DEVELOPMENT OF HARDWARE SIMULATOR FOR ROTOR SPEED AND PULSE WIDTH MODULATED SIGNALS IN GAS TURBINE SYSTEMS

Károly BENEDA*
AEROK Aviation Technical Training Centre Ltd.
**Corresponding author.* E-mail: kbeneda@enternet.hu

Summary. Gas turbine engines are widely spread in aviation, therefore their research is an important field of industry. The application range is very wide, from the giant turbofans of commercial airliners to the micro turbojet engines of hobbyist usage. Rotor speed is a key parameter in power management, indication and control, used in a vast majority of gas turbine systems. Control signals can also include pulse width modulation, among discrete or other analogue forms. The aim of this paper is to describe the development of a small simulation hardware that is intended to generate variable rotor speed and pulse width modulated signals for control or indicating system research, development, or diagnostics.

Keywords: gas turbine, turbojet, test equipment, rotor speed measurement, pulse width modulation, microcontroller

1. INTRODUCTION

Gas turbine engines are widely spread in aviation; the mostly utilized types are turbofan and turboprop engines, with a little share remaining for turbojets. There is a strong demand in developing gas turbines of reduced size, i.e. various micro turbines form an important field of research: turboprops ([1] or [2]), turbojets ([3], [4] and [5]) or special novel designs even for industrial applications [6].

Power indicating includes rotor speed indication in the vast majority of gas turbines. In a significant number of designs, rotor speed is also used for thrust management, as there is a strong correlation between the two parameters. For measurement, historically there were different types of generators that can produce an electrical signal proportional to the drive speed. These were heavy constructions which can still remain in new designs, if the generated signal does not simply carry the rotor speed information but is strong enough to provide power supply for FADEC systems [7]. In many engines rotor speed is monitored by variable reluctance sensors that can be created in small, light-weight units, apparently beneficial for airborne systems. Despite the different method of operation, both generate variable voltage levels, whose frequency depends on the speed, i.e. frequency measurement can be used to obtain speed information.

Sending proportional control information can be realized in several ways, e.g. current for a torque motor, discrete signal for a solenoid, or pulse width modulation (PWM) as well. Although the latter is not so popular in gas turbine technology, radio controlled hobbyist systems are exclusively based on this solution and several industrial applications can be found as well. It can be shown, if the PWM frequency is high enough, an additional low pass filter can transform the digital pulse train into a proportional analog voltage, or current, i.e. the it can be easily improved into a digital-analog (D/A) converter.

The goal of this article is to describe the development of such a small microcontroller-based simulation hardware that is able to generate digital pulse trains in two basic forms, changing frequency with constant pulse width (both relative and absolute) and changing pulse width of a constant frequency signal. Thus it can provide help to develop electronic equipment that are using these signals from a real gas turbine without the need of real engine runs.

2. DEVELOPMENT OF SIMULATOR HARDWARE AND SOFTWARE

2.1. Design criteria

The developed simulator must be able to create digital pulse trains as stated in the introduction for two purposes:

1. rotor speed signal simulation, requires variable frequency at constant pulse width;
2. PWM signal simulation, which requires stable frequency and alterable pulse width.

The rotor speed signals are estimated between 200Hz and 8 kHz. Depending on the speed sensor hardware, these values correspond to a minimum of 5.000rpm, and a maximum of 160.000rpm. These can be considered as typical rotor speeds mostly in micro gas turbines (e.g. a TJ100 engine has a nominal rotor speed of 59.000rpm [5]).

The pulse width modulated signals must correspond the standard servo control signal definition for radio controlled systems as described in [8], i.e. they must have a constant carrier frequency of 50Hz (20ms nominal repeating rate) and a 5…10% duty (1…2ms absolute pulse width).

As the hardware must be able to perform changes in either frequency or pulse width, some device is required to provide proportional information for these functions. Apparently, a variable resistance connected to an analog-digital (A/D) converter can solve this.

Because of the need to be compatible with commercially available radio controlled systems, the connector layout is also defined. This will determine the available input voltage range as well, which typically falls between 7 and 15 volts DC, as the most commonly used lithium-polymer batteries have 2-4 cells in series each having a nominal 3.7V level.

The hardware and software should be as simple as possible to allow for quick development process.

2.2. Hardware elements

For the above mentioned criteria it is evident, that a microcontroller unit (MCU) should be used that is a special microprocessor based system with integrated peripherals, like timer, A/D converter, etc. Due to the required simplicity, an 8-bit low-end MCU has been selected to perform the tasks, the NXP (formerly Freescale) MC9S08QD4. The leading particulars of the MCU are listed in Table 1 [9].

Table 1 Leading particulars of NXP MC9S08QD4 MCU [9]

Architecture	8-bit von Neumann
Pin count, package	SOIC-8, 1.27mm pitch
Program memory	4kB (FLASH)
A/D converter type	Successive approximation
A/D converter speed	approx. 200kS/s
Digital timers	2 x 16 bit resolution

Maximum clock speed	16MHz
Operating voltage	2,7…5,5VDC
Static RAM	256B
A/D converter resolution	10 bits
A/D converter channels	4
Timer channels	2 + 1

As the evaluation of Table 1 one can state that the selected MCU provides those features what were previously determined. The pinout of the MCU is presented on Fig. 1 together with the final pin usage.

The MCU has two 16 bit resolution timers (TPM1 and TPM2) which can be supplied with clock frequencies up to the bus speed or one can use built-in prescalers of exponents of 2. If one wants to realize e.g. 1MHz timer clock, it can be derived from an 8MHz bus clock and a prescaler of 8. The timer TPM1 has two channels, i.e. two physical pins can be used to either operate as output in a PWM mode or as input capture (e.g. in a rotor speed measurement). These pins (TPM1CH0 and TPM1CH1) will be used as PWM outputs. The timer TPM2 has a single channel, which is dedicated to generate rotor speed sensor simulated signals. However, due to hardware restrictions inside the MCU, there are two separate pins (No. 1 and 2), separately for input (TPM2CH0I) and for output (TPM2CH0O).

The MCU has a 10-bit resolution A/D converter with a maximum of 4 physical channels. It is an additional information that it has a total of 31 channels including some internal channels, like temperature sensor and input voltage measurement, but for external signals only four lines are specified (ADC1P0…ADC1P3). These cannot be used in full extent, as there is a pin multiplexing which is necessary due to the low number of pins (only 8, from which 2 are supply pins). For the given application, as timer TPM1 uses both of its channels, only two channels (on pins no. 3 and 4) remain for analogue purposes. This can be enough; one is dedicated for the PWM and the other for RPM input values, where potentiometers will be used to create a proportional signal what should be converted into the respective digital pulse train.

The MCU is also equipped with general purpose digital input-output features. These are arranged in a theoretically 8-bit port, called as port A (PTA), but the low pin count of the device does not allow for all the bits to be represented on the pins. Practically lines in the range of PTA0…PTA5 have a physical realization on the MCU package; however, as it has been noted at the timers, PTA4 is output-only, PTA5 is an input-only pin. For the second PWM there is no A/D channel left, consequently PTA5 has been selected to operate as a digital input to provide control of the second PWM. Due to the two logical levels available here, the second PWM channel will either provide minimum or maximum duty cycles, there is no possibility for proportional control.

There are some special functions that are also realized but not used in the final operation of the MCU. There are two programming lines, ~RESET and BKGD, these will be soldered temporarily to download firmware into the MCU FLASH program memory. There is the possibility of interrupt raising digital inputs, these are called as keyboard interrupt (KBI1P0…KBI1P3), but these pins are already occupied by timer and A/D functions, furthermore they would not be useful in the present application.

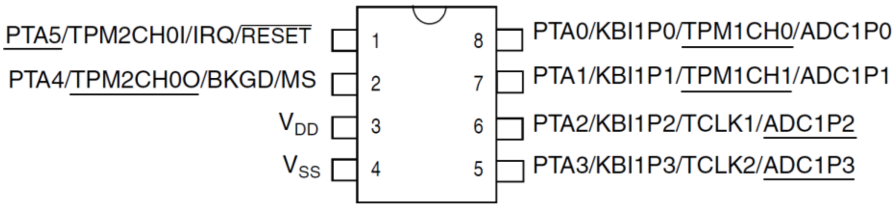

Figure 1 Pin usage of the main MC9S08QD4 MCU

The input voltage of the MCU is significantly lower than the expected battery voltages, therefore a low dropout linear voltage stabilizer must be implemented. Due to the small operating currents, the LP2985 small outline IC is selected, which delivers 5V nominal voltage under a maximum of 50mA load. It has a SOT23-5 package with a very small footprint that helps to keep overall circuit dimensions small.

However the overall sizes of the complete device should be minimized, in order to enable simple manufacturing the printed circuit board (PCB) was selected to be a single-sided design. Due to prototyping circumstances, the double-sided configuration would have more problems (like misaligned layers during manual layout) and the design procedure would be more complex as well.

The theoretical wiring diagram and the corresponding PCB layout can be evaluated on Fig. 2.

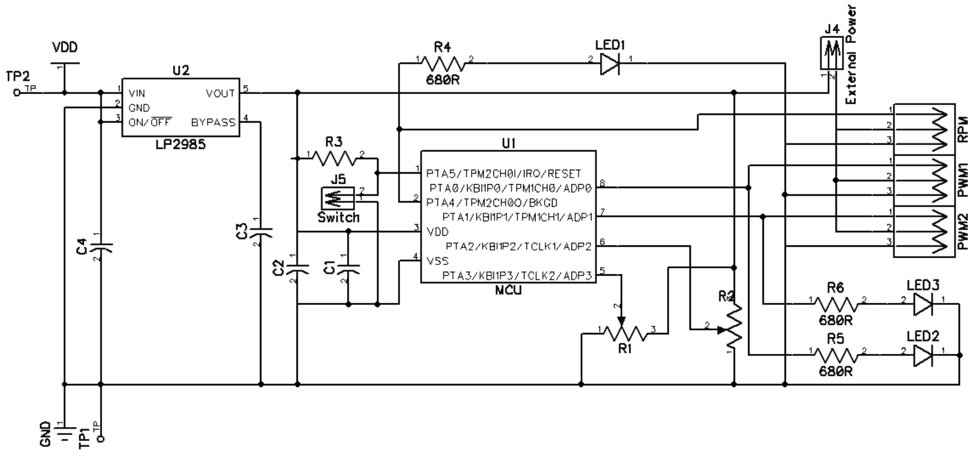

Figure 2 Wiring diagram of RPM and PWM simulator circuit

2.3. Firmware considerations

The firmware is the program what is downloaded into the non-volatile memory of the MCU to control its operation. It must prepare the peripherals for correct behaviour, then enter the main cycle where the following steps must be accomplished:

1. The program must fetch two analogue signals that will determine the pulse width for the first PWM channel and the frequency of the RPM signal.
2. The program must read the state of the digital input to determine pulse width for the second PWM channel.
3. The program must set duty cycle for both PWM channels and the frequency of the RPM channel.
4. The above steps must be repeated until power is removed.

The firmware program is also to be reduced in complexity, so it would not use interrupts, or similar events which are normally necessary in high-level programming. The main function includes an infinite loop that contains the above steps.

The firmware should be capable of allowing some basic selections like PWM duty cycle range or RPM frequency range. This can be easily implemented in such manner that the initial position of the respective potentiometer during power-up phase is acquired first and interpreted as a selection from multiple choices. In order to reduce inadvertent behaviour due to inherent measurement errors of the analogue lines, one must implement only a minimum number of choices (e.g. 2 or 3), and the respective ranges of the 10-bit A/D converter should be positioned far enough from each other. As an example, one can select PWM duty cycle range of 5-to-10% or 0-to-100%. If the PWM potentiometer channel reads 0-100 during power-up, this will result in the usage of the 5-to-10% range, i.e. potentiometer is approximately at zero position. If the given channel reads 900-1023 (maximum resolution of A/D, maximum position of potentiometer wiper) the selected range will be 0-to-100%. This means, that the user is able to perform basic settings for the given operation without the need of implementing higher level communication and increasing the complexity of the circuit.

The firmware is written in CodeWarrior for MCU integrated development environment (IDE), where the C language programs can be established, they can be compiled and built, and finally, the completed program can be downloaded into the MCU memory using a special programming device (P&E USB Multilink). This IDE also provides high-level tools to implement peripheral initialization. Instead of producing the code manually setting all bits to the required values, it allows human language communication to select the desired settings and it automatically generates the code afterwards.

3. THE REALIZED HARDWARE

The PCB has been designed in the single-sided layout due to the above mentioned constraints, and its final size is 37 x 22 millimetres. An additional jumper has been introduced, in order to allow the MCU to be disconnected from the power pins of the standard servo connectors. Many devices (like speed controllers, which require the PWM signal to operate) generate electrical power to drive others (e.g. receiver of the radio control system, servos, etc.). It might happen that the given situation requires the transmission of the PWM signal solely; in this case the MCU would operate from a different source. However, if needed, the jumper can be installed, and the MCU can receive the power from the servo connectors as well.

Two small surface mount (SMD) potentiometers with 3364W footprint have been selected to operate as the source of analogue control signals, whose position will be converted into the digital data.

Relatively small, 1206 form factor SMD light emitting diodes of different colours have been added to visualize signal line operations.

Figure 3 depicts the solder side of the hardware with the programming wires still soldered. These have been removed after the firmware has successfully been downloaded into the FLASH memory of the MCU. Due to the restrictions by design, no dedicated programming interface has been incorporated into the board, in order to reduce complexity, soldered components, etc.

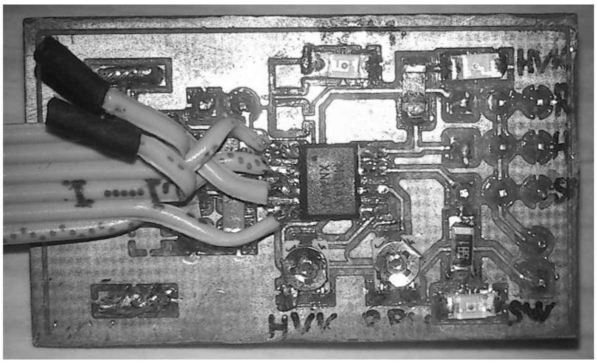

Figure 3 Solder side view of the realized printed circuit board

4. OPERATIONAL TESTS

4.1. Inspection with oscilloscope

Using a digital oscilloscope one can verify the correct operation of the realized device. The built-in measurement options can be used to determine accurate timing parameters of the generated pulse train outputs. Without any modifications required, this first evaluation has confirmed the correct behaviour of the equipment. A typical screenshot of the oscilloscope can be seen on Fig. 4, where the left screen shows the PWM signal, while the rotor speed signal is represented on the right. On both screens the right side column indicates the measured values: peak voltage, frequency, period, positive pulse width and duty of the respective signals can be read.

The PWM signal shown on the left screen of Fig. 4 has a frequency near to 50Hz; the difference is –0.3Hz, which is 0.6% deviation, i.e. it is suitable for most equipment. The positive pulse width is 996μs, i.e. the throttle setting is at the minimum level.

The rotor speed signal indicated on the right side of Fig. 4 was set at the minimum possible frequency as well. The rotor speed produces very short duty cycle peaks, as it was stated above the simulated sensor does the same. However, one can select another logic with the 50% constant duty as well to simulate the behaviour of different equipment.

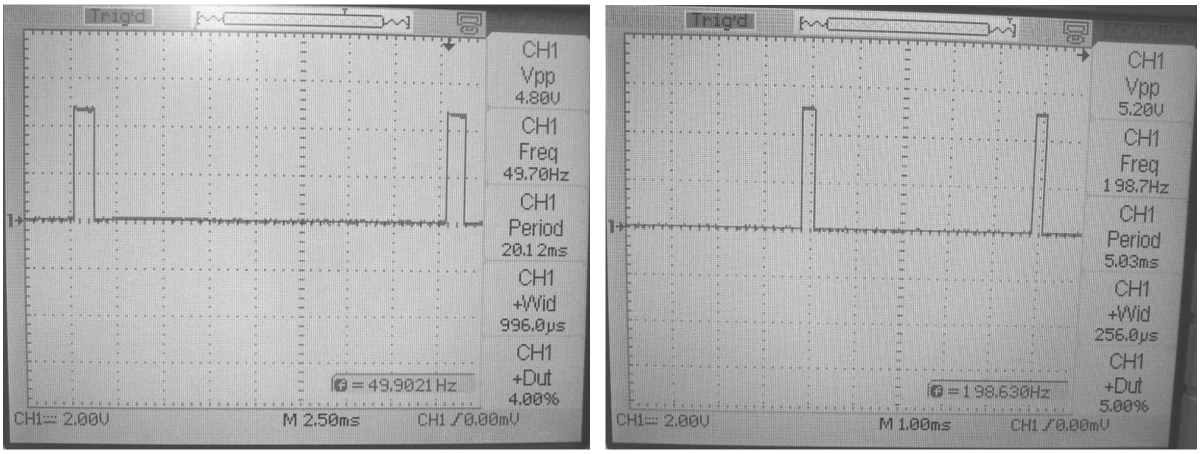

Figure 4 Oscilloscope inspection of the simulator circuit: left – PWM; right – RPM signal

4.2. Operational test using a FADEC controller

The single channel modular FADEC for PD-60R micro turbojet engine called MARCEL has been used to verify correct operation of the hardware under real circumstances. The FADEC system has been described in an earlier article [10].

The controller is able to measure the incoming PWM signal with a resolution of 1μs, i.e. the nominal range of this signal (min. 1; max. 2ms duty) can be measured with 0.1% accuracy, however, the indication shows the percentage of the nominal. On Fig. 5 the value on the LCD in the middle of the upper row (033) is an already transformed information that is obtained from a 1.33ms duty cycle PWM signal. As it has been described in Chapter 1, the minimum is 1ms, maximum is 2ms duty, the shown value is the percentage of this range.

The firmware for the simulator circuit has been written to provide slightly broader range of duty cycles than the standard. If the actual duty cycle falls out of the normal range, the indication will show dashes (see Fig. 5 right bottom corner, separated small picture detail) to identify out-of-range measurement. This feature allows the testing of fault tolerant software solutions that can further operate e.g. after a signal loss.

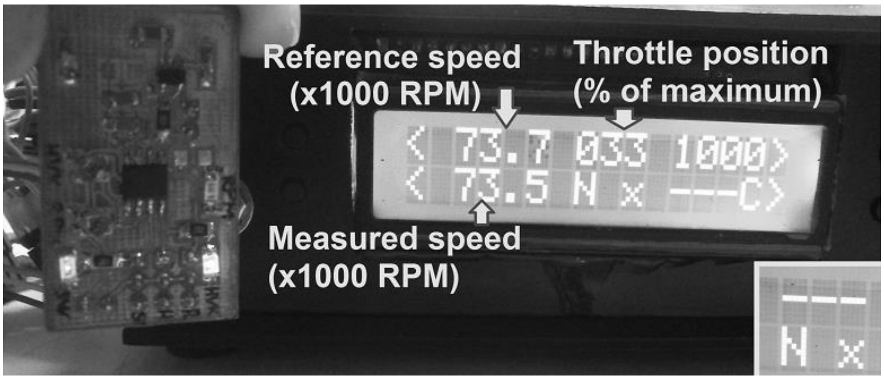

Figure 5 Generating RPM and throttle position PWM signals for the MARCEL turbojet controller

The RPM measurement has a fixed frequency (1MHz) timer for capturing input signal rising edges. Due to the 16-bit resolution of the timer, the measurement can provide approx. 0.08Hz resolution at 200Hz signal (lower end of range), meanwhile the resolution degrades to nearly 64Hz at 8000Hz signal (upper limit frequency). This can be still considered as less than 1% error, i.e. the measurement remains accurate enough.

The final goal is to use the circuit as intended, i.e. generate simulated rotor speed and throttle position signals for the controller, allowing it to be inspected without the need of engine running. This can be a powerful tool when evaluating the behaviour of the controller during the development of various control schemes. Figure 5 shows this cooperation between the different circuits.

5. CONCLUSION

5.1. Results

One can state that the design goals have been reached when the current electronic circuit has been created; it fully meets the prescribed requirements. It is able to generate digital pulse trains in a variable frequency (RPM) and pulse width modulated (PWM) manner, where the main parameter of the pulse train (frequency or duty cycle) can be modified with potentiometers.

Basic settings like frequency and duty cycle range selections for the two outputs can be realized via initial position of the control potentiometers which adds flexibility to the device to be used in different operating environments.

5.2. Further development

Despite the results reached, the current configuration still can be significantly improved in some aspects to enhance the capabilities.

In order to keep simplicity at the present level, increasing of the PWM outputs could be solved with a restricted variety of MCU's, e.g. MC9S08SV16. This chip offers two 16-bit timers with 2 and 6 channels, respectively, with a 12-channel, 10-bit A/D converter, which uses non-multiplexed independent pins, i.e. the digital and analog functions can be used at the same time. The MCU has also an RS232 interface, i.e. two-way communication with a host computer could be also realized to facilitate output control without the need of hardware elements (potentiometers).

6. LITERATURE LIST

References

Journals:

[1] Veress, Á., Bicsák, Gy., Rohács D. Pressure Loss and Flow Uniformity Analysis of Baseline and Redesigned Engine Inlet Duct for a Small Turboprop Aircraft. *Czech Aerospace Proceedings,* 2016/1, pp. 3-9.

[2] Novák, M., Zapach, M., Daňko, P. Analysis of Turboprop Engines Test Results and a Proposal to Improve the Reliability. *Acta Avionica,* 2016, Vol. 34, No. 1, ISSN 1339-9853.

[10] AUTHOR, 2015.

Books:

[7] Linke-Diesinger, A. *Systems of Commercial Turbofan Engines.* Berlin-Heidelberg: Springer Verlag. 2008. 233p., ISBN 978-3-540-73618-9.

Conference Proceedings:

[3] Hnidka, J., Pečinka, J. Measurement automation for small jet engine testing. In: *2016 IEEE 14th International Symposium on Applied Machine Intelligence and Informatics (SAMI).* p. 29-34. ISBN 978-1-4673-8740-8. doi: 10.1109/SAMI.2016.7422977

[4] Fozo, L., Andoga, R., Kovacs, R. Thermo-dynamic cycle computation of a micro turbojet engine. In: *2016 IEEE 17th International Symposium on Computational Intelligence and Informatics (CINTI),* Budapest, 2016, pp. 75-80. doi: 10.1109/CINTI.2016.7846382

[5] Katolický, Z., Bušov, B., Bartlová, M. Turbojet Innovation and TRIZ. In: *16th International Conference on Mechatronics*, pp. 16-23, 3-5 Dec. 2014. doi: 10.1109/MECHATRONIKA.2014.7018230

[6] Calabria, R., Chiariello, F., Massoli, P., Reale, F. CFD Analysis of Turbec T100 Combustor at Part Load by Varying Fuels. In: *Proceedings of ASME Turbo Expo 2015: Turbine Technical Conference and Exposition (GT2015)*, ISBN: 978-0-7918-5679-6, Paper No. GT2015-43455, pp. V008T23A020; 12 pages, doi:10.1115/GT2015-43455

[8] Behnke, S., Schreiber, M. Digital Position Control for Analog Servos. In: *Proceedings of the Workshop on Humanoid Soccer Robots of the 2006 IEEE-RAS International Conference on Humanoid Robots*, 2006, ISBN 88-900426-2-1, pp. 56-61.

Web sites:

[9] *MC9S08QD4 microcontroller unit datasheet – Rev. 6*. 2010. Available at: http://cache.nxp.com/assets/documents/data/en/data-sheets/MC9S08QD4.pdf