24th ABCM International Congress of Mechanical Engineering
December 3-8, 2017, Curitiba, PR, Brazil

COBEM-2017-2484
UTILIZATION OF AN ARDUINO® MICROCONTROLLER FOR DYNAMIC THRUST MEASUREMENT OF A POWERTRAIN ASSEMBLY USED IN A SAE AERO DESIGN AIRCRAFT

Gutembergy Ferreira Diniz
gutembergy_ferreira@hotmail.com

Prince Azsembergh Nogueira de Carvalho
prince.cfv@gmail.com

Marciel Bezerra de Moura
marcielllbezerra@hotmail.com

Giovanna Maria de Oliveira Fin
giovanna_fin@yahoo.com.br

Pedro Bruno Nogueira da Silva
nogueira372@gmail.com

Mateus Anderson Barreto Duarte
nogueira372@gmail.com

Zoroastro Torres Vilar
Universidade Federal Rural do Semi-Árido
zoroastro@ufersa.edu.br

Abstract. *The amount of thrust a powertrain assembly provides to the moving aircraft varies with speed, and it is very important for the design of aircraft like those used in Society of Automotive Engineers (SAE) aero design competitions. Many teams conduct their tests using wind tunnels, which is the most specific but less accessible method. These can also be carried out more quickly and cheaply with the help of a pickup truck, where the running train is subjected to a flow of air from the movement of the vehicle, bringing the test closer to the wind tunnel conditions. In order to increase the accuracy of the test performed in a pickup vehicle, an Arduino® microcontroller in conjunction with a load cell was implemented. The workbench equipped with the microcontroller and load cell obtained results different from the literature, but more reliable to the project, since the model tested is the one that most closely approximates the real model used in aeronautical aircraft, improving its reliability and accuracy, as well as making it an alternative to obtain dynamic thrust in powertrains*

Keywords: *SAE aero design, dynamic thrust, Arduino® microcontroller*

1. INTRODUCTION

The SAE Brazil AeroDesign competition is a challenge launched to engineering students whose objective is to promote diffusion, exchange of techniques, and knowledge of aeronautical engineering. This competition takes place since 1999, conceived annually by SAE BRASIL.

Competitor teams must design, construct and test an aircraft in a radio-controlled scale that meets the requirements of a regulation, variable annually imposed by SAE. The challenges imposed on teams through regulation range from the amount and type of cargo loaded to the constraints imposed directly on the aircraft geometry.

In general, teams are divided into specific subareas of the project to better and faster progress of the stages. Among these, the performance subarea is responsible for the proper selection of the propulsion motor group of the aircraft, which basically consists of the motor and propeller that will be used.

The powertrain is responsible for providing a traction force, called thrust or avaliable thrust, which gives movement to the aircraft. In an aircraft design, this force that is supplied to the system is trivial because it is related to the amount of mass that the aircraft can carry.

The available thrust of a powertrain assembly depends on the type of engine (internal combustion or reaction), propeller (or blades) being used, and the speed of incidence of airflow in the moving propeller. According to Durand and Lesley (1920), the theoretical static thrust ($T_{v=0}$) is given by a software called "propeller selector" or your mathematical model that relates the available power in the motor shaft (P_E), rotation (n), diameter (D) e pitch (p) of the propeller, as shown by the Eq. (1) and (2):

$$T_{v=0} = K_{T0} . \frac{P_E}{n.D} \tag{1}$$

$$K_{T0} = 57000.\left(1{,}97 - \frac{p}{D}\right) \tag{2}$$

Results of dynamic tests carried out in a wind tunnel shown by [4], show that the thrust provided by the powerplant, when there is incidence of air on the propeller (the same as with the aircraft in flight), decreases with increasing speed. T The behavior of thrust as a function of velocity is an important part of the SAE aero design project, since the amount of load to be loaded as well as the runway length for takeoff, which are dependent data of this function, are competition requirements. This paper shows an alternative way to find the buoyancy function with the aircraft speed. Instead of using a wind tunnel, a workbench is fixed in a pick-up vehicle (Figure 1), equipped with an Arduino® microcontroller (an open-source electronics platform based on easy-to-use hardware and software) and a load cell was set in motion to simulate a wind tunnel with better cost and accessibility.

2. EXPERIMENTAL PROCEDURE

In this paper, static tensions were measured, with the vehicle at rest, and dynamic thrust for four different speeds of the vehicle (20 km/h, 40 km/h, 60 km/h and 80 km/h). The dynamic tests were performed using a test bench equipped with the Arduino® microcontroller and 5kg load cell with the structure mounted on a pick-up vehicle (Fig. 1).

Figure 1. Experimental apparatus

The powertrain used consisted of an ASP .61 SII glow engine (Fig. 2) and propellers with more usual diameters (Fig. 3). In the static test, the rotation was measured in order to verify the accuracy of equations 1 and 2, using a conventional laser tachometer (Fig. 4).

Figure 2. ASP .61 glow engine

Figure 3. Propellers

Figure 4. Minipa Digital Tachometer

The fixed bed in the apparatus used was designed to generate relative movement between the powertrain in operation and the fixed load cell connected to a data acquisition system (Fig. 5).

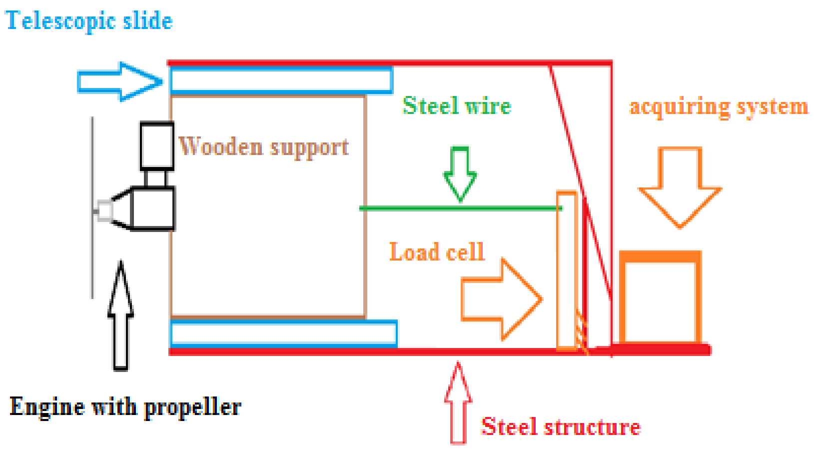

Figure 5. Scheme of the propulsion system and acquisition

The powertrain unit, fixed to the load cell through a rigid steel wire, transmits a displacement that is converted into force and measured by the acquisition system. The acquisition system (Fig. 6) consists of an Arduino® platform powered by a 9V battery with the addition of a 2Gb FAT 32 SD memory card for storing the data, an LED lamp, to indicate when the SD card is in fact storing the data, a 1000 Ω resistor connected to the lamp to prevent damage, the SD modules and an HX711 signal amplifier, which is connected to the load cell to collect the measured data with greater accuracy.

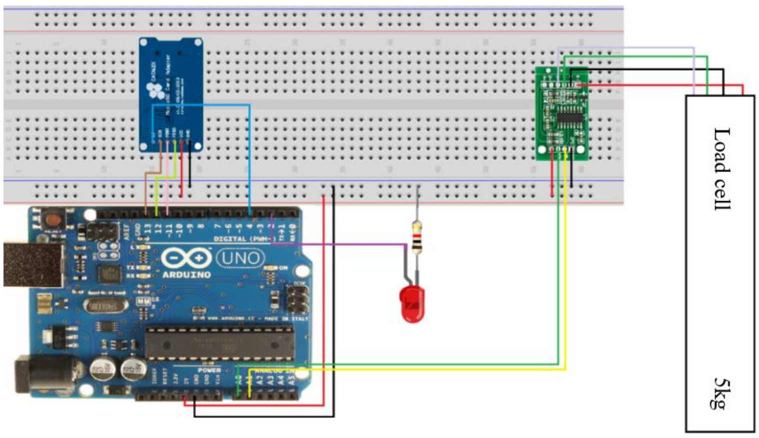

Figure 6. Sizing circuit used in the test.

The operation of the data acquisition system was initially made possible by the use of codes that carry out storage on the SD card (Fig. 7) and the calibration of the load cell.

```
#include <SPI.h>
#include <SD.h>
File myFile;
void setup()
{Serial.begin(9600);
  while (!Serial) {;}
  Serial.print("Initializing SD card...");
  if (!SD.begin(4)) {Serial.println("initialization failed!");
    return; }
 Serial.println("initialization done.");
 myFile = SD.open("test.txt", FILE_WRITE);
  if (myFile) { Serial.print("Writing to test.txt...");
    myFile.println("testing 1, 2, 3.");
    myFile.close();
    Serial.println("done.");
  } else { Serial.println("error opening test.txt");}
  myFile = SD.open("test.txt");
  if (myFile) {
    Serial.println("test.txt:");
    while (myFile.available()) {
      Serial.write(myFile.read());  }
    myFile.close();
  } else {Serial.println("error opening test.txt");
  }
}
void loop()
{
}
```

Figure 7. Arduino library standard SD card code.

Calibration of the load cell was necessary because the initial data did not show values consistent with the actual values. For the calibration, a known mass (Fig. 8 and 9) was placed on the cell, the measured value was then divided by the actual mass value, this scale was added to the microcontroller code (Fig. 10) and thus to make the system understand the relationship between actual and measured mass values. The selected ports for the load cell were the analog ports A0 and A1, where the clock pin was placed on port A0 and the data pin on port A1.

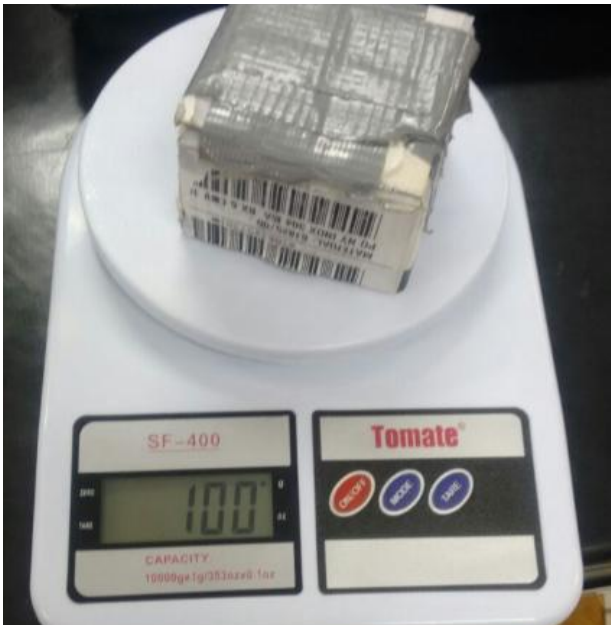

Figure 8. Precision balance measuring a standard weight for calibration.

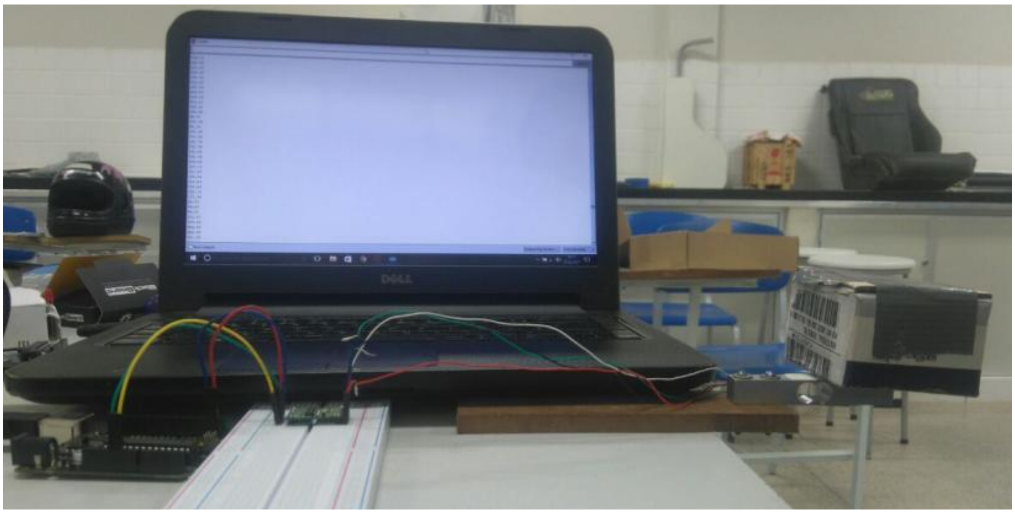

Figure 9. Load cell measuring a standard weight of 100 grams.

```
#include "HX711.h"
#define DT A1
#define SCK A0
HX711 cell(DT, SCK);
float T = 0;
void setup() {
  Serial.begin(9600);
}
void loop() {
  cell.set_scale(417555.981); //Escala de calibração
  Serial.print(cell.get_units(2),3);
  Serial.print("\n");
  Serial.print(T);
  Serial.println(" ");
  T = (T + 0.1);
}
```

Figure 10. Code used to set measured value in cell.

Calibration of the load cell showed an error of approximately 5 grams, considered acceptable. After calibrating the cell, the micro SD code with the load cell code was pooled for data collection and storage. A time variable T, with resolution of 0.1 seconds, was added to obtain an acquisition frequency of 10 Hz.

```
#include "HX711.h"
#include <SPI.h>
#include <SD.h>
#define DT A1
#define SCK A0
HX711 cell(DT, SCK);
File file;
float T = 0;
void setup() {
  pinMode(2, OUTPUT);
  Serial.begin(9600);
  while (!Serial) {
    ;
  }
  Serial.print("Inicializando...");
  if (!SD.begin(4)) {
    Serial.println("Falha na inicialização!");
    return;
  }
  Serial.println("Inicialização feita.");
  file = SD.open("test.txt", FILE_WRITE);
  file.print("Peso(kg)");
  file.print("\t");
  file.print("Tempo(s)");
  file.print("\t");
  file.println(" ");
  file.close();
  file = SD.open("test.txt");
  if (file) {
    Serial.println("test.txt:");
    while (file.available()) {
      Serial.write(file.read());
    }
    file.close();
  } else {
    Serial.println("erro abrindo test.txt");
  }
}
void loop() {
  file = SD.open("test.txt", FILE_WRITE);
  cell.set_scale(417555.981);
  Serial.println(cell.get_units(2), 3);
  if (file) {
    Serial.println("Escrevendo no cartão...");
    file.print(cell.get_units(2), 3);
    file.print("\t");
    file.print(T);
    file.println(" ");
    file.close();
    Serial.println("Feito.");
    digitalWrite(2, HIGH);
  } else {
    Serial.println("erro");
    digitalWrite(2, LOW);
  }
  T = (T + 0.1);
}
```

Figure 11. Code used in the engine test.

The acquisition system was mounted on a soft wooden support to avoid knocks and minimize vibration (Fig. 12), and only then connected the test bench (Fig. 13).

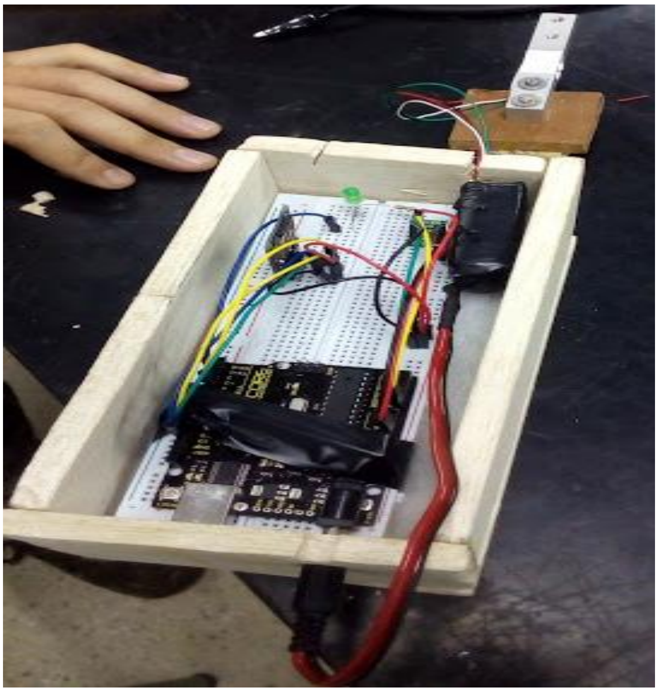

Figure 12. Bracket with mounted circuit

Figure 13. Test bench ready for measurement

3. RESULTS AND DISCUSSION

Equations (1) and (2), and the software propeller selector were used and compared with experimental results of static thrust and in contrast to what is stated Rodrigues (2010), the results obtained by them are not reliable. The comparison these results is show in Tab. 1.

Table 1. Comparison of static thrust

Propeller	K_{T0}	*R.P.M*	*Power (hp)*	*T. Analytic (kgf)*	*T. Prop. Select. (kgf)*	*T. Experim. (kgf)*	*Analytic Error (%)*	*Prop. Sel. Error (%)*
JC SuperProps 13x4	94751	12148	1,02	3,345	3,736	3,671	9,7	1,7
Aero star 12x4	93290	12795	0,87	2,870	3,152	3,515	22,5	11,5
Aero star 12x6	83790	12130	1,16	3,642	3,772	3,590	1,4	4,8
APC 12x6 Sport	83790	11697	1,04	3,390	3,395	3,899	15,0	14,9
APC 12x4 Sport	93290	14145	1,17	3,507	3,853	3,824	9,0	0,7
APC 13x4 Wide	94751	11840	0,95	3,178	3,549	4,455	40,2	25,5
APC 13x4 Sport	94751	11910	0,96	3,213	3,591	4,510	40,4	25,6

The experimental results of dynamic thrust were tested as follows. With the thrust points for each velocity obtained in the test (Tab. 2), a graph was plotted (Fig. 14). An adjustment of curve to obtain the equation, since this is what is important in the analysis of performance and performance of an aircraft.

Table 2. Statistical data of the test

Propeller	*Thrust (speed in m/s)*				
	T (0)	*T (5,55)*	*T (11,11)*	*T (16,66)*	*T (22,22)*
JC SuperProps 13x4	3,671	3,337	2,996	2,570	2,115
Aero star 12x4	3,515	3,178	2,835	2,411	1,961
Aero star 12x6	3,590	3,381	3,167	2,862	2,515
APC 12x6 Sport	3,899	3,650	3,396	3,045	2,650
APC 12x4 Sport	3,824	3,381	2,929	2,395	1,842
APC 13x4 Wide	4,455	3,973	3,481	2,891	2,275
APC 13x4 Sport	4,584	4,068	3,541	2,915	2,264

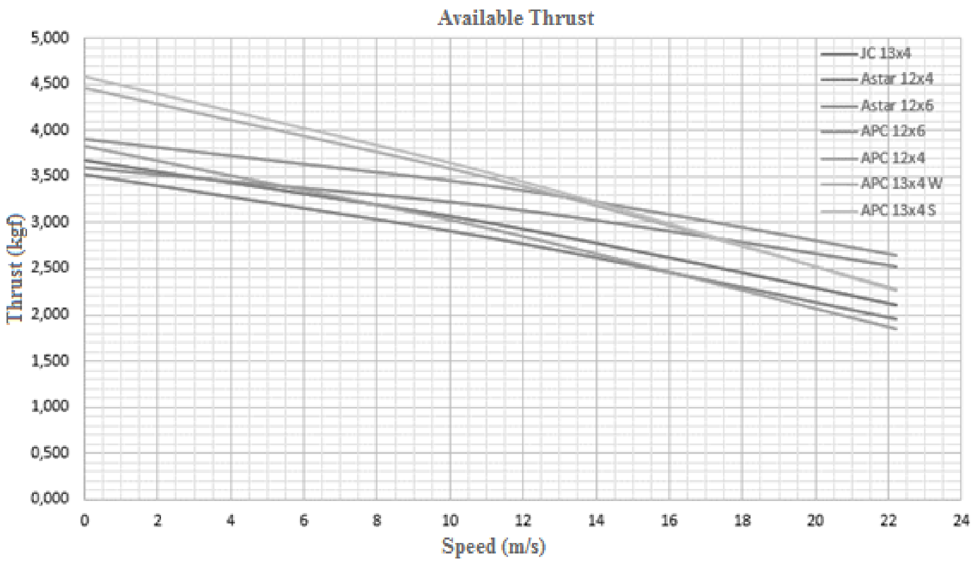

Figure 14. A thrust curve adjustment available.

A curve of the propeller 13x4 experimental available thrust was obtained (Eq. 3) in order to compare the difference of performance (Tab. 3) in a real scale plane using the analytic (Eq. 4), software equation and experimental curve.

$$T(v)_{\exp} = -0{,}0939.v + 4{,}584 \tag{3}$$

$$T(v)_{\exp} = -0{,}0537.v + 3{,}633 \tag{4}$$

$$T(v)_{\exp} = -0{,}0564.v + 3{,}522 \tag{5}$$

Table 3. Performance comparing

Situtation	*Payload carried (kg)*	*Error (%)*
Real Airplane	8,2	-
Simulated Performance with exp curve	8,4	2
Simulated Performance with anlt curve	7,3	12
Simulated Performance with software's curve	7,2	14

4. CONCLUSIONS

The work was satisfactory, since it achieved a new methodology of analysis and comparison of thrust of ASP .61 glow engine, being able to be easily applied to similar aero design engines and propellers, with good accuracy, precision and speed, providing an economically viable alternative to those that do not have a wind tunnel. It is intended for future work to analyze the behavior of parameters such as temperature and maximum yield in relation to the flow velocity.

5. REFERENCES

Durand, W. F., Lesley, E. P., *Experimental research on air propellers II*, T. R. nº30, NACA. 1920.

Lesley, E. P. *Propeller tests to determine the effect of number of blades at two typical solidities*. T. N. nº698, NACA. 1939.

Rodrigues, L. E. M. J. *Fundamentos da Engenharia Aeronáutica: Aplicações ao Projeto SAE-AeroDesign*. 1.ed. rev.- Instituto Federal de Educação, Ciência e Tecnologia de São Paulo. São Paulo: Edição do Autor, 2010.

Rosa, Edison da. *Introdução ao Projeto Aeronáutico: uma contribuição à Competição SAE AeroDesign*. 1ª ed. Universidade Federal de Santa Catarina. Florianópolis: Editora tribo da ilha, 2006.

SAE Brasil Aerodesign, <http://portal.saebrasil.org.br/programas-estudantis/sae-brasil-aerodesign>. Acessado em 20 de setembro de 2017.

6. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.