PROCEEDINGS

of the 18$^{\text{th}}$ International Conference on Process Control
Hotel Titris, Tatranská Lomnica, Slovakia, June 14 – 17, 2011
ISBN 978-80-227-3517-9
`http://www.kirp.chtf.stuba.sk/pc11`

Editors: M. Fikar and M. Kvasnica

Pestun, I., Halás, M., Kurčík, P.: Control of an RC Helicopter Model Through USB Interface, Editors: Fikar, M., Kvasnica, M., In *Proceedings of the 18th International Conference on Process Control*, Tatranská Lomnica, Slovakia, 174–181, 2011.

CONTROL OF AN RC HELICOPTER MODEL THROUGH USB INTERFACE

Ivan Pestún[*], **Miroslav Halás**[*], **Peter Kurčík**[*]

** Slovak University of Technology, Ilkovicova 3, 812 19 Bratislava, Slovakia*
fax : +........and e-mail : [ivan.pestun, miroslav.halas, peter.kurcik]@stuba.sk

Abstract: This paper presents interface between personal computer and radio controlled helicopter model through an electronic component – microcontroller board. The communication is established through USB interface. The paper describes philosophy of manual RC model control and exploitation of a classical RC model transmitter for controlling an RC model through the computer. Collection of feedback data and GUI for controlling the helicopter are also described.

Keywords: *helicopter control, USB interface, RC model, PWM, PPM*

1 INTRODUCTION

The helicopter is an unstable MIMO system and therefore suitable for tryout of various control problems and algorithms. To retain the helicopter in a stable and controlled flight is a very complex problem. A number of works have been dedicated to study the problem and related subtasks, as for instance creating a mathematical model of a helicopter in (Fogh, et al., 2004; Mettler, 2003; Pestun, 2009), UAV helicopter design (Cai, et al., 2008), stabilization of an RC helicopter in a hover (Hald, et al., 2006; Hald, et al., 2005), to mention a few. Designed control algorithms are usually tested on helicopter models of various sizes and types. For such experiments, it is convenient to put a helicopter in a laboratory and ensure it against damage. There are many possibilities how to control the helicopter. Frequently used approach for controlling the helicopter is to put the control unit directly on the helicopter's body with all the algorithms ensuring autonomous flight and to control desired position and other flight parameters wireless with a ground station PC, see for example (Fogh, et al., 2004; Hald, et al., 2006). This method is good to use especially when the helicopter is about to use outdoors for free flying with already tested control algorithms. The main disadvantage is lower performance of the control unit and the requirement of reprogramming the whole unit in case of every change. Another way how to control the RC helicopter, more suitable for laboratory tests, is to make interface with a high-performance PC and edit the flight and stabilization algorithms there, see for example (Andersen, et al., 2008). Our goal was to create interface between the helicopter and PC with high performance. The interface is also small control unit, but it is only used for transforming and forwarding signals from computer to the helicopter and for collecting feedback from sensors and sending them back to computer. So there is no need to reprogram the unit during the experiments, because all the control algorithms are programmed on the PC where they can be, thanks to enough performance, more easily edited in more sophisticated and user friendlier programs than microcontroller programming language is. Hence, this way was found to be more suitable for testing miscellaneous controllers for helicopter stabilization and control in laboratory conditions, forming the main scope of our interest in this work. In particular, the attention is paid to the communication established through USB interface. For that reason the so-called trainer mode is employed, allowing us to interconnect PC and the helicopter. Note that the similar idea was suggested in (Net-Scale Technologies, 2004) where an RC model of a car was controlled.

The paper is organized as follows. Section 2 describes signals used for control of the RC model, transmitter's mode used for interfacing the RC model and microcontroller board used as the interface element, Section 3 describes sensors used for feedback and their inadequacies and, finally, Section 4 describes the developed software for the helicopter control.

2 BASIC PRINCIPLES OF MANUAL RC MODEL CONTROL

2.1 *Signals used for control of an RC model*

The RC models, whether they fly, drive or float, consist of receiver, which receives the signals from a transmitter and controls of all actuators like driving gears and servo motors. Driving gears give the RC model propulsion and servo motors handle the controls. All this RC models actuators are controlled by an RC pulse signal. This signal is a specialized form of the so-called PWM (Pulse Width Modulation) signal. Typically the signal's period is 20 *ms* (50 *Hz* PWM frequency), but the signal itself has a width from 1 *ms* to 2 *ms* (The Model Electronics Company, Tech. note).

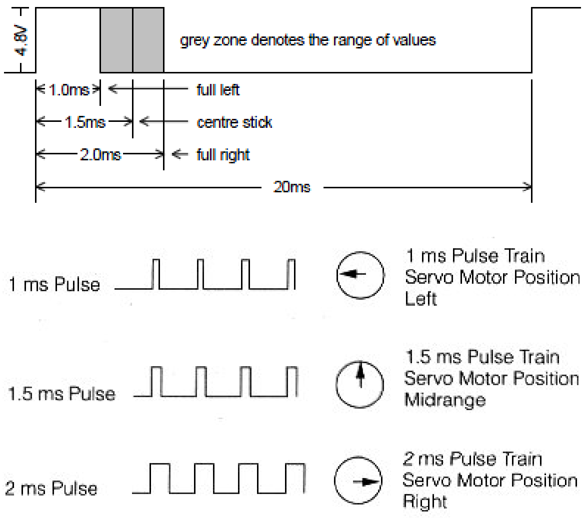


Fig. 1. RC pulse signal and the corresponding servo position examples.

The signal width represents the position of the servo or the power of the driving gear. The midrange (or center) position represents the signal with width of 1.5 ms. Then the left position is represented by the signal with width of 1 ms, and the right position is represented by the signal width of 2 ms (Vastianos, dipl. thesis), as shown in Fig. 1.

Each single actuator is controlled by its own PWM signal supplied by the receiver. For example, when the model has 6 actuators (like in our case of RC helicopter), 6 independent PWM signals (6 channels) are needed. Sending each PWM signal from the transmitter for each actuator separately would cost a lot of energy and the update frequency would depend on a number of signals. In order to be more effective, the information related to each channel is sent in a serial fashion, one after the other, over a single RF link (The Model Electronics Company, Tech. note). This signal is the so-called Pulse Position Modulation, or PPM.

PPM is a modulation which uses pulses with variable width and uniform height and time between the pulses. The PPM signal consists of channel sections (the number depends on the number of actuators) and a synchronization time space. Channel section is composed of a fixed time, usually 0.5 ms, and of a variable time with length of 0.5 ms to 1.5 ms (Fig. 2a). The synchronization time is the "dead time" and varies with the number of channels and also with the channel content. However, even for a system with more than 8 channels, it is much longer than the time between channel pulses. The receiver uses this synchronization time to synchronize itself to the pulse train, so that the positional information for channel 1 always drives the correct servo (The Model Electronics Company, Tech. note). Also in case of a signal loss in the middle of the transmitting, the receiver knows that after a time space much longer than 0.5 ms the first channel follows. Thus, when the signal recovers, the channels will not be disordered.

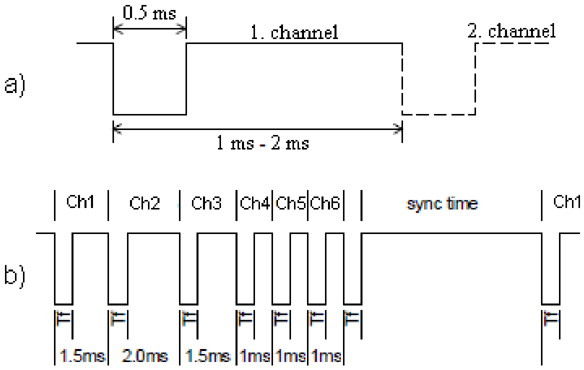


Fig. 2. a) single channel section of a PWM signal; b) example of a PPM signal carrying information for six channels

The variable time of the pulses can be compared to the pulse width in PWM. For example, 6 PWM pulses with the pulse widths of 1.5 ms, 2 ms, 1.5 ms, 1 ms, 1 ms, 1 ms will be modulated to a PPM signal shown in Fig. 2b.

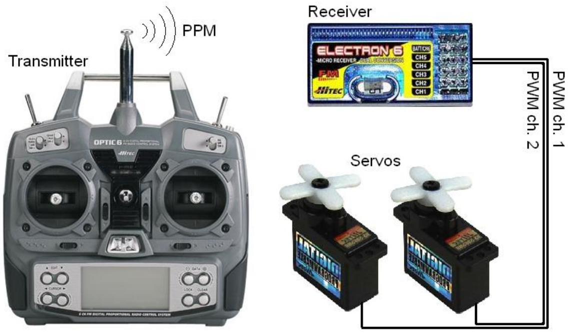


Fig. 3. Communication between the transmitter and receiver through PPM signals and servo control from the receiver through PWM signals.

In a standard manual hand control, the amount of variations on the transmitter's joysticks changes into the above mentioned PPM signal carrying information for controlling all the actuators, which is transformed into a radio signal and transmitted in a specific frequency. The receiver on the model, which is controlled by the transmitter, constantly monitors this frequency. When the radio burst from the transmitter is received, the receiver transforms the PPM signal to the PWM signals and split them up to the channels - for each actuator its own PWM signal, see Fig. 3.

2.2 *Description of a "trainer" mode*

In this section the so-called trainer mode is described, for it is employed to interconnect PC and the RC model.

In case when for instance the trainee pilot has problems with the control of the RC model, most of the transmitters are fitted with a trainer mode. For using this mode, two transmitters have to be interconnected with a cable, see Fig. 4. The signal, which the interconnected transmitter sends through the cable, is the inverted PPM signal with amplitude of transmitter's supply voltage.

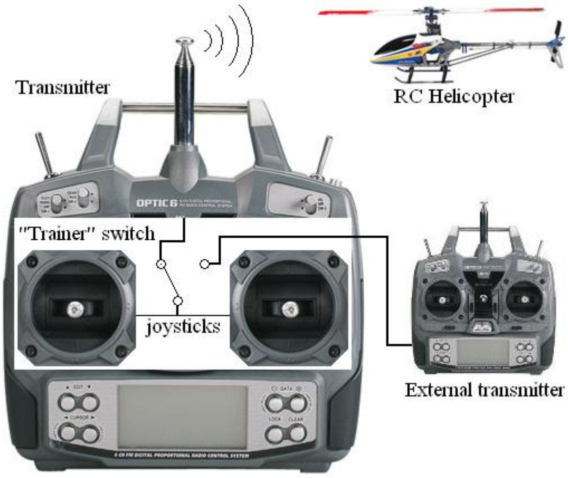


Fig. 4. Trainer switch function.

When the trainer mode on the transmitter, which is controlling the model, is turned on, the transmitter stops to send signals created from the positions of the joysticks and starts to send the signals from the interconnected transmitter, which is for instance in the hands of experienced pilot. However, in this work this idea was used to develop and establish an interconnection with PC, rather than with the second transmitter.

2.3 *Microcontroller interface board description*

To achieve such an interconnection the interface board, which creates PPM signals asked by the computer, was connected to the trainer port of the receiver. Through the USB the PC sends desired values for all channels to microcontroller interface board which creates, based on this values, corresponding PPM signal and sends it to the transmitter. For this purpose the transmitter is set to the trainer mode all the time, so the signals are forwarded from the interface board to the helicopter's receiver. Besides the sending of control signal, the microcontroller interface board ensures data collection from sensors and shifts them to the computer for evaluation and handling. The scheme of connection is shown in Fig. 5.

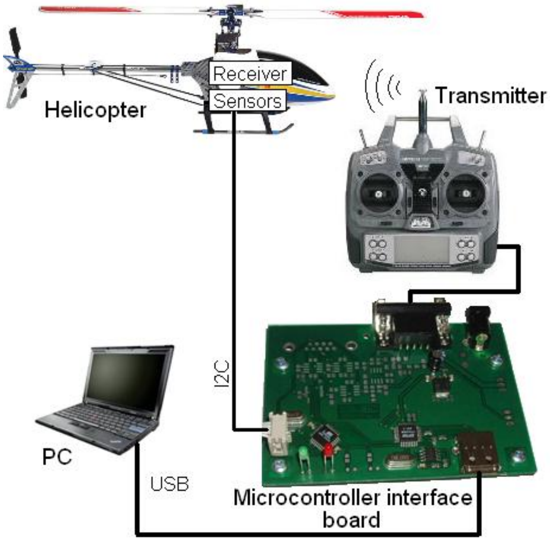


Fig. 5. The scheme of Helicopter – PC interconnection.

In order to match the desired transmitter's input polarity of the PPM signal via "trainer" connection, the PPM signal's amplitude have to be switched to the level of the transmitter's supply voltage. For the case of the chosen transmitter, it is the voltage 9.6 *V*. Example of the PPM signal entering the transmitter with all channels set to width of 1 *ms*, which represents left position on all the servos, can be seen in Fig. 6a.

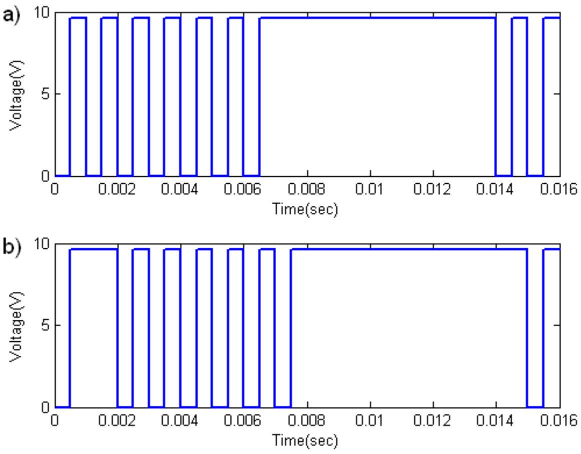


Fig. 6. PPM signal entering the transmitter from the "trainer" cable. a) All channels set to width of 1 *ms*. b) Channel 1 set to 2 *ms*.

Example of the PPM signal entering the transmitter with the first channel set to width of 2 *ms*, which represents right position of the servo connected to the first channel, can be seen in Fig. 6b.

To switch the PPM signal's amplitude to the level of the transmitter's supply voltage an NPN transistor was used in the "trainer" cable. However, the transistor also inverts the PPM signal. Therefore the signal generated by the microprocessor needs to be created in an inverted form (see Fig. 7). Such a PPM signal modification was necessary, otherwise the transmitter would not respond to the signal TTL levels from the microprocessor.

The PPM signal is generated in the microprocessor using a 16 bit counter TCNT1, set in a fast PWM mode with a variable length of the counter's TOP register. Since the 6 channels are needed, a state machine is created within the counter's overflow interrupt routine to alter the counter's registers after each transmitted impulse and counter overflow. The state machine sequence cycles through all 6 channels and then holds the signal low to transmit the synchronization delay. The whole process repeats in every cycle. In case of change of some channel, the new information affects the PPM signal in the next cycle. The outgoing signal from the microprocessor has amplitude of 5 *V*, synchronization time 7.5 *ms* and seven pulses with width 0.5 *ms*, displaced from each other 0.5 *ms* – 1.5 *ms*. Example of the signal generated by the microprocessor, which will lead, after inverting by the transistor in the "trainer" cable to the PPM signal shown in Fig. 6, can be seen in Fig. 7.

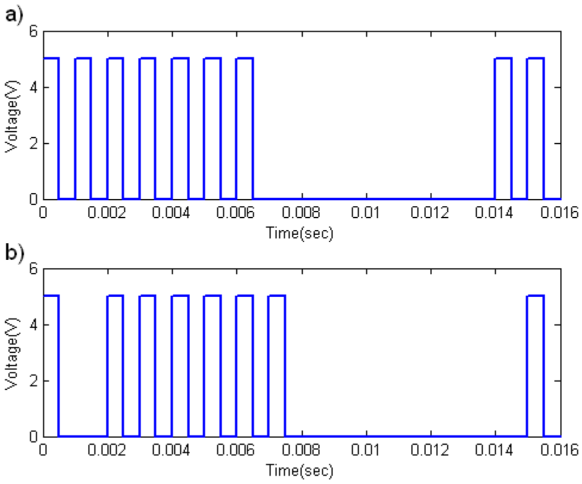


Fig. 7. Inverted PPM signal generated by the microprocessor. a) All channels set to width of 1 ms. b) Channel 1 set to 2 ms.

The PPM generator hardware is based on the Atmel ATmega8 microcontroller which features counters with PWM generation option and also serial port which is used to interface with PC using the serial to USB converter. FTDI FT232RL chip play a role of a serial to USB converter. It is easy to interface and incorporate in user programs either in C++ or Delphi. The microcontroller is also used to interface the sensors like 3-axis accelerometer and gyroscopes and is able to provide the acceleration and angular velocity readings via the USB interface for the PC. There are also unused free pins on the board which may be used for connection of additional sensors or other hardware if the need arises. The whole microcontroller interface board can be seen in Fig. 8.

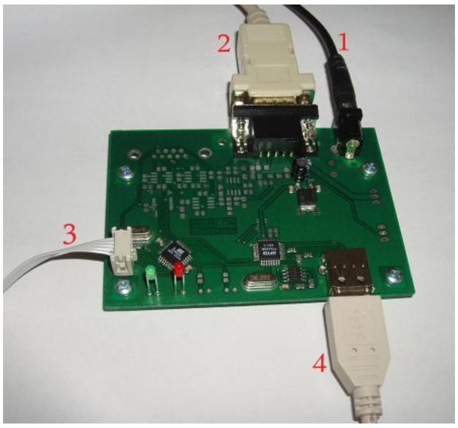


Fig. 8. Microcontroller board used as interface between the RC helicopter and computer. (1- Power connection; 2- connection with the transmitter; 3- sensors connection; 4- USB connection with PC)

3 FEEDBACK FROM HELICOPTER

To control the helicopter, a feedback is needed to detect the helicopter states. For measuring accelerations of the helicopter's body a triple axis accelerometer has been connected to the microcontroller interface board. The accelerometer measures accelerations in three axes orthogonal to each other. For detecting the heeling of the helicopter two single axis gyroscopes are employed and measure the angular velocities about the helicopter's longitudinal and lateral axes.

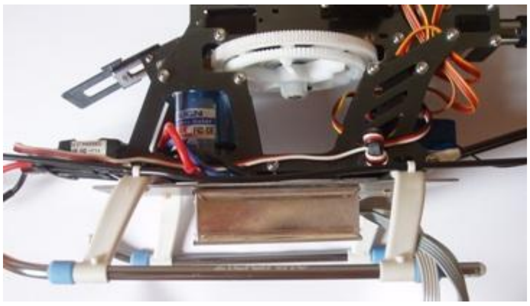


Fig. 9. Housing of the sensors attached to the helicopter.

These sensors were put into a thin metal plate box (see Fig. 9), which dispose of similar characteristics to a Faraday cage and bring the protection against spurious fields.

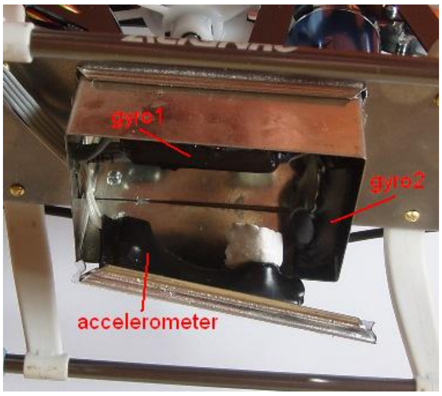


Fig. 10. Inside of the sensors housing.

The gyroscopes were mounted on the front side and the right side of the box such that they are orthogonal to each other (see Fig. 10) and measure the angular velocities about two axes. The accelerometer was mounted on the bottom side of the box (see Fig. 10).

3.1 *Triple axis accelerometer*

For our purposes, the triple axis accelerometer based on a LIS3LV02DQ chip (see Fig. 11) was chosen and connected to the microprocessor board through the I2C bus. The LIS3LV02DQ is a capacitive accelerometer chip. When the acceleration is applied to the sensor, the proof mass displaces from its nominal position, causing an imbalance in the capacitive half-bridge. This imbalance is measured using charge integration in response to a voltage pulse applied to the sense capacitor. The complete measurement chain is composed by a low-noise capacitive amplifier, which converts into an analog voltage the capacitive unbalanced voltage of the MEMS sensor and by three ΣΔ analog-to-digital converters, one for each axis, that translate the produced signal into a digital bit stream. The acceleration data may be accessed through an I2C/SPI interface, thus making the device particularly suitable for direct interfacing with a microcontroller (STMicroelectronics, 2005).

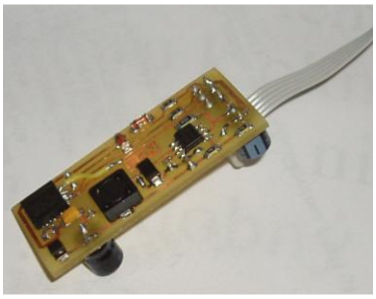


Fig. 11. Tripple axis accelerometer used for sensing accelerations of the helicopter’s body.

The accelerometer measures three types of accelerations: decomposition of the gravitational acceleration, accelerations caused by the translational displacement of the helicopter and accelerations caused by a centrifugal force when the helicopter is turning.

3.2 *Inadequacy of an accelerometer*

The accelerometer is mainly used for sensing the heeling of the helicopter. The principle is to measure the gravitational acceleration decomposed into the three orthogonal axes of the accelerometer, as can be seen in Fig. 12. From the components of the gravitational acceleration measured by the accelerometer, the longitudinal and lateral heeling of the helicopter can be estimated.

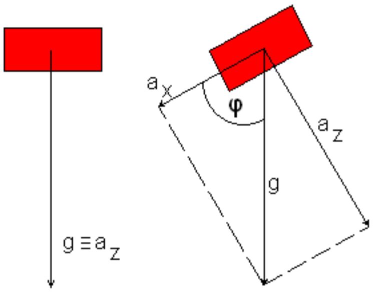


Fig. 12. Decomposition of the gravitational acceleration by the heeling of the acceleromete (mounted on the helicopter’s body).

As mentioned in Section III A, acceleration is measured by the displacement of the proof mass from its nominal position in the sensor. To depict this idea, a ball in a box is chosen to represent the accelerometer (Fig. 13). If the box heels the ball starts to move towards the direction of the heeling. This is, therefore, understood as an acceleration measurement in this direction, as shown in Fig. 13a. However, if the box is accelerating horizontally (and not heeling), the ball starts to move in to the opposite direction, which is also understood as an acceleration measurement but in the opposite direction, as shown in Fig. 13b. Clearly, this causes problems, for one cannot distinguish whether the helicopter heels or accelerates in the opposite direction. Therefore, the helicopter feedback has to be improved and is, for that reason, usually equipped with gyroscope sensors.

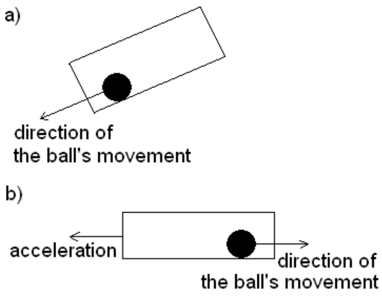


Fig. 13. Simplified figure of acceleration measuring principle in one axis. a) Acceleration measured by the heeling of the accelerometer. b) Acceleration measured by the linear acceleration of the accelerometer.

3.3 *Piezoelectric gyroscopes*

Heeling of the helicopter can be measured with the gyroscope. Classical rotating mechanical gyroscopes are rather expensive and for our purposes even too heavy and big-sized.

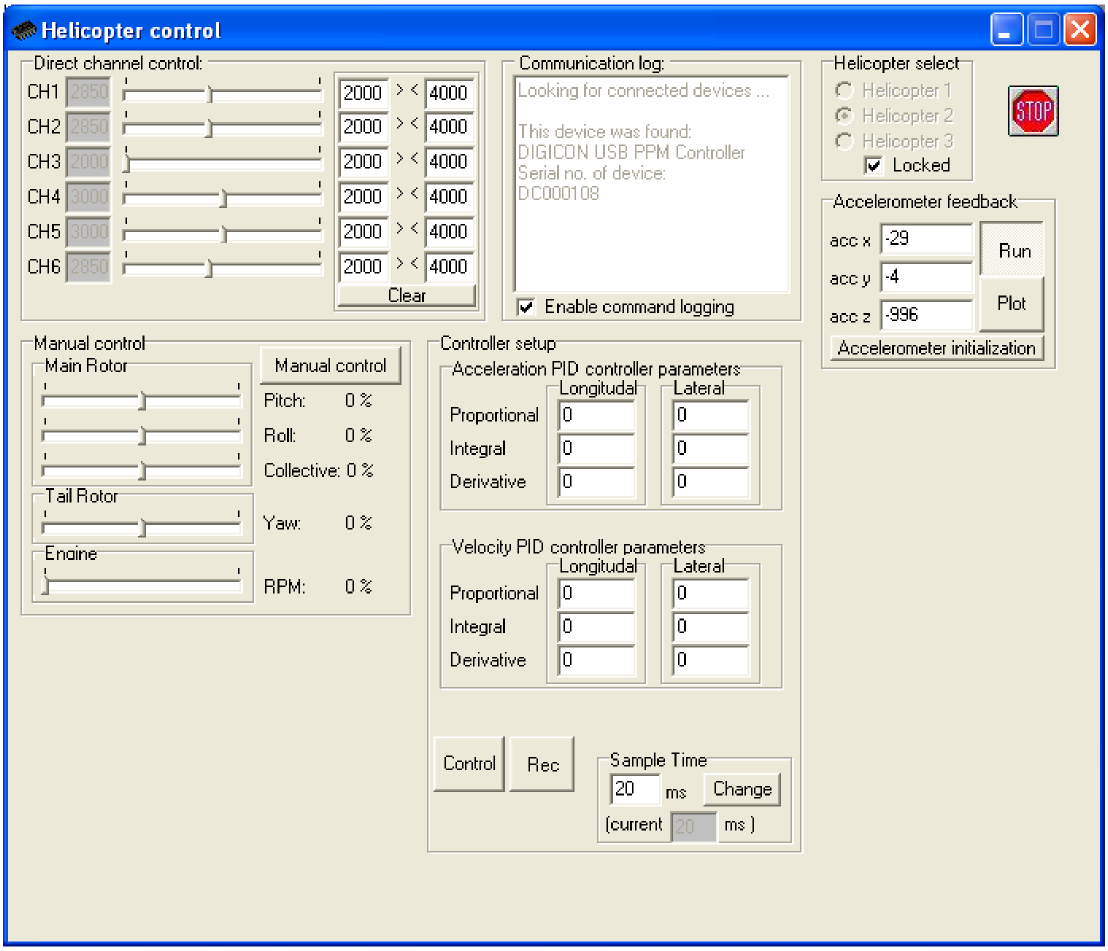
Fig. 14. Piezoelectric gyroscope with XV-3500CB chip used for sensing angular velocities of the helicopter's body.

Therefore, piezoelectric gyroscopes were chosen. Unlike the classical rotating mechanical gyroscope, the piezoelectric gyroscope does not measure the turn, but the angular velocity by the turning. The piezoelectric gyroscope consists of a vibrating piezoelectric material which tends to keep the vibrations in the same plane as its support is rotated. A Coriolis force can be measured to produce a signal related to the rate of rotation (Wikipedia, 2010).

For our purposes two angular rate piezoelectric gyroscope boards with XV-3500CB chip (see Fig. 14) were chosen and connected to the microprocessor board through the I2C bus, like the accelerometer. One of the gyroscopes is used for measuring the angular velocity about the longitudinal axis of the helicopter and the other for measuring the angular velocity about the lateral axis of the helicopter. The gyroscope board uses MCP3421 A/D converter chip for interfacing the I2C communication. This chip has a fixed address 1101000X (if "X" stands for 1, the AD conversion value and configuration bytes can be read, if "X" stands for 0, the configuration bytes writing would follow) (Sure electronics, 2008). Because the communication with two devices with the same address connected to the I2C bus would not be working, only one gyroscope was connected to the hardware I2C bus, used also for collecting the data from the accelerometer. For the second gyroscope, the new software I2C bus was created.

4 SOFTWARE FOR HELICOPTER CONTROL

The software for controlling the helicopter with the PC was programmed in BORLAND C++ Builder. The main GUI window of the software consists of



Helicopter control

Direct channel control:

CH1 2850 2000 > < 4000
CH2 2850 2000 > < 4000
CH3 2000 2000 > < 4000
CH4 3000 2000 > < 4000
CH5 3000 2000 > < 4000
CH6 2850 2000 > < 4000

Communication log:

Looking for connected devices ...

This device was found:
DIGICON USB PPM Controller
Serial no. of device:
DC000108

☑ Enable command logging

Helicopter select
○ Helicopter 1
● Helicopter 2
○ Helicopter 3
☑ Locked

Accelerometer feedback
acc x -29
acc y -4
acc z -996

Manual control
Pitch: 0 %
Roll: 0 %
Collective: 0 %
Yaw: 0 %
RPM: 0 %

Controller setup

Acceleration PID controller parameters

	Longitudal	Lateral
Proportional	0	0
Integral	0	0
Derivative	0	0

Velocity PID controller parameters

	Longitudal	Lateral
Proportional	0	0
Integral	0	0
Derivative	0	0

Sample Time
20 ms
(current 20 ms)

Fig. 15. Main window of the program.

several blocks with various functionalities, as can be seen in Fig. 15. The helicopter can have controlled each channel manually in a "Direct channel control" block. This can be done with moving 6 different track bars.

Longitudinal input (heeling of the helicopter forward/backward), lateral input (heeling of the helicopter to the sides) and collective input (ascent/descent of the helicopter) are controlled by the variable pitch of the helicopter's main rotor blades, executed by three servos. Hence, the control of these three channels has to be mixed such that the main rotor can be controlled correctly. Mixed control signals for controlling the direct helicopter inputs are controlled in the program with "Direct control" block. The software also offers sampling time modification, graphs for measured accelerations and angular velocities and basic PID controllers which serves as a first choice in experiments with the helicopter stabilization. For first experiments with helicopter stabilization, only feedback data from accelerometers were used.

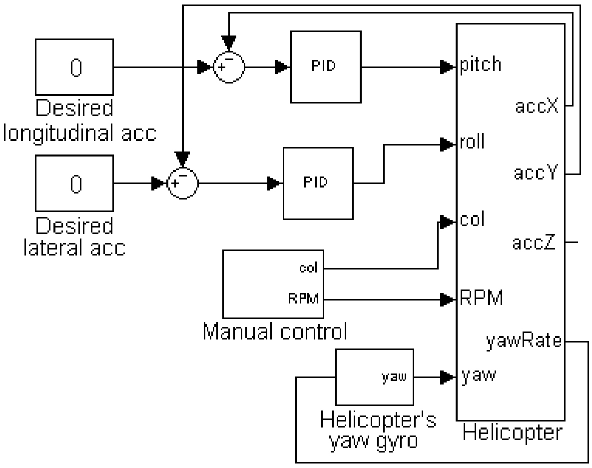


Fig. 16. Helicopter stabilization scheme using feedback data from accelerometer.

The aim was to keep longitudinal and lateral accelerations equal zero. Computer controlled pitch and roll of the helicopter, while the operator has a manual control of rotor RPM and collective angle of attack for controlling ascent/descent of the helicopter. RC helicopters are equipped with yaw gyroscope, which can hold the yaw position to make the control of the helicopter simpler. This yaw control was exploited to simplify the stabilization. The scheme used for this approach can be seen in Fig. 16. Values of PID controllers for longitudinal and lateral stabilizations can be set in "Acceleration PID controller parameters" block of the program (see Fig. 15).

Another approach for stabilizing the helicopter, which can be seen in Fig. 17., was to use feedback data from gyroscopes. Computer controls pitch and roll of the helicopter so that pitch and roll rates measured by the gyroscopes were zero. For this approach the operator also has manual control of the helicopter's ascent/descent and the yaw control is kept on the helicopter's gyro

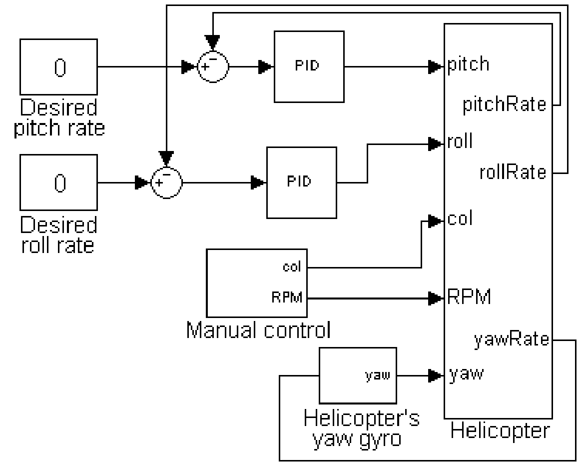


Fig. 17. Helicopter stabilization scheme using feedback data from accelerometer.

5 CONCLUSION

The aim of this work was to describe the developed interface between the helicopter and PC, which controls the helicopter and collects the feedback data. The PC sends through the USB desired values for all channels to the developed microcontroller interface which creates, based on these values, the corresponding PPM signal and sends it to the transmitter. The transmitter, which is set to the "trainer" mode, forwards the signals to the receiver and controls the RC helicopter. The helicopter has a triple-axis accelerometer and two piezoelectric gyroscopes ensuring the feedback from the helicopter. The sensors are connected through the I2C to the microcontroller interface board which collects the feedback data from sensors and sends them back to the computer through the USB. Evaluation of the feedback signals and creation of the control signals is treated by the PC. For this purpose, the software for controlling the helicopter manually and stabilization based on a PID controller was developed. However, due to the delays caused by the feedback data evaluation and processing, the stabilization did not work properly. By demand of the real time control, our next task is to develop software in Linux environment. In addition, the feedback data evaluation is to be improved. From that point of view, the complementary filter, described in (Rodina, et al., 2009), seems to be appropriate. As a control algorithm, the LQ controller described in (Kozakova, 2008) and, for example, used for the helicopter stabilization in (Andersen, et al., 2008), seems to be convenient.

ACKNOWLEDGMENTS

The work was supported by a grant (No. NIL-I-007-d) from Iceland, Liechtenstein and Norway through the EEA Financial Mechanism and the Norwegian Financial Mechanism. This project is also co-financed from the state budget of the Slovak Republic.

6 REFERENCES

Andersen, T. L., D. F. Lauritzen, J. T. Madsen, M. M. Sørensen, B. A. Mertz, Autonomous Inverted Hover of a Small Scale Helicopter, Report, Institute of Electronic Systems, Aalborg University, 2008.

Cai. G., L. Feng, B. M. Chen, T. H. Lee, Systematic design methodology and construction of UAV helicopters, Mechatronics, vol. 18, 2008, pp 545–558.

Fogh, M., T. H. Mikkelsen, E. Mustafic, R. Pettersen, J. M. Sommerset, Autonomous Helicopter, Report, Institute of Electronic Systems, Aalborg University, 2004.

Hald, U. B., M. V. Hesselbæk, M. Siegumfeldt, Nonlinear Modeling and Optimal Control of a Miniature Autonomous Helicopter, Diploma Thesis, Institute of Electronic Systems, Aalborg University, 2006.

Hald, U. B., M. V. Hesselbæk, J. T. Holmgaard, Ch. S. Jensen, S. L. Jakobsen, M. Siegumfeldt, Autonomous Helicopter - Modelling and Control, Report, Institute of Electronic Systems, Aalborg University, 2005.

Kozakova, A., Design of discrete-time compensator for reference tracking with disturbance rejection, Transactions of the VSB, in press, 2008.

Mettler, B., Identification Modelling and Characteristics of Miniature Rotorcraft, Kluwer Academic Publishers, 2003.

Pestun, I., Helicopter control, Diploma Thesis (in Slovak), Slovak University of Technology in Bratislava, Faculty of Electrical Engineering and Information Technology, 2009.

Rodina, J., P. Hubinsky, “Stability control of segway like differential drive by using MEMS sensors”, Conference on Modeling of the Mechanical and Mechatronic Systems, Zemplinska Sírava, Slovakia, pp. 483-487, 2009.

Vastianos, G., 16 channel servo controller for robotic applications, Diploma thesis, Faculty of Technological Applications, Technological Educational Institute of Piraeus,.

Net-Scale Technologies, Inc., Autonomous Off-Road Vehicle Control, Final Technical Report, 2004.

STMicroelectronics, LIS3LV02DQ MEMS inertial sensor, datasheet, 2005.

Sure electronics, Angular-Rate Sensor XV-3500CB Prototype PCB User’s Guide, 2008.

The Model Electronics Company, The PPM Radio Control System: Part 2, Technical Note MTN004.

http://en.wikipedia.org/wiki/Vibrating_structure_gyroscope , 2010.