure 4.7: MATLAB script in which data are processed.



### 4.3 Application: Speed control

In most industrial application in which DC motor are used a speed control is needed.

In the last part of this work has been shown how to control a DC MOTOR with a proportional controller ( $K_p = 0.05$ ), in particular three tests has been done, in which set point is settled at 30 RPM.

In the first test, there is no load torque applied, as it is shown in Fig. 4.8 angular velocity go to 30 RPM after a transitory.

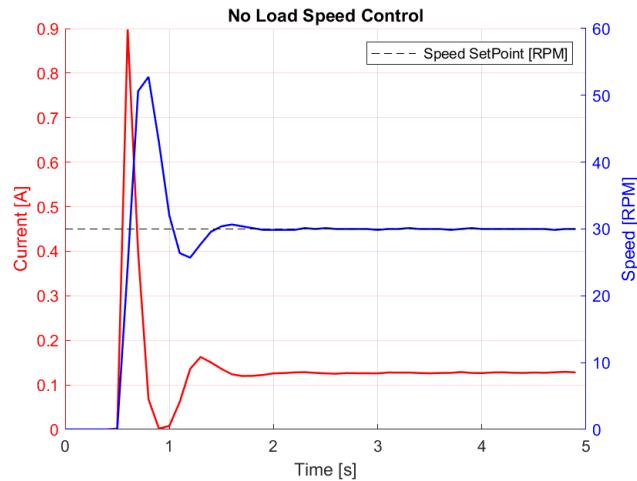


Figure 4.8: Speed control when is not applied a load torque.

In the second test, speed control is done when a constant torque is applied to the shaft. As it is shown in Fig. 4.9 when load torque change due to external disturbance, speed moves away from set point, then return to 30 RPM.



Figure 4.9: Speed control when is applied a constant torque.

In the last test, load torque change frequently. As shown in Fig. 4.10, when torque increase speed decrease and vice versa, after transitory speed reaches set-point value.

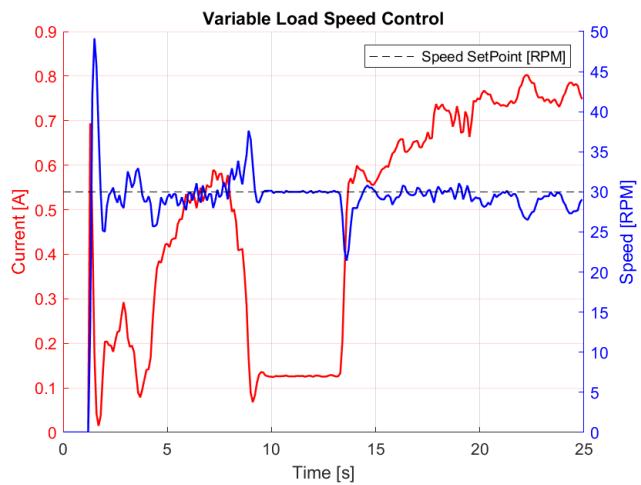


Figure 4.10: Speed control when is applied a variable torque.



#### 4.3.1 Firmware development

To control the speed of the motor a proportional controller with  $K_p = 0.05$  has been developed. The function *P\_controller* shown in Fig. 4.11 is implemented in order to calculate proportional term, given the angular velocity. This term in the function *Actuate\_Control\_Action*, is used to calculate the required voltage applied to the DC motor, using a driver controlled by a PWM signal. This function is shown in Fig. 4.12.

```
float P_controller(float speed)
{
    /* P Controller */
    float proportional_error = set_point - speed;
    float P_term = Kp * proportional_error;
    return(P_term);
}
```

Figure 4.11: *P\_controller* function.

```
void actuate_control_action(float P_term)
{
    /* Calculate duty cycle */
    v_out = v_out + P_term;
    float duty = v_out/Vm;
    /* Duty cycle saturation block */
    if (duty > 1)
    {
        duty = 1;
        v_out = Vm;
    }
    else if (duty < 0)
    {
        duty = 0;
        v_out = 0;
    }
    /* Set CCR in order to get the PWM signal */
    uint16_t ccr = (uint16_t)(fabs(duty)*(float)(1+HAL_TIM_GET_AUTORELOAD(&htim4)));
    HAL_TIM_SET_COMPARE(&htim4,TIM_CHANNEL_1,ccr);
}
```

Figure 4.12: *Actuate\_Control\_Action* function.

Note that DC motor work as low-pass filter, so that to avoid disturbances given by motor driver, PWM signal has been set at a frequency of 20 kHz which is the maximum working frequency of the driver.



## 5 Conclusions

The aim of this term project was to get in touch with some of the mainly used current sensors from a theoretical and, wherever possible, practical point of view.

The first current sensor analyzed has been the simplest one, the shunt resistor current sensor. After various tests, it was shown that the transduction coefficient is not constant when the sensor is used at the beginning of its full scale due to a voltage drop across the shunt resistor that does not match the minimum differential voltage the operational amplifier needs, according to the INA169 data-sheet.

The Hall effect current sensor has been used to get the static characteristic of a DC motor, process that needs a rotary incremental encoder to measure the angular velocity of the shaft, in addition to a current sensor. The DC motor used during the tests has a nominal current that is too far from the full scale of the current sensor, so external turns have been added in order to decrease its full scale. It has been noticed that when the current sensor is not connected to a closed circuit, its output is not null, presenting an offset transduced current that can be caused by the residual core magnetization that exist when the sensor is not used correctly. In order to eliminate this offset error, the sensor could be treated with a demagnetization process. Due to limited resources, the test-bed was not so appropriate but sufficient to reproduce a good static characteristic.

In addition, the Hall effect current sensor and the rotary incremental encoder has been used to monitor the variation of the DC motor load current in a speed control loop.

# 37D Metal Gearmotors



Pololu 37D Metal Gearmotors are powerful brushed DC motors paired with 37mm-diameter gearboxes. There are nine different gearbox options available, ranging from 6.3:1 to 150:1, and two different motor options: 12 V and 24 V. The 24 V versions offer approximately the same speed and torque at 24 V as their 12 V counterparts do at 12 V, with approximately half the current draw. This datasheet includes two sets of performance graphs for each version, one at its nominal voltage and one at half of its nominal voltage. Each version is available with an integrated 64 CPR quadrature encoder on the motor shaft.

Note: The original versions of these gearmotors had gearboxes with all spur gears. In August 2019, these were replaced by functionally identical “Helical Pinion” versions that feature helical gears for the first stage of the gearbox, which reduces noise and vibration and improves efficiency. The picture on the right shows the helical pinion gear and first mating gear.



## Performance summary and table of contents

Rated Voltage	Pololu Item #	Gear Ratio	No Load		At Maximum Efficiency				Max Power	Stall Extrapolation <sup>(2)</sup>		Graph Pages
			Speed	Current	Speed	Torque	Current	Output		Torque	Current	
			:1	RPM	A	RPM	kg·mm	A	W	W	kg·mm	A
12 V	4750 <sup>(1)</sup>	1	10,000	0.2						5		5.5
	4747, 4757	6.25	1600		1300	4.9	1.2	6.4	12	30		
	4748, 4758	10	1000		850	6.6	0.91	5.7	12	49		
	4741, 4751	18.75	530		470	10	0.76	5.0	12	85		
	4742, 4752	30	330		280	18	0.78	5.1	12	140		
	4743, 4753	50	200		180	22	0.66	4.0	10	210		
	4744, 4754	70	150		130	32	0.68	4.2	10 <sup>(3)</sup>	270		
	4745, 4755	102.08	100		87	42	0.72	3.8	8 <sup>(3)</sup>	340		
	4746, 4756	131.25	76		66	60	0.74	4.1	6 <sup>(3)</sup>	450		
	2828, 2829	150	67		58	65	0.72	3.8	6 <sup>(3)</sup>	490		
24 V	4690 <sup>(1)</sup>	1	10,000	0.1						5.5		3.0
	4688, 4698	6.25	1600		1300	5.5	0.58	7.4	14	35		
	4689, 4699	10	1000		850	7.5	0.49	6.6	14	55		
	4681, 4691	18.75	530		450	13	0.49	6.1	13	95		
	4682, 4692	30	330		280	19	0.46	5.5	13	150		
	4683, 4693	50	200		170	27	0.41	4.9	12	230		
	4684, 4694	70	140		120	39	0.42	5.0	10 <sup>(3)</sup>	310		
	4685, 4695	102.08	100		86	51	0.42	4.5	8 <sup>(3)</sup>	390		
	4686, 4696	131.25	79		68	63	0.40	4.4	6 <sup>(3)</sup>	470		
	4687, 4697	150	68		59	73	0.41	4.4	6 <sup>(3)</sup>	560		

### Notes:

- (1) Max efficiency data and performance graphs currently unavailable for the motors without gearboxes (items #4750 and #4690).
- (2) Listed stall torques and currents are theoretical extrapolations; units will typically stall well before these points as the motors heat up. Stalling or overloading gearmotors can greatly decrease their lifetimes and even result in immediate damage. The recommended upper limit for continuously applied loads is 100 kg·mm, and the recommended upper limit for instantaneous torque is 250 kg·mm. Stalls can also result in rapid (potentially on the order of seconds) thermal damage to the motor windings and brushes; a general recommendation for brushed DC motor operation is 25% or less of the stall current.
- (3) Output power for these units is constrained by gearbox load limits; spec provided is output power at max recommended load of 100 kg·mm.

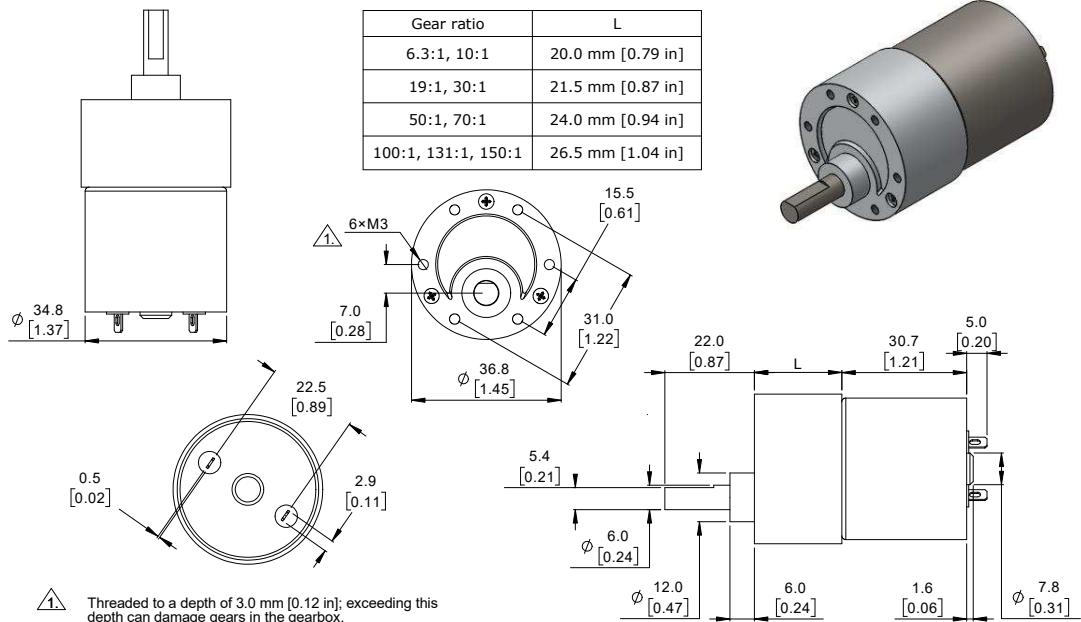
# 37D Metal Gearmotors



## Dimensions (units: mm over [inches])

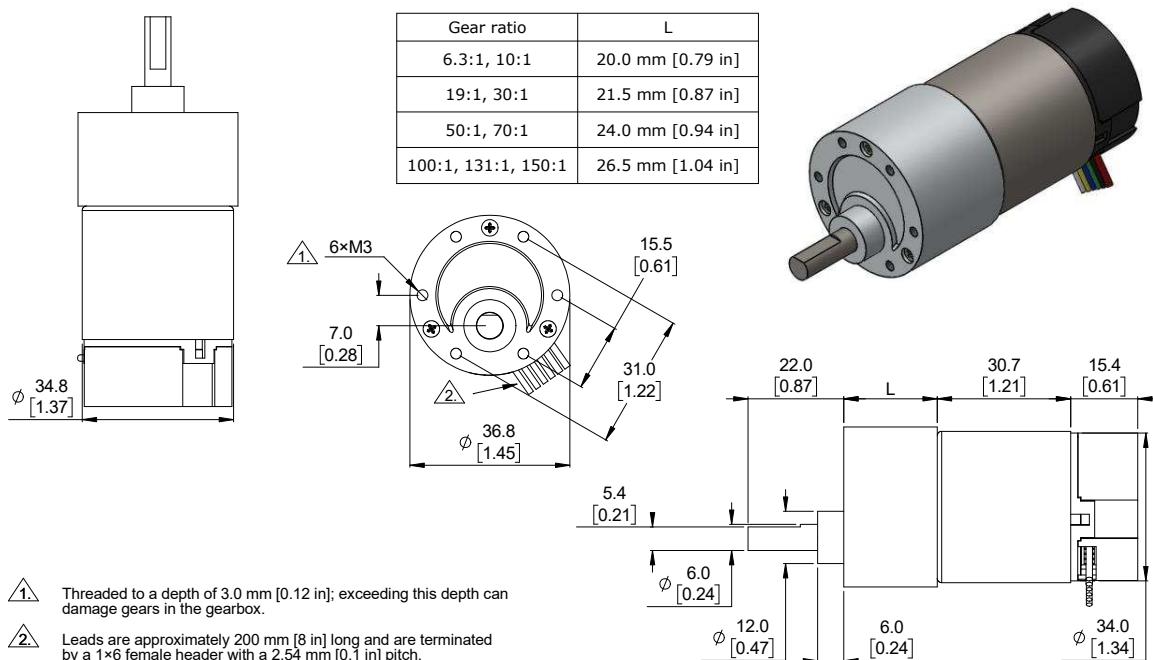
Gearmotor versions without encoders (items #2829, 4681–4689, 4741–4748)

weight: 175 g to 195 g



Gearmotor versions with encoders (items #2828, 4691–4699, 4751–4758)

weight: 190 g to 210 g

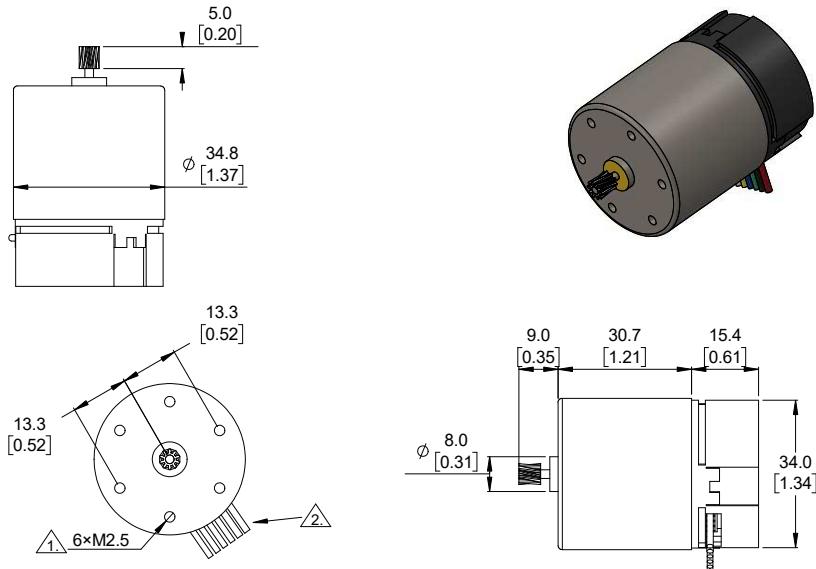


# 37D Metal Gearmotors



**Motor with encoder and no gearbox** (items #4690, 4750)

weight: 110 g



1 Threaded to a depth of 3.5 mm [0.14 in]; exceeding this depth can damage the motor.

2 Leads are approximately 200 mm [8 in] long and are terminated by a 1×6 female header with a 2.54 mm [0.1 in] pitch.

## Using the encoder

Versions with encoders have additional electronics mounted on the rear of the motor. Two Hall-effect sensors are used to sense the rotation of a magnetic disc on a rear protrusion of the motor shaft. The encoder electronics and magnetic disc are enclosed by a removable plastic end cap. The following pictures show what the encoder portion looks like with the end cap removed:



The quadrature encoder provides a resolution of 64 counts per revolution (CPR) of the motor shaft when counting both edges of both channels. To compute the counts per revolution of the gearbox output, multiply the gear ratio by 64.

The motor/encoder has six color-coded, 20 cm (8") leads terminated by a 1×6 female connector with a 2.54 mm (0.1") pitch. This connector works with standard 0.1" male breakaway headers and Pololu male premium jumper and precrimped wires. If this header is not convenient, the crimped wires can be pulled out of the 1×6 housing and used with different crimp connector housings instead (e.g. 1×2 for the motor power and 1×1 housings for the other four leads), or the connectors can be cut off entirely.

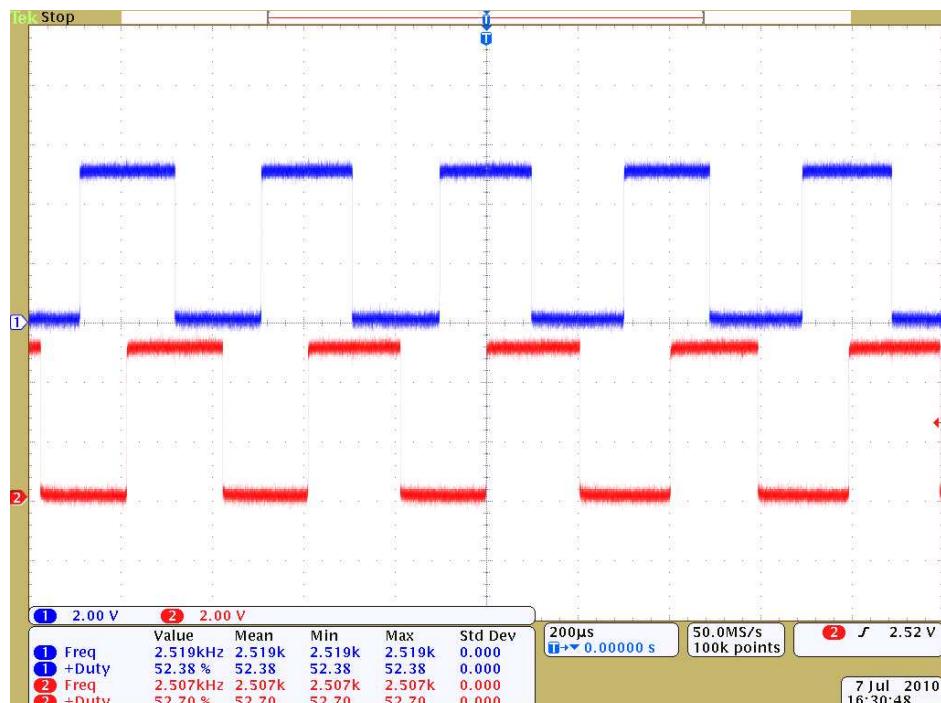
# 37D Metal Gearmotors



Lead Color	Function
Red	Motor power
Black	Motor power
Green	Encoder ground
Blue	Encoder Vcc (3.5 V to 20 V)
Yellow	Encoder A output
White	Encoder B output

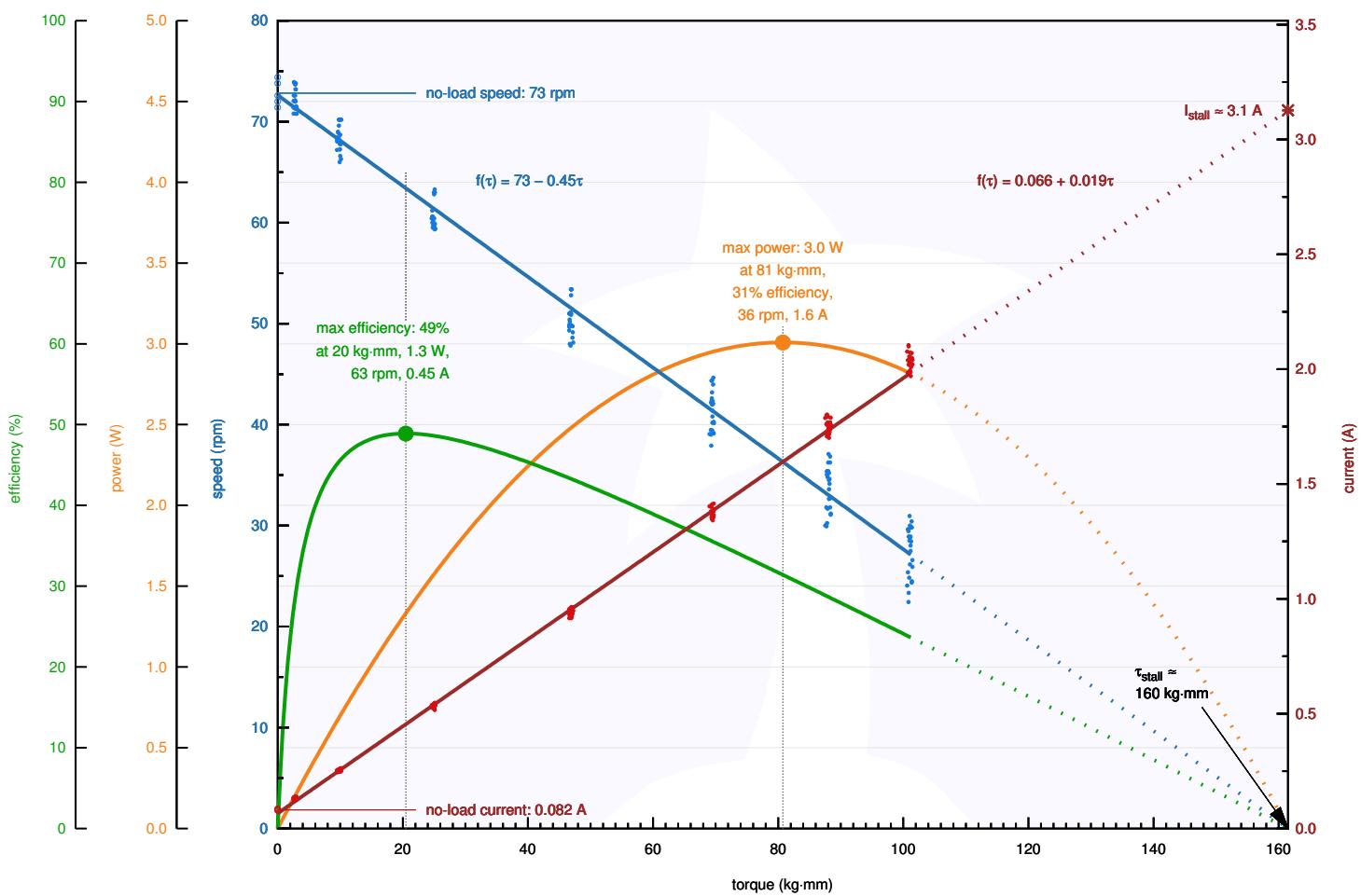


The Hall sensors require an input voltage, Vcc, between 3.5 V and 20 V and draw a maximum of 10 mA. The A and B outputs are square waves from 0 V to Vcc approximately 90° out of phase. The speed of the motor can be determined from the frequency, and the direction of rotation can be determined from the order of the transitions. The following oscilloscope capture shows the A and B (yellow and white) encoder outputs using a 12 V motor at 12 V and a Hall sensor Vcc of 5 V:



Counting both the rising and falling edges of both the A and B outputs results in 64 counts per revolution of the motor shaft. Using just a single edge of one channel results in 16 counts per revolution of the motor shaft, so the frequency of the A output in the above oscilloscope capture is 16 times the motor rotation frequency.

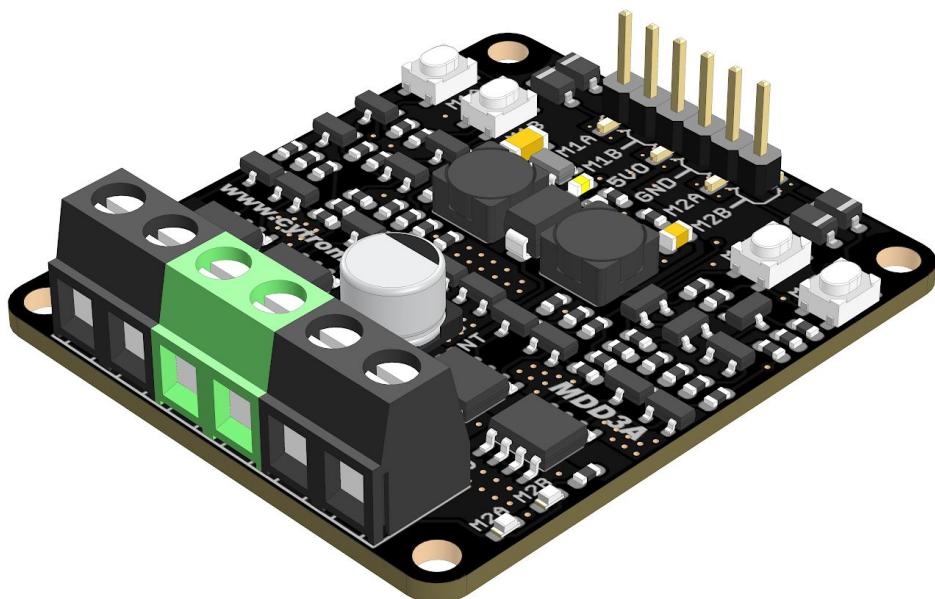
### Pololu Items #4744, #4754 (70:1 Metal Gearmotor 37D 12V) Performance at 6 V





# MDD3A

## 3Amp 4V-16V DC Motor Driver (2 Channels)



## Datasheet

Rev 1.0  
March 2019

Information in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. No representation or warranty is given and no liability is assumed by Cytron Technologies Incorporated with respect to the accuracy or use of such information or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Cytron Technologies's products as critical components in life support system is not authorized except with express written approval by Cytron Technologies. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.

## 1. BOARD LAYOUT & FUNCTION

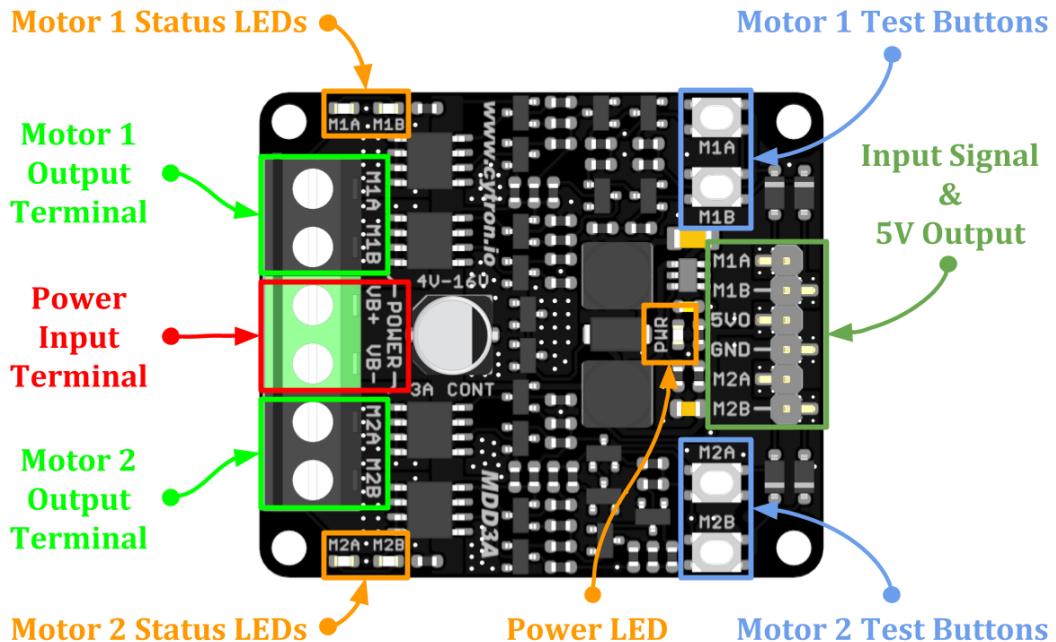


Figure 1: MDD3A Board Functions

Function	Description
<b>Power Input Terminal</b>	Connect to battery. <ul style="list-style-type: none"> <li>• VB+ : Positive</li> <li>• VB- : Negative</li> </ul>
<b>Motor Output Terminal</b>	Connect to motor terminal. Motor direction depends on the polarity.
<b>Power LED</b>	Turn on when power up.
<b>Motor Status LEDs</b>	Turn on when the motor is running. <ul style="list-style-type: none"> <li>• M1A / M2A : Forward*</li> <li>• M1B / M2B : Backward*</li> </ul>
<b>Test Buttons</b>	Press to test the functionality of the motor driver. Motor will run at full speed. <ul style="list-style-type: none"> <li>• M1A / M2A : Forward*</li> <li>• M1B / M2B : Backward*</li> </ul>
<b>Input Signal &amp; 5V Output</b>	Input signal from microcontroller to control the motor. +5V output can be used to power the microcontroller. <ul style="list-style-type: none"> <li>• M1A : PWM Input A for motor 1.</li> <li>• M1B : PWM Input B for motor 1.</li> <li>• 5VO : DC +5V Output (Maximum 200mA)</li> <li>• GND : Ground</li> <li>• M2A : PWM Input A for motor 2.</li> <li>• M2B : PWM Input B for motor 2.</li> </ul>

Table 1: MDD3A Board Functions

\* Actual motor direction is depending on the motor connection.

Swapping the connection (MA & MB) will reverse the direction.

## 2. SPECIFICATIONS

No	Parameters	Min	Max	Unit
1	Power Input Voltage (Vin)	4	16	VDC
2	Maximum Motor Current	Continuous	-	3 A
		Peak (< 5 seconds)	-	5 A
3	Logic Input Voltage (M1A, M1B, M2A, M2B)	Low Level	0	0.5 V
		High Level	1.7	12 V
4	PWM Frequency <i>(Output frequency is same as input frequency)</i>	DC	20	KHz
5	DC +5V Output Maximum Current	-	200	mA

Table 2: MDD3A Absolute Maximum Ratings

## 3. DIMENSION

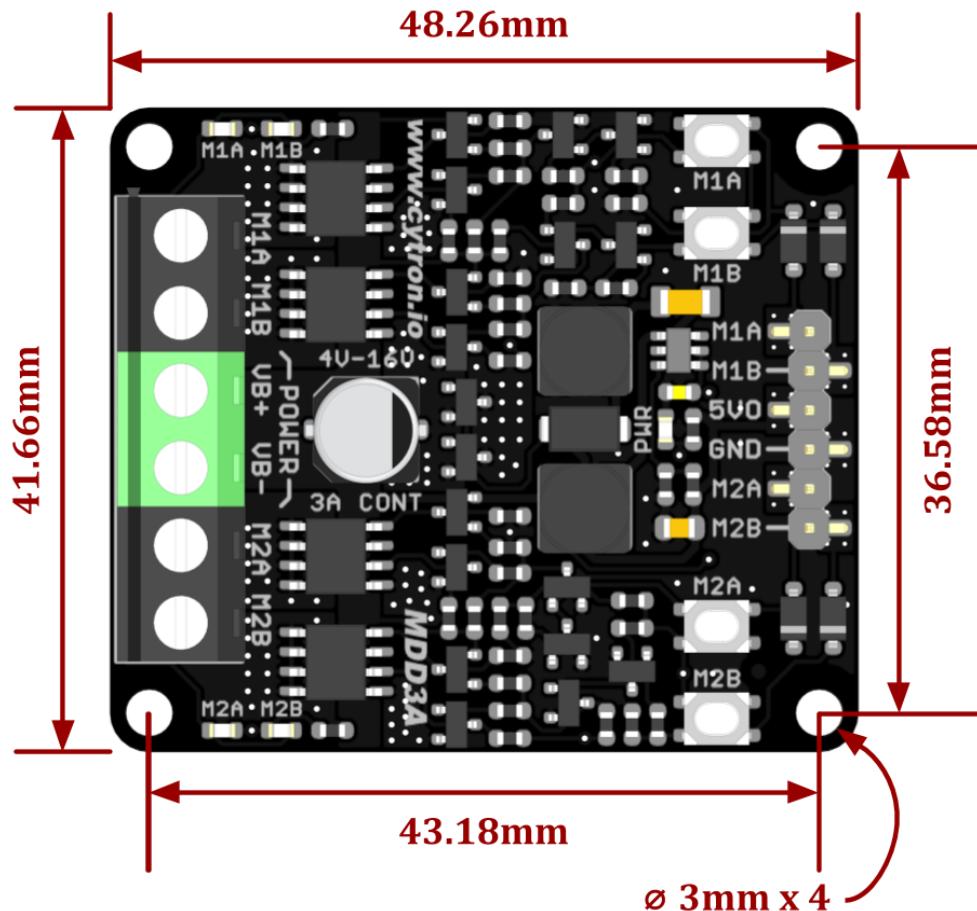


Figure 2: MDD3A Dimension

## 4. INTERFACE

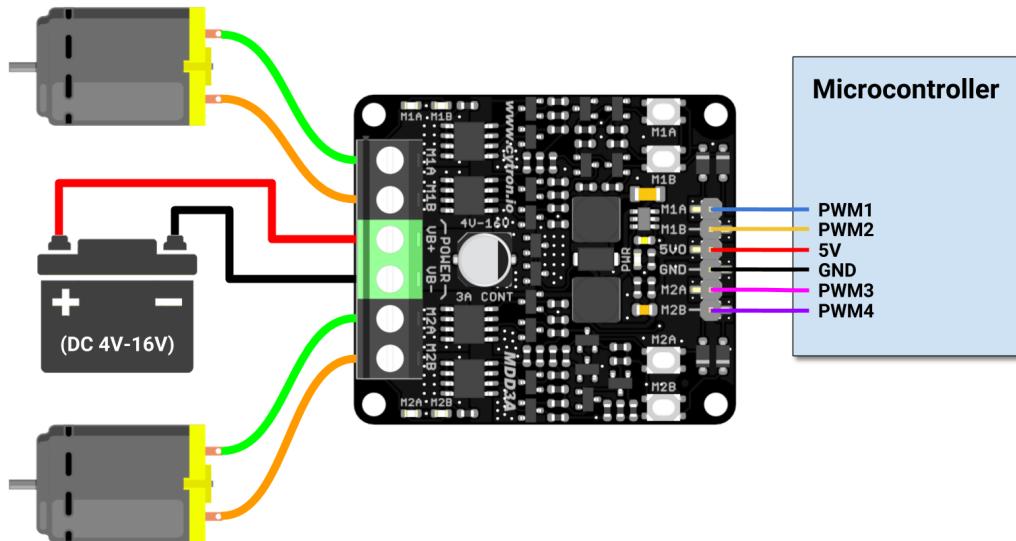


Figure 3: Connection Diagram for Brushed DC Motor

Input A (M1A / M2A)	Input B (M1B / M2B)	Output A (M1A / M2A)	Output B (M1B / M2B)	Motor
Low	Low	Low	Low	Brake
High	Low	High	Low	Forward*
Low	High	Low	High	Backward*
High	High	High	High	Brake

Table 3: Input Truth Table

\* Actual motor direction is depending on the motor connection.  
Swapping the connection (MA & MB) will reverse the direction.

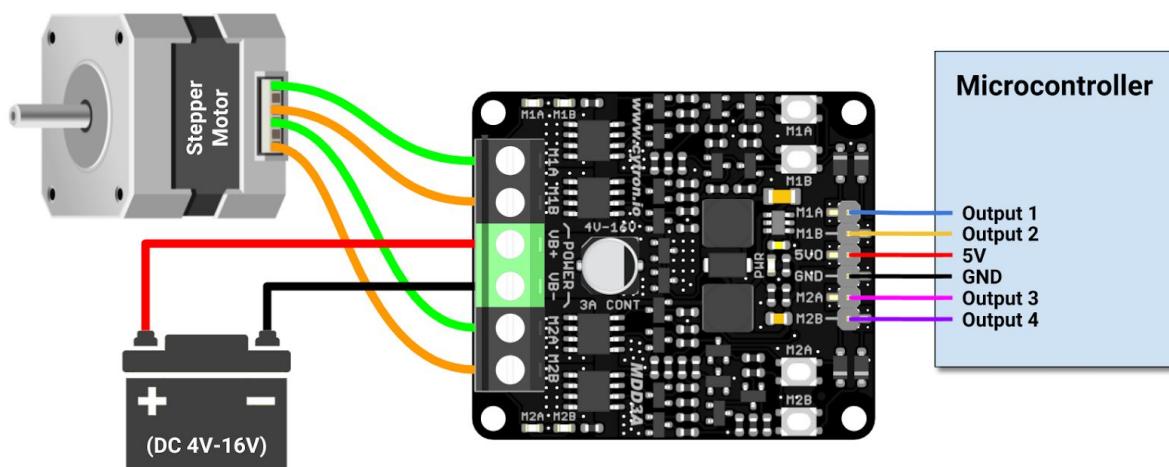


Figure 4: Connection Diagram for Stepper Motor



# LEM CURRENT TRANSDUCERS

Module LTA 50P/SP1  
(Instantaneous Outputs).

107-803.

## Definition - Principle

The LEM type LTA50P/SP1 is a transducer employing the Hall Effect to measure D.C. and complex waveform A.C. currents in a non invasive manner. Galvanic isolation is provided between the primary (measured) and the analogue output (control) signal.

A feature of the LTA100/SP1 is the direct provision of either a current output or voltage output.

## Electrical data

Nominal current $I_N$	: 50 Amps rms.
Output modes	
(1) Current Output	: 1mA/Amp.
Measuring Range	: 0 to $\pm 160A$ . (Supply voltage $\pm 15V$ ; $R_{meas} = 50\text{ Ohms}$ ).
Overall Accuracy at 25°C	: $\pm 0.5\%$ of $I_N$
(2) Voltage Output	: 100mV/Amp.
Measuring Range	: 0 to $\pm 50A$ .
Overall Accuracy at 25°C	: $\pm 1.0\%$ of $I_N$
Turns Ratio	: 1:1000.
Supply Voltage	: + and - 15V ( $\pm 5\%$ ).
Dielectric Strength	: 3 kV rms/50Hz/1 min.

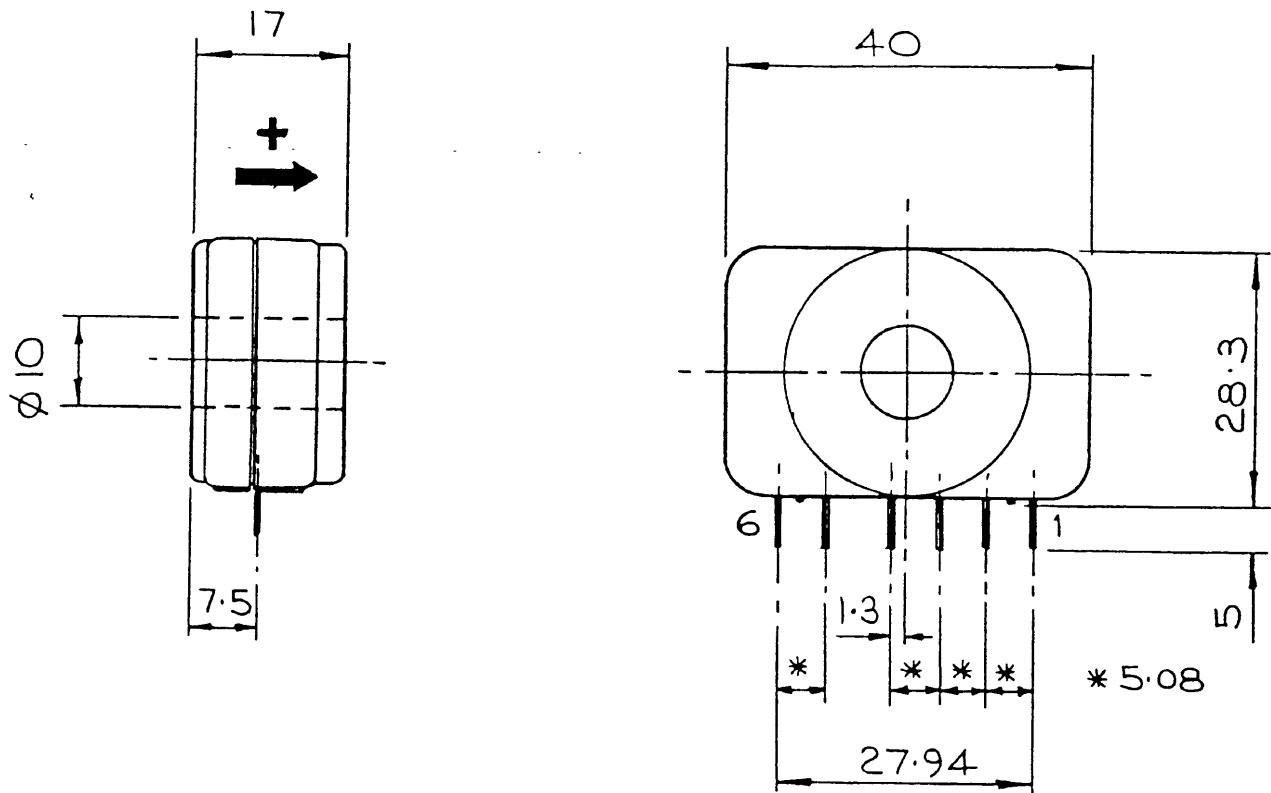
## Dynamic Performance

Zero Drift (between 0°C and 70°C)	: Max $\pm 0.6\text{mA}$
Linearity	: $\pm 0.1\%$ of $I_N$
Response Time	: < 1 microsecond.
di/dt Accurately Followed	: > 50 Amps per microsecond.
Frequency Range	: DC to 100kHz.

## General Data

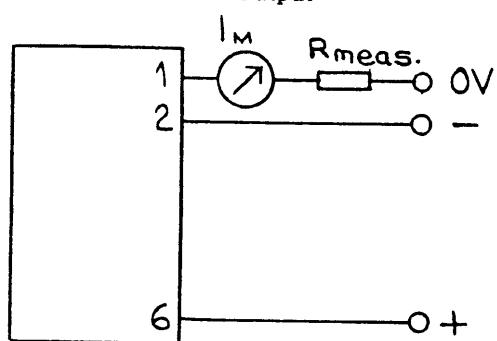
Operating Temperature	: 0°C to + 70°C.
Storage Temperature	: -25°C to +85°C.
Current Drain	: 15mA + Im (measuring current).
Internal Resistance (for current output)	: 25 Ohms.
Connections	: On 6 pins 0.63 x 0.56mm.
Case Material	: Flame retardant Noryl Grade V0-150
Weight	: 15 grams.
Output Provisions	
Current Output	: On Pin 1.
Voltage Output	: On Pin 4. (Pins 1 and 4 must be linked).

**Dimensions** : LTA50P/SPI

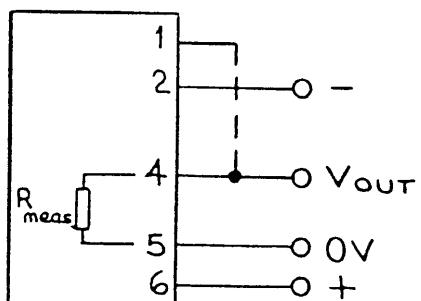


**Electrical Connections** : LTA50P/SPI

Current Output



Voltage Output



The Company policy is one of continual product improvement and the company reserves the right to revise the above specification without notice.



## References

- [1] Texas Instruments. “Texas Instruments INA139/INA169”. In: (2017).
- [2] LEM Components. “Isolated current and voltage transducers”. In: 3rd Edition ().
- [3] LEM Components. “High Precision Current Transducers”. In: () .



## List of Figures

2.1	Typical Application Circuit INA139-INA169.	5
2.2	PowerModule GM v1.0.	5
2.3	Schematic Power Module GM v1.0.	6
2.4	Instrumentation.	7
2.5	Building up the data acquisition system	7
2.6	Instrumentation set up.	8
2.7	Data Acquisition.	9
2.8	$K_a$ for each measurement	9
2.9	Linear Regression.	10
2.10	$K_v$ for each measurement.	10
2.11	Linear Regression	11
2.12	Hall effect electronic circuit [2].	12
2.13	LEM LTA/50p/SP1	13
2.14	How to calculate $k_T$ .	14
2.15	Test-bed.	15
2.16	Callback that computes the dynamic average.	15
2.17	Callback implemented to trasduce the average ADC Value in current and send it on UART.	16
2.18	Current measured changing the time window with a multiple of 20 ms.	16
2.19	Current measured changing the time window with a non-multiple of 20 ms.	17
2.20	Offset error over time considering and time window of 100 ms.	17
2.21	Fluxgate working principle.	18
2.22	Square wave voltage (a), Current created (b), Asymmetry of the created current (c)	19
2.23	Simplified base circuit for DC current compensation.	19
2.24	Solution against voltage peaks re-injection.	19
3.1	Hall Effect Sensor.	21
3.2	Pololu Rotary Encoder.	21
3.3	Output square waves, A and B.	22
3.4	<i>get_speed</i> function.	22
3.5	Function implemented to avoid overflow when the difference is computed.	23
4.1	Example of Static characteristic of DC Motor.	24
4.2	Test-bed.	25
4.3	Static characteristic @5 V obtained in this study work.	26
4.4	Static characteristic @6 V from data-sheet.	27
4.5	TIMER-BASE Callback.	28
4.6	MATLAB script for data acquisition.	28
4.7	MATLAB script in which data are processed.	29
4.8	Speed control when is not applied a load torque.	30
4.9	Speed control when is applied a constant torque.	30
4.10	Speed control when is applied a variable torque.	31
4.11	<i>P_controller</i> function.	32
4.12	<i>Actuate_Control_Action</i> function.	32