10th INTERNATIONAL SCIENTIFIC CONFERENCE
ON DEFENSIVE TECHNOLOGIES
OTEH 2022
Belgrade, Serbia, 13 – 14 October 2022

UAV ENGINE SPEED CONTROLLER FOR HOVER PERFORMANCE MEASUREMENT

LAZAR PETROVIĆ
Military Technical Institute, Belgrade, lazar.petrovic.node@gmail.com

MARIJA SAMARDŽIĆ
Military Technical Institute, Belgrade, marija.samardzic.vti@gmail.com

UROŠ IVKOVIĆ
Military Technical Institute, Belgrade, uske80@gmail.com

SNEŽANA ZUROVAC
Military Technical Institute, Belgrade, zurovac@medianis.net

Abstract: *This paper presents the characteristics of the interface for controlling the speed of each individual UAV engine using a computer, as well as the difficulties during its construction. The interface was designed and built in the Laboratory for Experimental Aerodynamics at the Military Technical Institute, Belgrade. The required engine speeds as well as its characteristics are set using a computer. Speed measurement is performed optically or by sampling the voltage on one phase of the motor. The ESC management protocol used is the 50 Hz PWM Standard. The communication protocol between the PC and the interface is RS485. The microcontroller used was PIC12F1572. The control program was written in the C programming language and a compiler from the microcontroller manufacturer was used.*

Keywords: *unmanned aircraft vehicle (UAV), electronic speed controller (ESC), standard PVM protocol, RS485, PIC microcontroller.*

1. INTRODUCTION

The control of the number of revolutions of each drone engine is controlled by a central controller that receives data from sensors (GPS, electronic compass, altimeter, camera...), remote control and the current state of the engine and battery, and on the basis of that data determines whether the number of engine revolutions should be increased or decreased, Figure 1 [1].

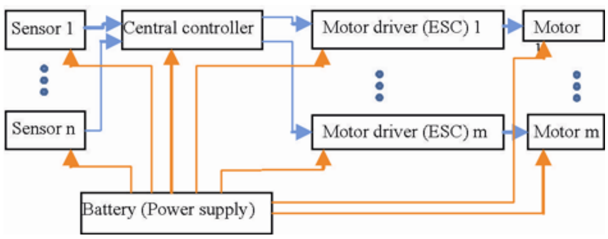

Figure 1. Block scheme UAV

Testing the characteristics of the drone implies a constant number of revolutions of the engine during the measurement. The number of revolutions of the engine is the result of the state of all sensors, so it is not possible to achieve a constant number of revolutions of the engine using the remote control. Simulating the data of all the sensors that affect the engine's behavior would be a more demanding task than creating a computer-controlled controller individually for each engine [2].

The diagram of computer-controlled drone motors is shown in Figure 2.

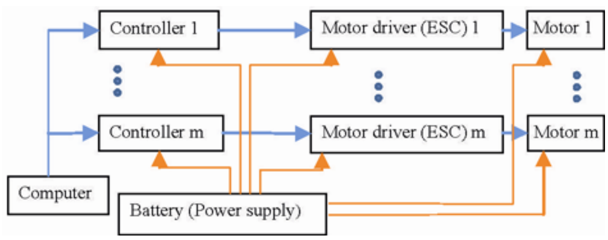

Figure 2. Computer-controlled drone motors

2. REQUIRED CHARACTERISTICS OF THE CONTROLLER

The controller should meet the following requirements:

– Use of standard engine speed control protocol

– Two-way communication between the controller and the computer

– Measuring the number of revolutions of the engine

– Supply voltage 6 – 24V

– Small dimensions of the controller so that it can be installed in the drone.

There are several protocols for controlling the speed of the drone's engine: Standard PWM, Oneshot125, Oneshot42, Multishot, Dshot, ProShot,.... The oldest and most universal is the PWM Standard. The disadvantage of this protocol is that the response of the drone engine is slower compared to other protocols. The difference in reaction time is measured in tens of milliseconds so it is not important for this test. The PWM period is 20 ms and can be changed if necessary, while the active part of the period lasts from 1 to 20 ms. The active part of the period less than 1 ms (with some ESC 2 ms) stops the motor, while for the maximum number of revolutions of the motor, the active part of the period is more than 2 ms. For the sake of simplicity and reliability of operation, the selected microcontroller (part of the controller) should have hardware PWM.

The communication between the computer and the controller should be reliable, resistant to induced interference, and the components needed to implement it (cables and electronic components) should be as small as possible. RS485 was chosen because only one pair is needed, it is possible to connect several devices, a minimum number of components is required, it is simple to implement both with the computer and with the controller, and it is resistant to interference caused by the operation of the motor. The communication speed is 9600 bps, which is sufficient for testing purposes. It is essential that the microcontroller has at least one serial communication port.

It is possible to perform three methods for measuring the number of revolutions: an optical sensor on the motor rotor, measuring the time between two rising edges on one phase of the motor and finding the dominant frequency of the signal on one phase of the motor.

The voltage supply of the microcontroller is in the range of 2 to 5V. The PWM output needs to be 3.3 or 5 V.

In order to reduce the number of components, the microcontroller is powered by a 5V stabilizer, so there is no need for a voltage translator for the ESC, whose PWM requires 5V. Drones powered by only one lithium cell are so small that it is not possible to place this controller inside the drone itself. It is necessary to make a suitable controller for such drones.

3. PROGRAM REQUIREMENTS

The functions to be performed by the controller are related to the number of required microcontroller pins. RS485 communication requires 3 pins (Rx, Tx and DE), one pin is used to measure the number of revolutions, one pin is PWM output and one pin is to control the operation mode (programming/normal operation). The minimum number of microcontroller pins is 6 + 2 for power (the programming pins are used as standard I/O pins during normal operation). The PIC12F1572 microcontroller was chosen as the most suitable [3].

Communication is done via the microcontroller's serial port (UART) which is connected to the RS232(TTL)/RS485 conversion IC, SN75176, which is always receiving except when it needs to send data [4]. The pins DE and RE of the circuit are controlled by the microcontroller. The microcontroller program should continuously receive characters in the FIFO register and measure the time between receiving two characters. If the time between receiving two characters is longer than the time required to send two characters, then the FIFO register is emptied. The character reception is in the interrupt routine. If the number of characters in the FIFO register is equal to the length of the command, the contents of the FIFO register are analyzed and the contents are acted upon.

PWM with initial parameters is activated on the pin intended for PWM output immediately after starting the program. Changes of the period and the active part of the cycle are performed by commands without interrupting the operation of the PWM.

4. COMMANDS

Controlling the number of revolutions of the engine is done by sending commands through the serial port of the computer. The commands implemented are:

– Setting the period for PWM
– Setting the active part of the PWM cycle
– Setting the number of motor poles
– Setting the engine speed
– Setting the number of motor speed samples for averaging.

The command syntax is:

– Command code (1 byte)
– Engine address (1 byte)
– Command argument (2 bytes)
– End of command (1 byte)

After receiving a valid command (with the appropriate address, code and syntax and argument value), the controller responds with a message with the following syntax:

– Command confirmation code (1 byte)
– Engine address (1 byte)
– Last measured engine speed (2 bytes)
– End of command confirmation (1 byte)

The first two commands control the signal sent to the ESC. The transmission characteristic of each individual ESC - motor pair is different, so setting the same parameters will not result in the same number of revolutions on the drone's motors. Also, the number of revolutions of the engine depends on the voltage of the battery, the load on the engine, the temperature of the ESC itself and other parameters, so it is necessary to create a program feedback loop in the computer that gives the commands, based on the current number of revolutions of the engine. The program feedback loop should be tolerant to errors in the measurement of the number of revolutions of the engine both in terms of value and time (missing samples) [5].

The command for setting the motor poles is a number by which the measured number of pulses on one phase of the motor is divided to obtain the actual number of revolutions of the motor. It is usual that the number of poles of the drone motor is seven, which is the initial value.

By setting the (target) number of engine revolutions, an internal program feedback loop is activated in the device itself so that the device changes the duration of the active part of the PWM cycle in order to achieve the set number of engine revolutions. This way of controlling the engine speed gives satisfactory results in the engine speed range of interest because the current control for each phase of the PWM motor is often more than an order of magnitude higher than the basic phase frequency. The band-pass filter at the input of the microcontroller for measuring the number of revolutions is adjusted to the expected range of frequencies, so that invalid measurements of the number of revolutions are significantly more frequent outside the expected range.

4. REALIZATION OF THE DEVICE

The approximate dimensions of the double-sided printed circuit board of the first version of the device are 50x26x 10 mm, which is enough to install the device in the drone housing for each motor separately. The schematic is shown in Figure 3, and the appearance of the assembled device is shown in Figure 4.

By using components that are commercially available and smaller in size, as well as multilayer PCB, it is possible to significantly reduce the dimensions of the device.

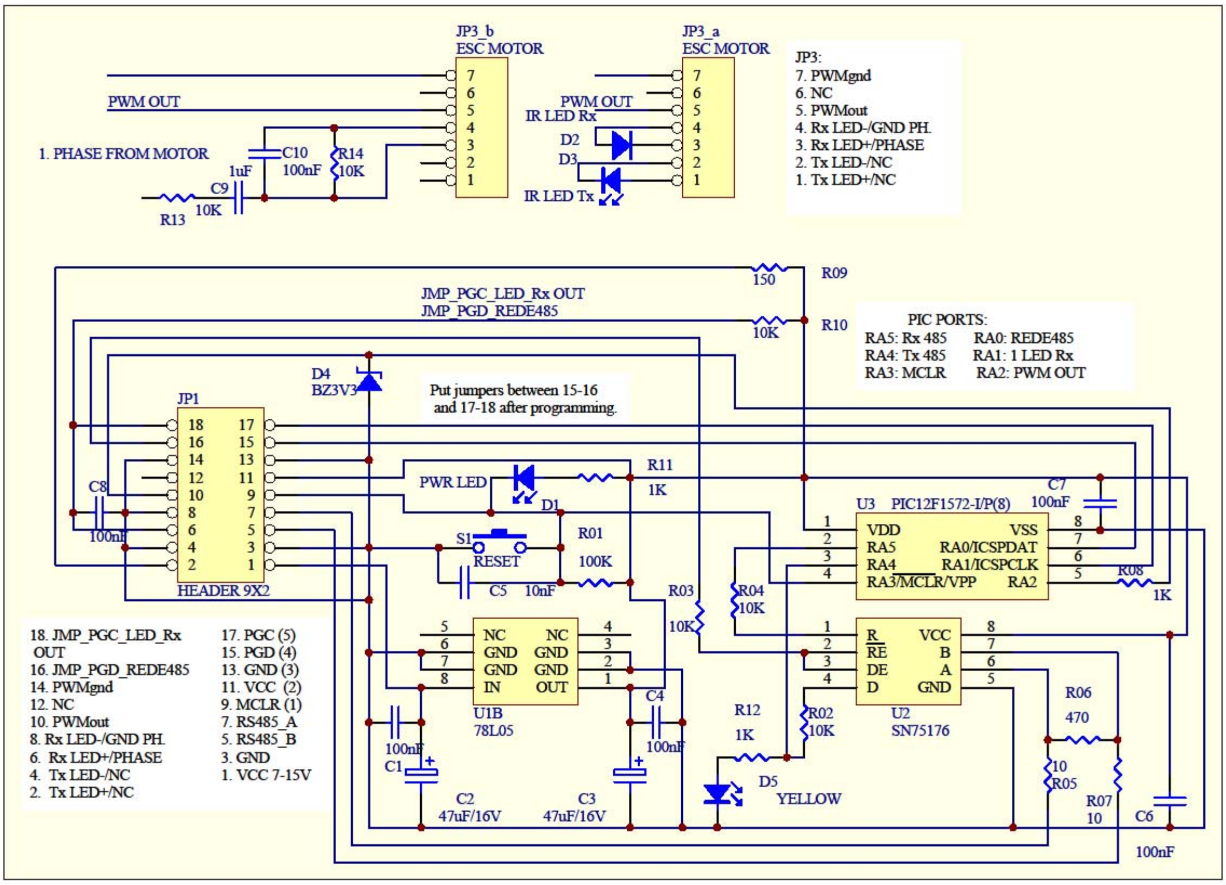

Figure 3. Controller's schematics

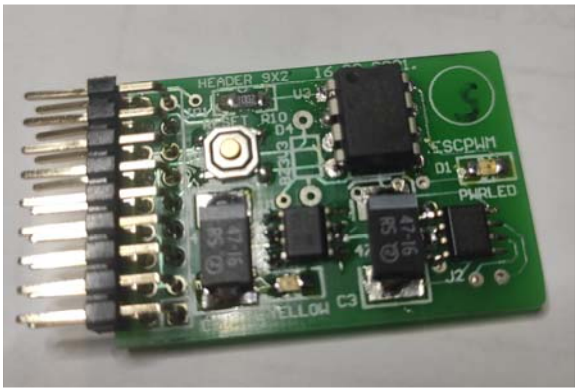

Figure 4. Assembled device

6. DEVICE TESTING

Program development and device testing was performed on an external ESC module and motor with similar characteristics as the drone that was tested. The difference between the ESC in the drone and the external ESC is in the voltage of the active level of the PWM signal. For the drone, that voltage is 3.3 V, while for the external ESC it is 5 V, which can be seen from the omitted Zener diode D4 of 3.3 V on the device.

The use of CRC was initially foreseen in the communication, but later it turned out that it was not necessary because during the development and testing there was not a single case of error in the communication. The device communicates with the computer according to the command-response principle. The computer records the absence of a response from the device as an error and repeats the command. It was found that the optimal period for sending commands is about 0.5 seconds, because this way the trend of the change in the number of revolutions of the engine can be followed, and in case of a wrong measurement, that measurement can be omitted.

During testing, it was noticed that the device stopped working for an unknown reason and that, after the reset, it continued to work. A temporary solution to this situation was the introduction of WDT, which resets the device in such situations. Resetting the device takes a few seconds, which are critical during the measurement because the motor stops working. Later it was found that it was necessary to disable the interrupt until the processing of the previous interrupt was completed. After this program change, no reset due to WDT was observed.

After connecting the drone instead of one ESC to the motor, it was noticed that sometimes the device does not respond to the command. By increasing the time between the transitions from reception to transmission in the computer program, this deficiency was eliminated. The probable reason for this phenomenon is a longer communication wire and more participants in the communication.

7. CONCLUSION

The device for measurement of engine speeds performs its function, but it is necessary to increase its accuracy. One of the reasons of its low reliability is the lack of program memory space in the microcontroller itself. The program now occupies almost all available memory space, less than ten assembly instructions are not filled. Using another microcontroller (PIC16F18015 instead of PIC12F1572) would increase the program space by 4 times and it would be possible to implement an algorithm for more reliable measurement of engine revolutions.

The dimensions of the device could be significantly smaller if SMD components of appropriate dimensions were used and if the connector for programming the microcontroller was eliminated. The programming connector could be replaced by contact pads on the PCB itself.

The command syntax could have been more adapted for this special case, but the syntax previously established for controlling the actuators on the model was respected.

ACKNOWLEDGEMENTS

This work was supported by the Ministry of Education, Science and Techonological Development of the Republic of Serbia (Contract No. 451-03-68/2022-14/200325).

References

[1] https://oscarliang.com/ ESC Firmware and Protocols Overview

[2] RUSSELL,C., WILLINK, G., THEODORE, C., JUNG, J., GLASNER, B.: *Wind Tunnel and Hover Performance Tests Results for Multicopter UAS Vehicle,* NASA/TM-2018-219758, Ames Research Center, 2018.

[3] www.microchip.com, datasheet for PIC12 F1571/2, document DS40001723D.pdf

[4] www.ti.com , datasheet for SN75176A Differential Bus Transceiver, document sn75176a.pdf

[5] GHUDE, S., WEN, P.: *PID controller evaluation with missing samples in a networked control system using spline interpolation method,* International Conference on Computer and Automation Engineering (ICCAE 2009), 8-10 Mar 2009, Bangkok, Thailand