Arduino-based PID controller for an Undergraduate Laser Cooling Experiment

Robert W.A. Brooke, Julie M. Gillis, Isaac Davies, Theodore A. Corcovilos
Dept. of Physics, Duquesne University, Pittsburgh, PA; Pittsburgh Quantum Institute (http://www.pqi.org)

INTRODUCTION

Arduino is an open-source microcontroller platform designed for quick development and easy interfacing, making it ideal for novice programmers and instrument designers. We have chosen to use the Arduino because it streamlines data acquisition and processing when compared to circuits using discrete components and has robust enough software to run complex algorithms and control routines on the microprocessors themselves.

The use of a digital feedback controller allows us to place the relevant signal processing functions and input/output equations into the code, rather than calculate the response ahead of time and build a dedicated circuit for each instrument. This allows for a flexible “central circuit” and algorithm which can be used with a variety of inputs (analog or digital), and produce a variety of outputs.

With the central algorithm in place, the relatively simple input and output circuits can be easily prototyped and debugged, allowing for rapid implementation and development. Likewise, the gain parameters can be easily changed through the display, allowing a novice designer without a controls background to determine the necessary gain parameters through “brute force” experimentation or well-known protocols like Ziegler–Nichols.

The first instrument is a temperature controller for a potassium vapor cell used in saturated absorption spectroscopy, giving a temperature stability of 0.01°C over ~100 sampling intervals for a cell heated to 55°C.

The second instrument is a laser frequency controller which reduces the difference in intensity between two laser cavity resonance modes of a Helium-Neon laser and drives a resistive heater to control the length, giving an inferred lower bound to frequency drifts of 6 kHz over 30 sampling times.

Figure 1 – Algorithm Block Diagram. The sampling period is 300 ms.

The input signal from the device under control is amplified and scaled to cover a range matching the input range of the Arduino Uno microcontroller board: 0 to ±5V, maximizing the dynamic range. This analog signal is read by the Arduino using its onboard 10-bit analog-to-digital converter (ADC). Over-sampling of the 10-bit ADC by averaging of 256 samples per input allows us to use an effective 14 bits, resulting in higher input resolution (Ref. 1).

The gain parameters are set using buttons on the LCD display (Fig. 2), and are stored in the Arduino. We have also modified an open-source Arduino PID library (Ref. 2, 3) to use faster integer math. This will facilitate higher-bandwidth feedback applications in the future.

The algorithm converts the input signal and the gain parameters into the required units for the transfer function. The loop gain function then executes within the code, providing the desired output for the controlling device. In the two applications shown in this poster, the output is converted to an 8-bit Pulse With Modulation (PWM) output modulated at 1 kHz. This PWM signal then drives external output circuits, which can be modified to drive a resistive heater, an electromagnet, a servo motor, or another controlling device. The central circuit can also be modified by the addition of a digital-to-analog converter (DAC) or upgrading to an Arduino model with built-in DAC's to provide true analog outputs.

Figure 2 – LCD interface of the temperature controller showing the loop parameters including the temperature setpoint (in °C) and the PID gain parameters (as powers of two). The settings are changed using the buttons in the lower left.

Figure 3 – Simplified schematic of the temperature controller circuit.

We present a temperature control (Fig. 3) driven by the central circuit described in the left column. The PT100 platinum resistance thermometer, unbalanced bridge, and instrumentation amplifier map a temperature range of 0 to 80°C to a voltage range of 0V to 5V. This voltage is read by the ADC on the Arduino.

Since the bridge and ADC are driven by the same V_REF = 5 V, the signal is insensitive to supply fluctuation. Precision, low temperature coefficient components are used for stability and reliability. Over-sampling of the 10-bit analog-to-digital converter (ADC) in the Arduino allows us to use an effective 14 bits, resulting in a temperature resolution of 6 mK.

The output is an 8-bit PWM signal modulated at 1 kHz, which drives a high-side p-channel MOSFET to drive a 15-W resistive heater (Thorlabs HEBS-25-70) attached to the vapor cell. The controller consists of two independent channels, reading two sensors and outputting to two heaters mounted at the two ends of the cell.

Characterization Data

We optimized the stability of the temperature by manually varying the proportional, integral and derivative gain parameters. We measured mean temperature of 54.965°C compared to our setpoint of 55.0°C. The settling time of the controlled circuit is about 300 s (blue curve, Fig. 4a), contrasting with an uncontrolled (1.6 W constant heater output) settling time of about an hour (red curve). To characterize the time constant, we measured the Allan deviation over a 17 hour data run (Fig. 4b).

The Allan deviation σ_a(τ) is defined by

\[ σ_a(τ) = \left( \frac{1}{2} \sum_i (y_i - y_{i-1})^2 \right)^{1/2} \]

where \( y_i \) is the mean of readings taken over the 2^j interval of length \( τ \) in the overall data set.

The highest measured stability is ±0.002°C over 10^-5 interval measurements.

Future Work

We will use the temperature controller circuit to heat a potassium vapor cell for saturated absorption spectroscopy. The circuit and software will be made publicly available soon. The Helium Neon laser will be used to calibrate the wavelength of the experimental laser. Other projects considered include a resistive polarization lens and an electromagnetic levitation track for an optical array.

CONCLUSIONS

This flexible PID controller allows novice users to produce lab-quality instrumentation quickly and with minimal theoretical background. The central circuit and algorithm can be adapted to a wide variety of inputs and outputs. This modular design allows for rapid prototyping and development.

REFERENCES


ACKNOWLEDGEMENTS

We thank Duquesne University for financial and material support, particularly the Bayer School of Natural and Environmental Sciences, the Office of the Provost, the Undergraduate Research Program, and the Faculty Development Fund. Technical help was provided by the Bayer School instrumentation shop.

CONTACT INFORMATION

Ted Corcovilos, Department of Physics, Duquesne University, 600 Forbes Ave. 317 Fisher Hall, Pittsburgh, PA 15282. Email: corcovilos@duq.edu Ph: (412) 396-3973