

Alat *Monitoring* Air Melalui Multi Jalur Komunikasi Data (*RS-232, RS-485, CANBus, dan TCP/IP*)

Cesilia Margaritha¹, Alfaiz Ferdiansyah², Ferdito Nois Firmansyah³, Andhika Maulana⁴
Muhammad Syafei Gozali⁵

Politeknik Negeri Batam, *Electrical Engineering*

Instrumentation Engineering Study Program

Jl Ahmad Yani, Batam Centre, Batam 29461, Indonesia

E-mail: syafei@polibatam.ac.id

Abstrak

Terpenuhinya kebutuhan akan air yang memiliki kualitas yang baik, menyebabkan dibutuhkan alat yang dapat memantau kualitas air. Pada penelitian ini kami mencoba membuat alat yang dapat memantau kualitas air dengan empat komunikasi data yang berbeda. Parameter yang digunakan pada penelitian ini adalah tingkat keasaman, kebasaaan, dan kekeruhan suatu air. Pengolahan data dilakukan pada Arduino UNO, dengan menggunakan sensor *pH* seri E-201, dan sensor *turbidity* SEN0189. Range dari sensor *pH* mampu membaca tingkat keasaman, dan kebasaaan mulai dari *pH* 0 – 14. Range pembacaan sensor *turbidity* mampu membaca mulai dari 0 – 1000 *NTU*. Komunikasi data yang digunakan pada penelitian ini ada empat yaitu Serial 232, *RS-485, CANBus, dan TCP/IP*. Hasil pengujian dari kedua sensor yang digunakan, sensor *pH* mampu memiliki tingkat persentase *error* yang cukup rendah jika dibandingkan dengan hasil pengujian dari nilai sensor *turbidity*. Penelitian ini memiliki hasil akhir yaitu pembacaan data sensor menggunakan multi komunikasi data mampu tampil pada aplikasi yang telah dibuat.

Kata Kunci: Air, sensor, komunikasi

Abstract

The need for good quality water is met, resulting in the need for tools that can monitor water quality. In this research we try to create a tool that can monitor water quality with four different data communications. The parameters used in this research are the level of acidity, alkalinity and turbidity of water. Data processing was carried out on Arduino UNO, using the E-201 series pH sensor and the SEN0189 turbidity sensor. The range of the pH sensor is capable of reading acidity and alkalinity levels ranging from pH 0 – 14. The reading range of the turbidity sensor is capable of reading from 0 – 1000 NTU. There are four data communications used in this research, namely Serial 232, RS-485, CANBus, and TCP/IP. The test results of the two sensors used showed that the pH sensor was able to have a fairly low percentage error rate when compared with the test results of the turbidity sensor value. This research has the final result, namely reading sensor data using multiple data communications which can appear in the application that has been created.

Keywords: Water, sensors, communication

1. Pendahuluan

Kualitas air layak guna diperhatikan dari beberapa parameter diantaranya tingkat keasaman (*pH*) dan tingkat kejernihan air. Maka diperlukanlah sistem pemantau kualitas air yang mampu menampilkan data tingkat keasaman dan kejernihan air. Seiring dengan kebutuhan data kualitas air tersebut maka dibutuhkan juga sistem monitoring yang mampu menyediakan jalur pengiriman data yang beragam seperti serial *RS-232, RS-485, canbus dan TCP-IP*.

Penelitian dan pengembangan mengenai sistem monitoring kualitas air telah dilakukan oleh beberapa peneliti, diantaranya (Yunior, Y. T. K., & Kusriani, K. (2021)) dengan penelitian yang berjudul “ Sistem *Monitoring* Kualitas Air pada Budidaya Perikanan Berbasis *IoT* dan Manajemen Data “. Pada penelitian ini dijelaskan bahwasanya penggunaan metode *Internet of Things (IoT)* dapat membantu mempermudah sistem monitoring dan *control* kualitas air sehingga berdampak efisiensi sistem budidaya.

Pembahasan juga meliputi penggunaan parameter *pH*, *turbidity*, dilakukan sebagai identifikasi kadar kualitas air sehingga dapat dijadikan sebagai acuan dan analisa pada pemberian rekomendasi manajemen *control* kualitas air kolam[1]

Penelitian dengan judul “Sistem Monitoring pH dan Kekeruhan Kolam Ikan Koi Berbasis Internet of Things” oleh (Taufik, A., & Fadhil, A.)Menggunakan Aplikasi Blynk“. Menjelaskan tentang fokus untuk merancang sebuah alat yang digunakan untuk memantau *pH* dan kekeruhan air kolam ikan koi menggunakan media terpal, system ini memberi informasi yang sedang berlangsung melalui aplikasi *Blynk*. Hasil akhir pada system ini bahwa alat dapat memantau *pH* dan kekeruhan secara otomatis dan tepat. Persentase error yang diperoleh sensor *pH* adalah 0.6853%, dan persentase error pada sensor kekeruhan sebesar 3,704% [2]

Penelitian dengan judul “Pengukur Tingkat Kekeruhan Keasaman Dan Suhu Air Menggunakan Mikrokontroler Atmega328p Berbasis Android “. (Rufiyanto, A., Abdilah, G. S., & Purwaningrum, S. D.)membuat sistem alat pendeteksi air yang berfokus pada parameter kekeruhan air yang dideteksi oleh sensor, keasaman dan kebasaaan air diukur menggunakan sensor *pH*. Parameter yang diolah akan diproses melalui mikrokontroler, dengan berbasis program arduino ide, untuk menghubungkan mikro dengan smartphone android, maka modul bluetooth dipasang pada mikro. Aplikasi android yang digunakan untuk menampilkan data dari alat ukur ke smartphone dibuat di App inventor. Penulis menyimpulkan bahwa alat ukur yang dibuat, membuat pengukuran lebih efisien.[3]

Berdasarkan latar belakang melalui penelitian diatas penulis bermaksud merancang sebuah alat yang dapat memonitoring air dengan software interface, penelitian ini berjudul “Alat Monitoring Air dengan Menggunakan Multi Jalur Komunikasi Data (Serial 232, RS-485, CANBus, dan TCP/IP). Dengan tujuan alat ini dapat memonitoring kualitas air dengan jarak,

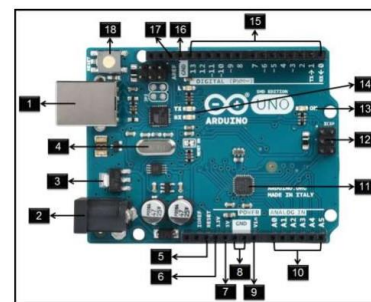
dengan parameter keasaman, kebasaaan, dan kekeruhan.

2. Tinjauan Pustaka

Penelitian ini mempunyai beberapa alat atau komponen yang digunakan yang akan dijelaskan diantaranya:

1. Arduino

Arduino Uno adalah salah satu produk elektroniknya memiliki mikrokontroler ATmega 328. Mikrokontroler dapat digunakan, dengan menghubungkan *board* arduino Uno ke *computer* dengan menggunakan kabel *USB* untuk menjalankannya. Spesifikasi Arduino Uno memiliki 6 pin analog, dan 14 pin digital. Arduino Uno berfungsi membuat program untuk mengendalikan berbagai komponen elektronika. Arduino dengan komputer atau power bank lewat koneksi *USB* sebagai *supply* listrik[4]



Gambar 1 Arduino Uno

2. Sensor pH

Sensor yang sudah dilengkapi dengan *probe* dan modul *converter* dengan keluaran antarmuka data *analog*, sehingga bisa dihubungkan dengan *board* Arduino UNO. Elektroda kaca berfungsi untuk merubah jumlah ion yang ada dalam larutan dan elektroda referensi berfungsi untuk merubah jumlah ion yang terbaca oleh elektroda kaca menjadi nilai tegangan analog. Dengan prinsip kerja yaitu semakin banyak elektron yang terdeteksi pada sampel maka semakin bernilai pula cairan tersebut, dan apabila semakin sedikit elektron yang terdeteksi maka sampel cairan tersebut bersifat basa[5]. Sensor *pH* memiliki rentang pengukuran 0.00-14.00 *PH*. Modul *pH* sensor memiliki spesifikasi tegangan 0V– 5V, dengan *output* pin analog. Gambar 1 merupakan kelengkapan modul sensor *pH*, berikut adalah penjelasan komponen dari sensor *pH* tersebut:



Gambar 2 Sensor pH

3. Sensor Turbidity

Sensor kekeruhan mendeteksi air dengan mengukur tingkat kekeruhan, sensor ini menggunakan cahaya untuk mendeteksi partikel tersuspensi dalam air dengan mengukur transmisi cahaya yang berubah dengan jumlah total padatan tersuspensi (TSS) dalam air [6]. Sensor ini menyediakan mode keluaran sinyal *analog* dan *digital*. Keluaran sinyal analog, nilai keluaran akan menurun saat berada dalam cairan dengan kekeruhan tinggi. Sensor ini memiliki spesifikasi tegangan keluaran 0-4 - 5V, dan memiliki range pembacaan sensor yaitu 0 – 1000 NTU.



Gambar 3 Sensor Turbidity

4. RS-232

Komunikasi Serial-232 adalah standar komunikasi serial yang didefinisikan sebagai antarmuka antara perangkat terminal data, dan perangkat komunikasi data [7]. Berdasarkan spesifikasi *serial-232* memiliki kecepatan 256 kbps atau lebih rendah dengan jarak kurang dari 15 meter, komunikasi *RS-485* dapat digunakan untuk komunikasi data antara dua komputer secara langsung.

5. RS485 Shield

RS-485 adalah teknik komunikasi data serial, komunikasi data yang dapat dilakukan pada jarak yang cukup jauh yaitu 1.2 Km [8]. *RS-485 shield* mampu menghubungkan beberapa penerima dalam satu

pengirim. Kecepatan transmisi dari RS-485 dapat diukur dalam *baudrate*. Banyak faktor yang mempengaruhi kecepatan maksimum yang dapat dicapai, termasuk kualitas kabel, dan kemampuan perangkat keras yang terhubung dengan komunikasi.

6. TCP/IP

TCP/IP adalah rangkaian protocol komunikasi yang digunakan untuk menghubungkan perangkat jaringan di internet [9]. *TCP* mendefinisikan bagaimana aplikasi dapat membuat saluran komunikasi diseluruh jaringan. *IP* memberi pengertian cara menangani setiap paket untuk memastikan paket mencapai tujuan yang benar.

7. CAN Bus Shield

CANBus dapat digunakan untuk komunikasi data multi point sama seperti EIA-485 [10]. *CANBus* bisa terdapat lebih dari dua perangkat yang dapat bertukar data. Tidak ada *master* dan *slave* pada komunikasi *CANBus*, semua perangkat yang terhubung dalam satu jaringan disebut dengan node, dan masing-masing node dapat berkomunikasi sesama node lainnya selama masih dalam satu jaringan yang sama

8. LCD

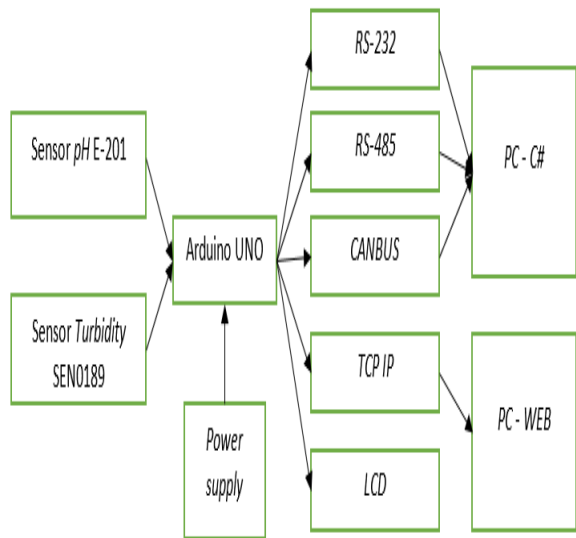
LCD I2C adalah *module LCD* yang dikendalikan secara serial sinkron dengan protocol *I2C* [11]. Arduino sendiri sudah mendukung protocol *I2C*, pada papan Arduino UNO, port *I2C* terletak pada pin A4 untuk jalur SDA (serial data), dan pin A5 untuk jalur SCL (serial clock). Untuk sisi *software*, Arduino membantu bekerja dengan protocol ini melalui *library* 'Wire.h'. *Library LiquidCrystal_I2C.h* akan dimanfaatkan untuk mengkonversi jalur *paralel LCD* menjadi jalur serial *I2C*.

3. Metode Penelitian

Dalam sistem ini, digunakan dua sensor yaitu pH dan turbidity, yang terhubung kepada mikrokontroler Arduino UNO. Mikro berperan sebagai mengumpulkan data dari kedua sensor dan mengirimkannya melalui beberapa komunikasi yang digunakan. Dalam komunikasi serial 232, data dari sensor dikirimkan melalui koneksi serial 232,

demikian pada komunikasi *RS-485*, dan *CANBus*, data sensor dikirimkan melalui *protocol* masing – masing komunikasi. Selanjutnya data ditampilkan pada *interface*, program yang digunakan adalah program *C#*, program *C#* dapat membaca data dari Arduino UNO melalui koneksi serial 232, *RS-485*, dan *CANBus*.

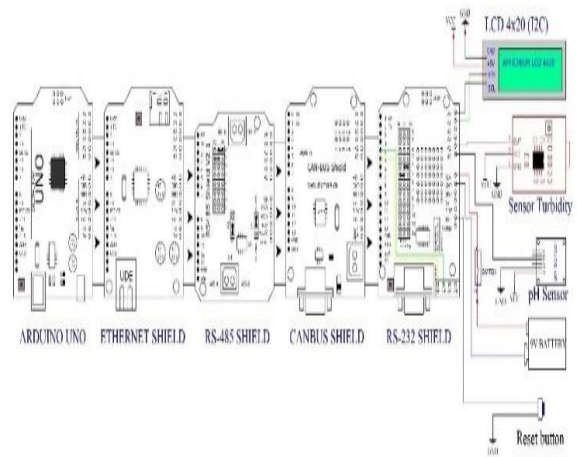
Data juga dapat diakses melalui jaringan protocol TCP/IP, tampilan *web* akan memberikan akses ke data sensor dari perangkat yang terhubung kepada jaringan. Sistem ini juga dilengkapi dengan tombol *reset* guna memungkinkan dalam melakukan *reset* pada *system*, tombol *reset* dapat digunakan untuk memulai ulang mikrokontroler Arduino UNO. Sistem ini diberi sumber oleh baterai 9 volt.



Gambar 4 Blok Diagram Sistem

3.1 Perancangan Wiring Elektrikal

Perancangan elektrikal, dimulai dari mikrokontroler yang dihubungkan dengan keempat komunikasi. Komunikasi yang berada dipaling atas, merupakan komunikasi yang terhubung kepada *LCD*, sensor, dan tombol *reset*, komunikasi tersebut merupakan komunikasi serial 232. Perancangan elektrikal menunjukkan terhubungnya komunikasi dengan dua buah sensor, dan satu buah tombol *reset*, tombol *reset* bertujuan untuk mengulang kembali suatu proses. Masing – masing dari komunikasi memiliki kabel keluarannya.



Gambar 5 Wiring Electrical

Tabel 1 Wiring Schedule

Board Arduino UNO	Board Ethernet Shield	Board RS-485 Shield	Board CANBus Shield	Board RS-232 Shield
5V	-	VCC	VCC	VCC
GND	GND	GND	GND	GND
3,3	VCC	-	-	-
Digital pin 2	-	TXD	-	-
Digital pin 3	-	RXD	-	-
Digital pin 4	-	-	-	RXD
Digital pin 5	-	-	-	TXD
Digital pin 10	CS	-	CS-	-
Digital pin 11	SI (MOSI)	-	SI (MOSI)	-
Digital pin 12	SO (MISO)	-	SO (MISO)	-
Digital pin 13	SCK	-	SCK	-

Tabel 2 Wiring Schedule

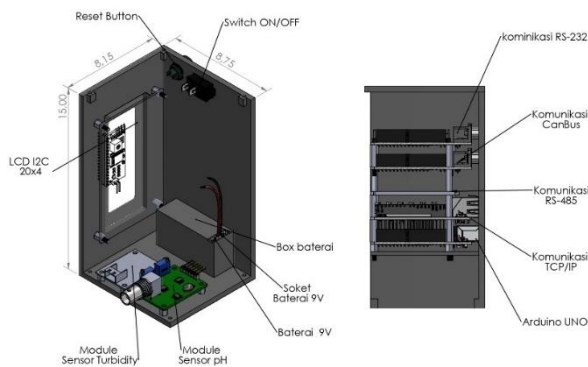
Arduino UNO	Sensor Turbidity	Sensor pH	Battery 9V	Reset Button	LCD I2C 20x4
5V	VCC	V+	(+)	-	5V
GND	GND	G	(-)	(-)	GND
VIN	-	-	-	-	-
RESET	-	-	-	(+)	-
Analog pin 0	-	PO	-	-	-
Analog pin 1	OUT	-	-	-	-
Analog pin 4	-	-	-	-	SDA
Analog pin 5	-	-	-	-	SCL

Tabel diatas merupakan tabel *wiring schedule*, dibuat guna untuk menyediakan panduan yang jelas akan pengaturan kabel dan konektor. Hal ini membantu memastikan bahwa setiap kabel terhubung dengan

benar, dan dapat meminimalisir kesalahan dalam pemasangan rangkaian, serta memungkinkan untuk mempermudah dalam perbaikan.

3.2 Perancangan Mekanikal

Pada desain mekanikal diatas, ukuran cover alat adalah 15 x 8,15 cm, dan pada cover alat ini kami membaginya menjadi 2 bagian agar memudahkan saat melakukan perakitan alat, pada bagian cover pertama terdapat 4 buah komunikasi, dan pada bagian cover yang kedua terdapat sensor *pH* dan *turbidity*, *battery* 9v, *LCD* 20x4, dan pada tiap komponen dipasang spacer 0,5 cm pada bagian bawah, tujuannya adalah agar tiap komponen saat dilakukan perakitan bisa sejajar, pada penutupan alat dengan menyatukan 2 bagian dengan baut M3x7.



Gambar 6 Design Mekanikal

3.2 Perancangan Software

Desain tampilan harus dapat memastikan bahwa informasi yang ditampilkan dapat dibaca dan dipahami oleh pengguna. Pada alat monitoring air, dengan multi komunikasi, tampilan dirancang agar dapat menampilkan data dari berbagai sumber komunikasi.

4 Hasil dan Pembahasan

Dalam pengujian akuisisi data, ada dua sensor yang akan diakuisisi datanya, yaitu sensor *pH*, dan sensor *turbidity*, akuisisi data sensor menggunakan Arduino UNO sebagai perangkat keras, dan Arduino IDE sebagai perangkat lunak,

4.1 Hasil Pengujian Sensor *pH* Menggunakan Arduino Uno

Akuisisi data sensor *pH*, pada pengujian akuisisi data pada sensor *pH*, penelitian ini menggunakan alat pembanding sensor yaitu *pH* meter, *pH* meter disini

berfungsi sebagai alat untuk membandingkan nilai, dengan sensor pada sistem yang dibuat. Selanjutnya data akan tampil pada serial monitor arduino IDE, menggunakan *source code* yang telah disediakan, ketika data tampil akan menunggu banyaknya data, sesuai dengan kebutuhan, *delay* yang digunakan adalah satu detik, jadi untuk dua puluh data yang dibutuhkan, perlu menunggu kurang lebih dua puluh detik untuk dapat menyalin data pada serial monitor Arduino IDE. Setelah data disalin dari serial monitor, pengolahan data dilakukan pada software Excel. Pada tabel dibawah ini, ditunjukkan hasil pengukuran dari *pH* meter, dan persentase error yang telah didapatkan melalui rumus persentase error.

Tabel 3 Pengujian Sensor *pH*

pH Meter	Sensor pH	Tegangan (mV)	Error %
1,3	1,04	3420	20,00
2,33	2,31	3240	0,86
3,73	3,43	3081	8,04
4,16	4,04	2994	2,88
6,65	7,52	2501	13,08
7,19	7,19	2549	0,00
7,3	7,41	2518	1,51
7,59	7,69	2478	1,32
7,95	8,01	2432	0,75
9,17	8,96	2298	2,29
9,19	9,53	2216	3,70
10,17	10,35	2100	1,77
13,11	13,35	1676	1,83
13,31	13,77	1615	3,46

4.2 Hasil Pengujian Sensor *Turbidity*

Pengambilan data oleh sensor *turbidity* menggunakan sampel air keruh yaitu teh tarik dilakukan sebanyak 18 kali. Tiap ganti satu kali percobaan, diambil 5ml air keruh, dan sebagai gantinya dimasukkan air bersih ke wadah sebanyak 5ml. Selain menggunakan sensor *turbidity*, digunakan juga *turbidity* meter, sebagai pembanding, untuk membandingkan dengan sensor *turbidity* yang dibuat. Berikut merupakan tabel dari pengambilan data yang dilakukan

Tabel 4 Pengambilan data sensor *turbidity*

Percobaan	<i>Turbidity</i> Meter	Rata - rata <i>Turbidity</i> Sensor	Error (%)
1	1009	1042,69	3,34
2	938	854,36	8,92
3	796	685,27	13,91
4	644	716,72	11,29
5	511	565,52	10,67
6	387	463,83	19,85

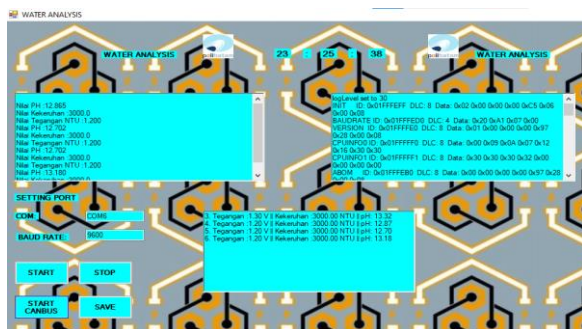
7	296	275,17	7,04
8	208	177,99	14,43
9	152	146,71	3,48
10	101	121,8	20,59
11	76	88,85	16,91
12	60	69,95	16,58
13	46	50,21	9,15
14	35,73	32,98	7,70
15	28,14	17,43	38,06
16	21,75	4,72	78,30
17	18,6	13,41	27,90
18	20,37	16,59	18,56

4.3 Pengujian Menggunakan Komunikasi dengan Hasil Data Sensor Tampil Pada Aplikasi

Tahap selanjutnya, masuk pada proses pengujian alat. Bagian ini akan menjelaskan hasil pengukuran dari sensor *pH* dan *turbidity* yang dikomunikasikan melalui berbagai metode seperti Serial 232, *RS-485*, *CANBus*, dan *TCP/IP*, serta bagaimana tampilan data disajikan pada aplikasi *C#*, dan *web*. Pembahasan ini memberikan informasi tentang kesiapan alat dalam memantau kualitas air menggunakan komunikasi.

1. Hasil Pengujian Data Sensor *pH* dan *Turbidity* Menggunakan Komunikasi *CANBus*

Gambar dibawah ini menunjukkan tampilan aplikasi *C#*, menggunakan komunikasi *CANBus*. Terdapat kabel yang kompatibel dengan komunikasi *CANBus*. Data tampil secara *real time* pada aplikasi yang digunakan.



Gambar 7 Tampilan pada *C#* menggunakan komunikasi *CANBus*

2. Hasil Pengujian Data Sensor *pH* dan *Turbidity* Menggunakan Komunikasi *TCP/IP*

Terdapat kabel *LAN* berwarna kuning yang merupakan

kabel yang kompatibel dengan komunikasi shield *TCP/IP*. Data tampil secara *real time*, namun tidak sejalan dengan data yang tampil pada *LCD*.



Gambar 8 Tampilan Web Menggunakan Komunikasi *TCP/IP*

3. Hasil Pengujian Data Sensor *pH* dan *Turbidity* Menggunakan Komunikasi Serial 232

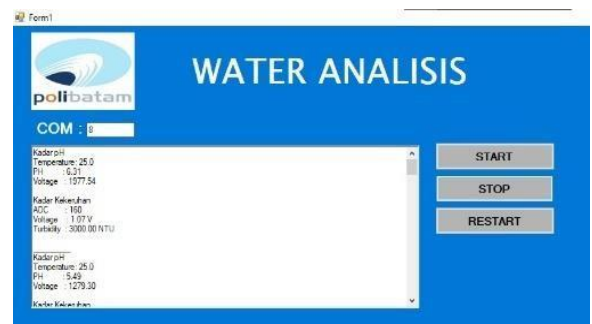
Gambar berikut ini, merupakan gambar dari tampilan pada aplikasi *C#*, yang dihubungkan melalui kabel *USB* yang sesuai dengan Serial 232. Tampilan data yang ada pada *display C#* adalah tampilan data sensor *pH* dan *turbidity*.



Gambar 9 Tampilan Komunikasi Serial 232

4. Hasil Pengujian Data Sensor *pH* dan *Turbidity* Menggunakan Komunikasi *RS485*

Pada gambar dibawah, merupakan tampilan pada aplikasi *C#*, yang dihubungkan melalui kabel *USB RS-485*.



Gambar 10 Tampilan Komunikasi *RS-485*

5 Kesimpulan

Berdasarkan dengan penelitian yang sudah dilakukan, sub bab kesimpulan akan membawa kita pada titik akhir dari penelitian yang berjudul “Alat Monitoring Air dengan Menggunakan Jalur Multi Komunikasi Data“ Melalui setiap kegiatan, kami berupaya untuk membuat kesimpulan, berikut merupakan kesimpulan dari penelitian ini:

- A. Mikrokontroler Arduino Uno digunakan sebagai perangkat keras, dan Arduino IDE sebagai perangkat lunak untuk melakukan akuisisi data. Dengan rata – rata error yang didapatkan dari masing – masing sensor, dengan pendekatan regresi linear. Pada sensor *pH* didapatkan rata – rata error sebesar 4.39 %, dan sensor *turbidity* dengan rata – rata error sebesar 18.15 %.
- B. Data sensor telah terkirim melalui komunikasi data seperti Serial 232, *RS-485*, *CANBus*, dan *TCP/IP*. Pada komunikasi data Serial 232, *RS-485*, *CANBus* memiliki *COM* yang terdaftar pada perangkat penerima, dan komunikasi data *TCP/IP* pada *IP statis* (192.168.137.177)
- C. Tampilan hasil data pengukuran sensor *pH*, dan sensor *turbidity* tersaji pada *LCD*, aplikasi web, dan *C#* untuk setiap tipe komunikasi data.
- D. Pembacaan sensor telah dilengkapi dengan proses kalibrasi dengan pendekatan regresi linear untuk tipe sensor yang berbeda. Dengan persamaan linear pada sensor *pH* yaitu $y = -2.7342x + 13,489$, sedangkan pada sensor *turbidity* menggunakan persamaan $y = -0.5133x + 2062.1$

Daftar Pustaka

- [1] Yunior, Y. T. K., & Kusriani, K. (2021). Sistem Monitoring Kualitas Air Pada Budidaya Perikanan Berbasis IoT dan Manajemen Data. *Creative Information Technology Journal*, 6(2), 153-164. protokol TCP/IP. *Transient: Jurnal Ilmiah Teknik Elektro*, 2(3), pp.643- 650.
- [2] Taufik, A., & Fadhil, A. (2023). Sistem Monitoring pH dan Kekeruhan Kolam Ikan Koi Berbasis Internet of Things Menggunakan Aplikasi Blynk. *Jurnal Teknologi Elektro*, 14(01).
- [3] Zarkashie, M. F. (2021). Rancang bangun sistem

pengukuran kualitas air Untuk keperluan higiene sanitasi berbasis Arduino uno (Bachelor's thesis, Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta).

- [4] Ikram sufi, 2022, Mengenal Bagian - Bagian dan Features pada Papan Arduino UNO beserta Fungsinya Yuk
- [5] Wibi Hiryanto, (2021), Sistem Monitoring Kejernihan dan *pH* Air Berbasis Arduino
- [6] Mohammad Alfian Ikhsan, (2018), PENDETEKSI KEKERUHAN AIR DI TANDON RUMAH BERBASIS ARDUINO UNO
- [7] Jason Yang, (2019), Bagaimana RS232 Komunikasi Serial Bekerja
- [8] Nurul Septiani, (2024) , Sistem Komunikasi Antar Arduino Menggunakan Potokol RS-485
- [9] Kinda Yasar, (2024), Apa itu TCP/IP dan Bagaimana Cara Kerjanya?
- [10] SIMOR Technology, (2021), apa itu canbus?bisa kah arduino menggunakan protokol ini?
- [11] Hery Suryantoro, (2019), PROTOTYPE SISTEM MONITORING LEVEL AIR BERBASIS LABVIEW & ARDUINO SEBAGAI SARANA PENDUKUNG PRAKTIKUM INSTRUMENTASI SISTEM KENDALI