INFOKUM
Volume 13, Number 04, 2025, DOI 10.58471/infokum.v13i04
ESSN 2722-4635 (Online)
https://infor.seaninstitute.org/index.php/infokum

Optimization of Automatic Pressure Valve System Based on Sensors and Microcontroller For Industrial Secondary Water Tank

Edy Syahputra Saragih[1], Andri Ramadhan[2]
Universitas Al-Azhar, Medan, North Sumatera, Indonesia

Article Info

Keywords:
Pressure control,
smart valve,
microcontroller,
PID control,
and industrial automation.

ABSTRACT

The efficiency and reliability of water distribution systems in industrial environments are highly dependent on the stability of water pressure within secondary water tanks. This study presents the design and optimization of an automatic pressure valve system based on pressure sensors and a microcontroller to regulate and maintain stable pressure levels in an industrial secondary water tank. The system utilizes real-time data from pressure sensors to control the opening and closing of an electric valve through a programmed microcontroller, ensuring responsive and adaptive performance. A PID (Proportional-Integral-Derivative) control algorithm is implemented to fine-tune the valve actuation, thereby reducing pressure fluctuations and improving overall system efficiency. Experimental testing was conducted under various load conditions to evaluate system responsiveness, accuracy, and energy consumption. Results show that the optimized control system significantly improves pressure stability, reduces manual intervention, and enhances operational reliability. This research contributes to the advancement of smart fluid control technologies in industrial applications, promoting more sustainable and automated water management systems.

Corresponding Author:
Andri Ramadhan
Universitas Al-Azhar, Medan, North Sumatera, Indonesia
andriramadhan2@gmail.com

INTRODUCTION

In industrial water distribution systems, maintaining stable water pressure is a critical factor that affects the efficiency and reliability of production processes. The secondary water tank functions as a buffer that supplies water to various industrial units, and thus, the pressure within the tank must be kept within an optimal range to ensure uninterrupted operations. In practice, many systems still rely on manual pressure valve control, which is prone to operator error, pressure fluctuations, and delayed response to changes in demand. With the advancement of automation technology and the Internet of Things (IoT), pressure regulation can now be optimized through the integration of pressure sensors and microcontrollers. Such systems can monitor pressure conditions in real time and automatically adjust valve openings using control algorithms such as Proportional-Integral-Derivative (PID). This approach enhances the system's responsiveness, improves energy efficiency, and reduces maintenance frequency caused by mechanical wear. This study focuses on optimizing an automatic pressure valve system based on sensors and microcontroller technology, specifically designed for use in an industrial secondary water tank. The primary objective is

INFOKUM
Volume 13, Number 04, 2025, DOI 10.58471/infokum.v13i04
ESSN 2722-4635 (Online)
https://infor.seaninstitute.org/index.php/infokum

to design, implement, and evaluate a smart pressure control system capable of maintaining consistent pressure levels under varying load conditions. By adopting intelligent control mechanisms, the system aims to improve operational efficiency, reduce manual intervention, and enhance the reliability of industrial water distribution.

Literature Review

Arduino

Arduino is an open-source electronics prototyping platform based on flexible and easy-to-use hardware and software. The Arduino platform consists of the Arduino board, shield, Arduino processing (programming language), and Arduino Integrated Development Environment (IDE). Arduino boards usually have a basic AtmelAVR ATmega8 microcontroller chip and its derivatives. Shield is a board that can be installed on the Arduino board to increase the capabilities of the Arduino board. Processing is a high-level programming language commonly used to create software embedded in Arduino boards. The Arduino programming language is similar to the C++ programming language, making it easier for users who are used to creating programs with that language. Arduino Integrated Development Environment (IDE) is software used to write and compile or compile programs into binary code for Arduino. Arduino Integrated Development Environment (IDE) is also used to upload compiled programs to the Arduino board's program memory.

The Arduino used to create this final project is the Arduino Uno with type R3. Arduino Uno is an Arduino board that uses the ATmega328 microcontroller. Arduino Uno has 14 digital pins (6 pins can be used as PWM outputs), 6 analog inputs, a 16MHz crystal oscillator, a USB connection, a power source connector, an ICSP header, and a reset button. Arduino Uno contains everything needed to support a microcontroller. Only with connect it to a computer via USB or provide voltage DC from a battery or an AC to DC adapter should do the trick.

Power Source and Voltage Pins

Arduino uno can be powered via USB (Universal Serial Bus) connection or via external power supply. If Arduino uno is connected to both power sources simultaneously then Arduino uno will automatically select one of the power sources to use. Without doing any configuration, once an Arduino board is removed from its packaging box, the Arduino board can be directly connected to a computer via a USB cable. In addition to functioning as a connector for data exchange, this USB cable will also flow 5 Volt DC current to the Arduino board.

When receiving power supply, the power indicator LED on the Arduino board will light up indicating that it is ready to work. On the Arduino Uno board there is a small LED connected to pin no. 13. This LED can be used as an output when a user creates a program and he needs a marker of the program's progress. This is a practical way when the user is testing.

External power supply(not via USB) can come from an AC to DC adapter or a battery. The adapter can be connected to the power socket on the Arduino Uno. If using a battery, the ends of the wires connected to the battery are inserted into the GND and V pins.inwhich is located on the power connector.

INFOKUM
Volume 13, Number 04, 2025, DOI 10.58471/infokum.v13i04
ESSN 2722-4635 (Online)
https://infor.seaninstitute.org/index.php/infokum

Arduino uno can operate at a voltage of 6 Volts to 20 Volts. If the Arduino uno is given a voltage below 7 Volts, then the 5 Volt pin will provide a voltage below 5 Volts and the Arduino uno may work unstable. If given a voltage exceeding 12 Volts, the voltage stabilizer may overheat and damage the Arduino uno.

Memory Map

Arduino Uno is an Arduino board that uses the ATmega328 microcontroller. So the Arduino Uno memory map is the same as the memory map on the ATmega328 microcontroller. ATMega328 has 32 Kbyte On-chip In-System Reprogrammable FlashMemory to store programs. Flash memory is divided into two parts, namely the bootloader and application program parts. The bootloader is a small program that works when the system starts which can insert the entire application program into the processor memory.

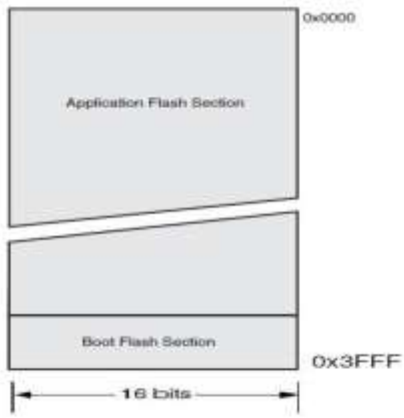

Figure 1. ATMega328 Program Memory Map

The TMega328 data memory is divided into 4 parts, namely 32 locations for general registers, 64 locations for I/O registers, 160 locations for additional I/O registers and the remaining 2048 locations for internal SRAM data. The general register occupies the lower datater address, namely 0x0000 to 0x001F. The I/O register occupies the next 64 addresses starting from 0x0020 to 0x005F. The additional I/O register occupies the next 160 addresses starting from 0x0060 to 0x00FF. The remaining addresses starting from 0x0100 to 0x08FF are used for internal SRAM. The data memory map of the ATMega328 can be seen in Arduino uno consists of 1KByte of EEPROM data memory. In EEPROM memory, data can be written / read back and when the power supply is turned off, the last data written to the EEPROM memory is still stored in this memory, or in other words, EEPROM memory is non-volatile. EEPROM addresses start from 0x000 to 0x3FF.

Generally, the microcontroller on the Arduino board has loaded a small program that will turn it on and blink in a one-second interval. So it is very easy to test whether a new Arduino board is in good condition or not, just connect the board to a computer and see if the power indicator LED is on constantly and the LED with pin 13 is flashing.

Software(Software) Arduino.

Software Arduino that will be used is the Arduino IDE (Integrated Development Environment) software version 1.6.4, although there are still some other software that are

INFOKUM
Volume 13, Number 04, 2025, DOI 10.58471/infokum.v13i04
ESSN 2722-4635 (Online)
https://infor.seaninstitute.org/index.php/infokum

very useful during Arduino development. The Arduino IDE is a very sophisticated software written using Java and C language. Arduino IDE consists of:

1. Editor-program, a window that allows users to write and edit programs in a processing language.
2. Compiler, a module that converts program code (processing language) into binary code. However, a microcontroller will not be able to understand processing language. What a microcontroller can understand is binary code. That is why a compiler is needed in this case.
3. Uploader, a module that loads binary code from a computer into memory on the Arduino board.

Arduino uses programming with the C language. The following is an explanation of the C language characters in the Arduino IDE software. Every Arduino program (usually called a sketch) has 2 (two) functions that must be present, including:/

1. Void setup( ) { }
   All code inside the curly braces will be executed only once when the Arduino program is run for the first time.
2. Void loop( ) { }
   This function will be executed after setup (void setup function) is complete. After running it once, this function will be run again and again continuously until the power supply is removed.

Cooling System (Cooling Water System)

The cooling system is one of the most important systems in a machine because it is used to maintain the temperature of a piece of equipment so that it remains at its nominal working level. The cooling system is installed on the machine so that it does not overheat. In cooling the machine usually uses an air cooling system or a water cooling system. In general, the machine uses a water cooling system. The water cooling system is more difficult and expensive than the air cooling system, but the water cooling system has several advantages, namely that the cooling water is safer because the combustion chamber is surrounded by cooling water (water plus various additives), which also acts as a sound dampener.

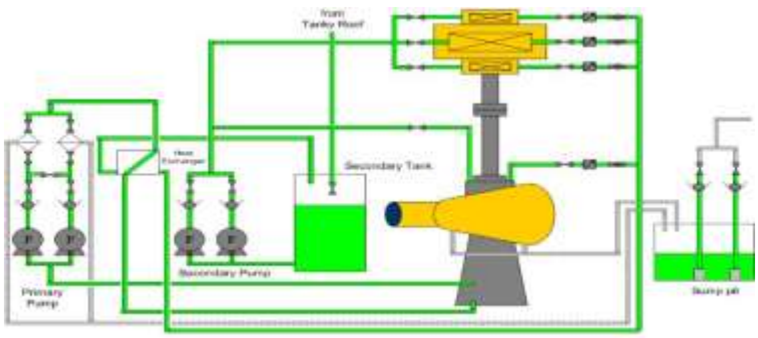

Figure 2. Hydroelectric Power Plant Cooling System

The primary cooler functions to cool the secondary cooler, the primary cooler system is said to be an open cooling system because the water used by the primary cooler is taken from the draft tube, namely river water and discharged back into the river. Equipment found in the primary cooler:

1. Motor
2. Pump
3. Strainer
4. valve
5. Heat ExchangePlate type

The secondary cooler functions to cool the oil in the bearings.

Bearing which is in the UL PLTA Sipansihaporas power plant. The secondary cooling system is said to be a closed system because the water used in the secondary cooling system is taken from the secondary water tank and flows to the pipes where the bearing oil is located and then to the Plate Heat Exchanger to carry out heat exchange and the cooling water returns to *secondary water tank*and reused to cool the oil in the bearings

*bearing*and so on.

1. Motor
2. Pump
3. valve
4. Secondary Water Tank

Temperature sensor

Most temperature sensors have a narrow measurable range and low accuracy but have high costs. The DS18B20 temperature sensor with waterproof capabilities is suitable for measuring temperature in difficult or wet places. Because the output data of this sensor is digital data, we do not need to worry about data degradation when using it for long distances. DS18B20 provides 9 bits to 12 bits of configurable data. Because each DS18B20 sensor has a unique silicon serial number, several DS18B20 sensors can be installed in 1 bus. This allows temperature readings from various places. Although in the datasheet. This sensor can read well up to 125°C, but with a PVC cable cover it is recommended for use not exceeding 100°C.

Temperature Sensor Specifications

a. The voltage required by the sensor is from 3.0V to 5.5V power/data
b. The accuracy is ±0.5°C to -10°C, and -10°C to +85°C
c. Sensor temperature range from -55 to 125°C or -67°F to +257°F
d. provides 9 bit to 12 bit configurable data
e. Using 1 Interface cable and only 1 digital pin for communication.
f. Identity recognition data stored 64 bits
g. Has a warning limit if the temperature is high
h. Temperature-limit alarm system
i. Data entry wait time 750ms
j. Interface cable Red cable: VCC
k. Black cable: GND
l. White cable : DATAStainless Material
m. ess steel cylinder 6mm diameter 35mm long

n. Cable diameter: 4mm
o. Cable length: 90cm

Solenoid Valve

Solenoid valve is a valve that is controlled by electric current either AC or DC through a coil / solenoid. This solenoid valve is the most frequently used control element in fluid systems. Such as in pneumatic systems, hydraulic systems or in machine control systems that require automatic control elements. For example, in a pneumatic system, the solenoid valve is tasked with controlling the pressurized air channel to the pneumatic actuator (cylinder). Or in a water tank that requires a solenoid valve as a water filling regulator, so that the tank does not empty. And various other examples that I cannot possibly explain one by one here.

There are many types of solenoid valves, because this solenoid valve is designed according to its use. Starting from 2 channels, 3 channels, 4 channels and so on. For example, a 2-channel solenoid valve or often called a 2/2 directional control valve. Has 2 types according to how it works, namely NC and NO. So its function is only to close or open the channel because it only has 1 inlet hole and 1 outlet hole. Or on a 3-channel solenoid that has 1 inlet hole, 1 outlet hole, and 1 exhaust. Where the inlet hole functions as the entry of fluid, the outlet hole functions as the exit of fluid and the exhaust functions as the disposal of trapped fluid. And this 3-channel solenoid is usually used or applied to pneumatic actuators (single acting cylinders).

The solenoid valve will work if the coil gets an electric current voltage that matches the working voltage (most solenoid valve working voltages are 100/200VAC and most working voltages on DC voltage are 12/24VDC). And a pin will be attracted by the magnetic force generated from the solenoid coil. And when the pin is pulled up, the fluid will flow from space C to part D quickly. So that the pressure in space C drops and the pressure of the incoming fluid lifts the diaphragm. So that the main valve opens and the fluid flows directly from A to F. To see the use of solenoid valves in pneumatic systems.

1. Level Sensor to detect the height point of solid material that flows continuously, either in the form of grains or powder.
   a. Vibrating pointial solid
   b. Rotating paddle
   c. Admittance type
2. Level Sensor to detect the height point of Liquid material
   a. Pulse-Wave Ultrasonic (Non-Invasive)
   b. Magnetic and mechanical float level measurements
   c. Pneumatic level measurement
   d. Conductive level measurement
3. Sensor Level to measure both, namely to detect height points and monitor solid and liquid material sensors
   a. Ultrasonic level sensor
   b. Capacitance level sensor
   c. Optical interface level sensors

INFOKUM
Volume 13, Number 04, 2025, DOI 10.58471/infokum.v13i04
ESSN 2722-4635 (Online)
https://infor.seaninstitute.org/index.php/infokum

   d. Microwave level sensor
4. Level Sensor to measure the level of liquid continuously
   a. Magnetostrictive level measurement
   b. Resistive chain level measurement
   c. Hydrostatic pressure level measurement
   d. Air bubbler level measurement
   e. Gamma ray level measurement

METHOD

This study was conducted in several stages, including system design, hardware integration, control algorithm development, implementation, and performance evaluation. The research aims to develop and optimize an automatic pressure valve system using pressure sensors and a microcontroller to stabilize the pressure within an industrial secondary water tank. The system consists of three main components:

a. Pressure sensor (e.g., MPX5700 or similar) to measure the real-time water pressure in the tank.
b. Microcontroller unit (e.g., Arduino Uno or ESP32) to process input data and control the valve.
c. Electrically actuated valve to regulate water flow based on control signals from the microcontroller.

The schematic design includes sensor placement in the outlet pipe, valve positioning, and wiring configuration between components. Power supply and communication interfaces were also considered. A PID (Proportional-Integral-Derivative) control algorithm was implemented to manage the valve operation. The PID controller calculates the error between the desired pressure setpoint and the measured pressure, and adjusts the valve accordingly. Tuning of PID parameters (Kp, Ki, Kd) was done using the Ziegler-Nichols method to ensure optimal responsiveness and system stability. The microcontroller was programmed using Arduino IDE. The system was calibrated to recognize safe pressure limits (e.g., 1–3 bar), and the valve was configured to open/close gradually based on sensor feedback. A real-time monitoring interface was developed to display pressure readings and valve status. The prototype was tested under various simulated load conditions, including fluctuating inlet pressure and variable outflow demand. Data collected included:

a. Pressure response time
b. Valve actuation behavior
c. Energy consumption
d. Stability (overshoot and steady-state error)

Each test was repeated multiple times to ensure data accuracy and consistency. Performance metrics were analysed by comparing pressure stability before and after optimization, measuring response time, and evaluating energy savings. The effectiveness of the PID controller was assessed using standard performance indicators such as Rise Time, Settling Time, and Integral of Time-weighted Absolute Error (ITAE).

INFOKUM
Volume 13, Number 04, 2025, DOI 10.58471/infokum.v13i04
ESSN 2722-4635 (Online)
https://infor.seaninstitute.org/index.php/infokum

RESULT

Hardware Testing

Hardware testing is done to find out whether the hardware that has been designed can work or function properly as desired. Testing carried out on hardware includes several hardware circuit blocks that have been designed and also testing a combination of several circuit blocks. The power supply tested here is the power supply used to provide a voltage source to the Arduino, and the voltage required by the Arduino is 6-20 VDC. The output of the power supply used is 12VDC.

Figure 3. Power supply output voltage specifications

Testing on Full Water Level Sensor

Testing on the water level sensor set point full is done to find out whether the level sensor is working or not, testing the level sensor is done by means of continuity on the level sensor terminal, if the level sensor shows full water then the contact child on the level switch will be released (not connected) as seen in Figure 4.

Figure 4. Testing Water Level Sensor

Table 1. Measurement Results of Full Setpoint Air Lever Sensor

Continuity value	Information
0.2 Ohm	Not connected

Testing on Water Level Sensor (Low).

Testing on the low set point water level sensor is carried out with the aim of finding out whether the level sensor is working or not. Testing the level sensor is carried out by means of continuity on the level sensor terminal, if the level sensor indicates low water, the child contact at the switch level will be connected, as can be seen in Figure 5.

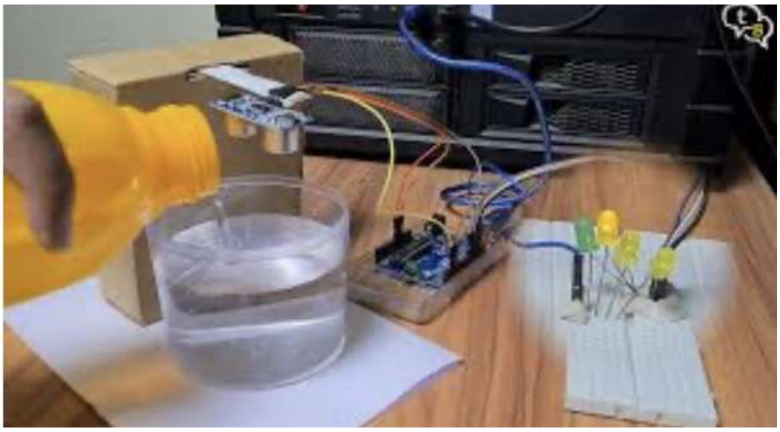

Figure 5. Test Low Water .Low setpoint level sensor measurement

Table 2. Water Level Sensor Setpoint Low Measurement Results

Continuity Value	Information
1.3 Ohm	Connected

Testing Relay Driver Input on Pump

The testing of the Driver Relay input on this pump was carried out to determine how much voltage the Arduino gives to the relay driver on the pump so that the relay driver on the pump can work, and the test results can be seen in Figure 4.8. and how much voltage flows to the relay driver on the pump when the relay driver on the pump is off can be seen in Figure 4.9. After individual testing is carried out on each component by looking at the working voltage on each component, testing is carried out on the work tool as a whole to see whether the work tool works according to the design And after testing the work tool, the tool worked as a whole according to plan, when the water in the tank reads 33℃.So the solenoid valve in and out is working and can also be seen through the indicator light. that is, when the water temperature in the tank is 30℃, the solenoid valve in and out will be off, and when the water level shows "Low" the solenoid valve in will be "ON", and when the water level in the tank shows "full" the solenoid valve in will be "OFF".

CONCLUSION

Based on the results of this study, it can be concluded that the research successfully designed and optimized an automatic pressure valve system based on sensors and a microcontroller for secondary water tanks in industrial environments. The integration of pressure and level sensors with the microcontroller allows for real-time detection and response to changes in tank conditions, enabling the valve to operate with greater accuracy and reliability. The main conclusions of this research are as follows: The sensor and

INFOKUM
Volume 13, Number 04, 2025, DOI 10.58471/infokum.v13i04
ESSN 2722-4635 (Online)
https://infor.seaninstitute.org/index.php/infokum

microcontroller-based system has proven to improve accuracy and response time in pressure regulation compared to conventional manual systems. The optimized system is capable of maintaining tank pressure within the desired operating range, effectively reducing the risk of overpressure and preventing water wastage. The implementation of automated control leads to better operational efficiency and a reduction in maintenance costs in the secondary water supply system. Overall, the use of this smart control system not only enhances the safety and reliability of the system but also supports the realization of industrial automation through stable, efficient, and energy-saving performance.

REFERENCES

Agung, M.Bangun. (2014). *Arduino For Beginners*.

Aryza S. et al (2024) A ROBUST OPTIMIZATION TO DYNAMIC SUPPLIER DECISIONS AND SUPPLY ALLOCATION PROBLEMS IN THE MULTI-RETAIL INDUSTRY. *Eastern-European Journal of Enterprise Technologies*, (3).

Djuandi, Feri. (2011). *Pengenalan Arduino*.

Ramadhan, A., Hasanah, M., & Solahuddin, S. (2023). Pengoptimalan Waktu Terhadap Hasil Adukan Pada Alat Pengaduk Bahan Dasar Sabun Cair. *G-Tech: Jurnal Teknologi Terapan*, *7*(1), 62-69.

S Ardi, R Kusuma. sinergi. (2016). Desain Otomatisai Proses *BVC (BASE VALVE COMPLETE) Assembly Press* Berbasis Kendali *Program Logic Controller.*

Malvio, Albert. 1991. *Prinsip-prinsip Elektronika Dasar*, Jilid 2, Edisi Ketiga. Jakarta : Erlangga.

Ismail, Budiono. 2011. *Dasar Teknik Elektro*, Jilid 1 Malang : Universitas Brawijaya Malang.

Syam, Rafiuddin. 2013. *Dasar Dasar Teknik Sensor*. Fakultas Teknik Universitas Hasanuddin. ISBN 978-979-17225-7-5

M. Putri et al (2021) Desain dan Pengujian Rangkaian Kombinasi Seri-Paralel menggunaan Multisim 14.0. In *Prosiding Seminar Nasional Teknik (Semnastek) UISU* (pp. 238-241).

Zurairah, M., Saragih, E. S., & Misdawati, M. (2021). Comparison Of Thermo Gravimetric Characterization Of Active Carbon Analyzer Rice Husk With Rice Husk Implementation. *Jurnal Matematika Dan Ilmu Pengetahuan Alam LLDikti Wilayah 1 (JUMPA)*, *1*(2), 64-68.