

Project

Goal: Implementing a basic control system for an autonomous underwater glider.

A microcontroller board is connected to three motors. One is the main propeller, pushing the glider forward. The second motor actuates the rudder of the glider, allowing it to have a yaw motion. The third one is a linear motor that moves the battery pack, which in turns allows to statically change the pitch of the vehicle.

The microcontroller receives desired reference values for the forward speed, the pitch, and the rudder angle. These reference signals are sent through a serial interface. The microcontroller sends feedback messages back to the control PC to report a few status information.

Hardware specifications

- The main motor can run from -10000 to 10000 RPMs;
- The second and third motors can run from -100 to 100 RPMs.
- The motors are controlled through a PWM signal.
 - The frequency must be 1 kHz.
 - For motor 1, 50% duty cycle corresponds to 0 RPM, 0% corresponds to -11000 RPM and 100% corresponds to 11000 RPMs.
 - For motor 2 and 3, 50% duty cycle corresponds to 0 RPM, 0% corresponds to -110 RPM and 100% corresponds to 110 RPMs.
 - A dead time of at least 3 microseconds should be used to prevent problems with the H-bridge controlling the motors.
 - Running the motors above their limits can damage them and should be avoided.
- Ignoring dynamics, when motor 1 runs at 10000 RPMs the vehicle moves with a forward speed of 2 m/s. The speed is approximated to be linear with the RPMs.
- When motor 2 runs at 100 RPMs, the battery pack moves with 5 mm/s. The battery pack can move from position 0 to a maximum of 10 cm. The pitch is linear with the position of the battery pack: again, ignoring dynamics, when the battery is at position 5 cm, the pitch is 0 degrees. When the battery is at position 10 cm, the pitch is +20 degrees, and when it is at 0 cm, the pitch is 0 degrees.
- When motor 3 runs at 100 RPMs, the rudder moves with 5 deg/s. The rudder can move from -30 to 30 deg.

Firmware requirements

- The control systems should compute the desired RPMs of the three motors, given the reference speed and position (pitch and rudder angle).
- The firmware must simulate the movement of the rudder and of the battery pack, using the aforementioned relationships between PWM and derivatives of rudder angle and battery pack position (integrating the velocities).
- The control system must never generate PWM signals outside of the specifications of the motors.
- The rudder angle and battery pack limits should never be exceeded.
- If no references (i.e., the MCREF command) are received from the PC for more than 5 seconds, the firmware should enter a timeout mode:
 - All motors velocity should be set to zero.
 - Led D4 should blink at 5 Hz to signal timeout.
 - When a new reference is read, then the led D4 should stop blinking and commands should be given again to the motors.
- The firmware must support receiving references at least at 10 Hz frequency (through a proper choice of baud rate).
- The firmware must refresh the PWM values at least at 10 Hz frequency.
- The firmware must send the PWM feedback message MCPWM at 5 Hz frequency.
- The firmware must send the Position feedback message MCPOS at 10 Hz frequency.
- The control system should blink led D3 at 1 Hz to always signal a correct functioning of the main loop, regardless of any state.
- Given the chosen UART baudrate, the firmware should never lose a message due to its implementation (i.e., proper dimensioning of buffers), even with full use of the bandwidth
- The firmware should write on the LCD
 - First row: "Speed: x", where x is the desired forward speed
 - Second row: "R: n1", where n1 is the applied motor1 RPM (e.g. "R: 900")
 - If the button S6 is pressed, the data displayed on the LCD changes as follows:
 - First row: "R: x P: y", where x is the desired rudder angle, and P is the desired pitch angle
 - Second row: "R: x P: y", where x is the *actual* rudder angle, and P is the *actual* pitch angle
 - If the button S6 is pressed again, the data displayed toggles again to the first one.

Messages from the PC

\$HLREF,speed,pitch,rudder* where *speed* (m/s) is the desired linear velocity for the robot, *pitch* (deg) is the desired pitch angle, and *rudder* (deg) is the desired rudder angle.

Messages to the PC

\$MCPWM,n1,n2,n3* where $n1$, $n2$, $n3$ are the *applied* RPM.

\$MCPOS,p1,p2*, where $p1$ (deg) is the position of the rudder, and $p2$ (mm) is the position of the battery pack.