

Accessing Ping Sonar Echosounder Produksi ROVMAKER pada Mikrokontroler dan *Data Noise Reduction* dengan Kalman Filter

Adam Maulana¹, Joko Endrasmono², Zindhu Maulana Ahmad Putra³, Lilik Subiyanto⁴, Mohammad Basuki Rahmat⁵, M.Khoirul Hasin⁶, Isa Rachman⁷, Agus Khumaidi⁸, Yuning Widiarti⁹, Ryan Yudha Adhitya¹⁰, Dimas Pristovani Riananda¹¹

e-mail: adammaulana24@student.ppns.ac.id, endrasmono@ppns.ac.id, zindhu@ppns.ac.id, llksubiyanto@gmail.com, mbasuki.rahmat@ppns.ac.id, khoirul.hasin@ppns.ac.id, isarachman@ppns.ac.id, aguskhumaidi@ppns.ac.id, yuning.widiarti@ppns.ac.id, ryanyudhaadhitya@ppns.ac.id, dimaspristovani@ppns.ac.id

^{1,2,5,6,7,8,10} Jurusan Teknik Otomasi, Politeknik Perkapalan Negeri Surabaya, Jalan Teknik Kimia Kampus ITS, Surabaya, Indonesia
^{3,4,9,11} Jurusan Teknik Kelistrikan Kapal, Politeknik Perkapalan Negeri Surabaya, Jalan Teknik Kimia Kampus ITS, Surabaya, Indonesia

Informasi Artikel

Riwayat Artikel

Diterima 15 Mei 2024

Direvisi 28 Mei 2024

Diterbitkan 31 Mei 2024

Kata kunci:

Survei Batimetri
Ping Sonar Echosounder
Kalman Filter
Noise Reduction

Keywords:

Bathymetric Survey
Ping Sonar Echosounder
Kalman Filter
Noise Reduction

ABSTRAK

Dua pertiga wilayah Indonesia merupakan perairan yang memiliki risiko banjir yang tinggi. Survei batimetri penting untuk pengelolaan banjir karena menyediakan data kedalaman dan topografi sebagai penentu kebutuhan pengerukan terhadap suatu wilayah perairan. Survei batimetri tradisional menghasilkan cakupan yang kurang optimal dikarenakan pergerakan kapal yang terbatas. Penelitian ini bertujuan untuk mengembangkan sistem batimetri otomatis menggunakan sensor ping sonar echosounder yang terintegrasi dengan USV (*Unmanned Surface Vehicle*). Ping Sonar echosounder produksi ROVMAKER umumnya hanya dapat diakses menggunakan *software* bawaan dari produsen ROVMAKER dan kini dapat terintegrasi dengan sistem mikrokontroler pada USV untuk meningkatkan efisiensi kinerja sistem batimetri. Pada penelitian ini juga menggunakan metode kalman filter untuk mengurangi *noise* pembacaan sensor. Validasi kelayakan sensor menunjukkan rata-rata *error* 2,94% pada kedalaman 1400mm dan 2,195% pada kedalaman 2200mm yang menunjukkan bahwa sensor tergolong layak digunakan. *Noise* yang tereduksi optimal pada eksperimen kalman filter yaitu pada rasio varian (R/Q) 1000 dengan RMSE 5,05mm pada kedalaman 1400mm dan 3,99mm pada kedalaman 2200mm. Sistem ini terbukti dapat meningkatkan keakuratan data batimetri dan efisiensi pengaksesan sistem.

ABSTRACT

Two-thirds of Indonesia's territory is waters that have a high risk of flooding. Bathymetric surveys are important for flood management because they provide depth and topography data to determine the need for dredging of a water area. Traditional bathymetric surveys produce less than optimal coverage due to limited ship movements. This research aims to develop an automatic bathymetry system using a ping sonar echosounder sensor integrated with a USV (Unmanned Surface Vehicle). The ping sonar echosounder produced by ROVMAKER can generally only be accessed using the built-in software from the ROVMAKER manufacturer and can now be integrated with the microcontroller system on the USV to increase the efficiency of bathymetric system performance. This research also uses the Kalman filter method to reduce sensor reading noise. Validation of the suitability of the sensor shows an average error of 2.94% at a depth of 1400mm and 2.195% at a depth of 2200mm, which shows that the sensor is classified as suitable for use. The optimal

p-ISSN: 2356-0533; e-ISSN: 2355-9195



9 772356 053009

noise reduction in the Kalman filter experiment is at a variance ratio (R/Q) of 1000 with an RMSE of 5.05mm at a depth of 1400mm and 3.99mm at a depth of 2200mm. This system has been proven to increase the accuracy of bathymetric data and system access efficiency.

Penulis Korespondensi:

Adam Maulana,
Jurusan Teknik Otomasi,
Politeknik Perkapalan Negeri Surabaya,
Jalan Teknik Kimia Kampus ITS, Surabaya, Indonesia,
Email: adammaulana24@student.ppns.ac.id
Nomor HP/WA aktif: +6282139823139

1. PENDAHULUAN

Indonesia merupakan negara maritim yang memiliki luas wilayah perairan 2/3 dari seluruh luas wilayahnya. [1] Dengan luasnya wilayah perairan dan daerah aliran sungai (DAS), Indonesia memiliki potensi bencana banjir yang cukup tinggi terutama pada musim penghujan [2], [3], [4]. Bencana banjir disebabkan dimana daerah tampungan air hujan seperti sungai dan waduk tidak lagi berfungsi dengan baik sebagai sistem drainase [5]. Sungai dan waduk yang dangkal tidak akan mampu menampung volume air dalam jumlah besar sehingga air akan meluap menggenangi wilayah sekitarnya [6].

Sebagai upaya penanggulangan bencana banjir maka perlu diadakan survei batimetri secara berkala. Survei batimetri bertujuan untuk memperoleh data mengenai kedalaman atau topografi dasar perairan, termasuk letak dan luas obyek-obyek yang ada di dalamnya [7] serta data-data yang diperoleh menjadi acuan perlu atau tidaknya dilakukan pengerukan tanah di dasar perairan. Survei batimetri konvensional memanfaatkan metode akustik menggunakan perangkat echosounder [8] sebagai pengukur kedalaman perairan yang terpasang pada kapal dan hanya dapat dioperasikan secara manual oleh surveyor. Salah satu permasalahan dalam survei secara konvensional adalah keterbatasan area survei batimetri karena kemampuan gerak kapal yang terbatas sehingga hasil dari survei kurang optimal [9], [10].

Berdasarkan permasalahan diatas, maka diperlukan sebuah perangkat survei yang dapat dioperasikan secara otomatis dan dapat dikontrol jarak jauh. Hal tersebut dapat diwujudkan dengan memanfaatkan sensor ping sonar echosounder yang terpasang pada *unmanned surface vehicle* atau kapal tanpa awak. Pada penelitian ini sensor ping sonar echosounder yang digunakan yaitu ping sonar produksi ROVMAKER yang memiliki harga lebih terjangkau dari para pesaingnya. Namun sensor ini hanya dapat diakses menggunakan *software* Underwater Explorer yang disediakan oleh pihak ROVMAKER dan tidak bersifat *open source* sehingga jarang digunakan oleh konsumen. Maka dari itu peneliti melakukan penelitian terhadap pengaksesan sensor ping sonar echosounder produksi ROVMAKER menggunakan mikrokontroler agar sensor dapat terintegrasi dengan sistem kontrol *unmanned surface vehicle*.

Pada lokasi survei batimetri, kondisi perairan tidak selalu kondusif dan dapat dipastikan terdapat faktor gangguan eksternal yang dapat menyebabkan hasil pembacaan sensor ping sonar echosounder memiliki *noise* sehingga mengurangi tingkat akurasi pada hasil akhir survey batimetri. Dalam survei batimetri *noise* umumnya disebabkan oleh berbagai faktor seperti getaran dari motor penggerak USV, gelombang perairan, posisi sudut kemiringan sensor terhadap permukaan dasar perairan dan faktor lainnya. Maka diperlukan *filtering* terhadap data hasil pembacaan sensor ping sonar echosounder untuk mengurangi *noise*.

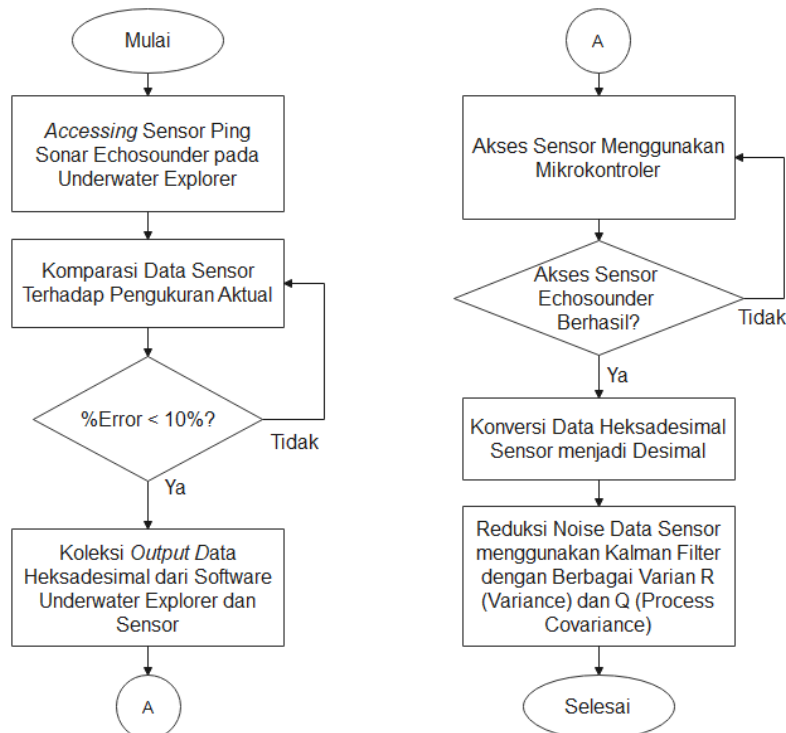
Sensor dengan sensitivitas tinggi seperti ping sonar echosounder cenderung rentan terhadap *noise* saat mengukur data dari lingkungan [11]. Namun, meskipun rentan terhadap *noise*, sensor dengan sensitivitas tinggi



sangat dibutuhkan untuk menunjang performa kinerja sistem yang lebih responsif [12]. *Noise* pada hasil pembacaan sensor seperti ping sonar echosounder dapat direduksi menggunakan metode Kalman Filter [13]. Kalman Filter (KF) merupakan salah satu metode yang mampu memprediksi data yang akan diperoleh berdasarkan data yang telah diterima sebelumnya [14] serta dapat mereduksi *noise* pada sinyal pembacaan sensor [15]. Kalman filter populer karena memiliki keunggulan yaitu memiliki komputasi yang praktis, kebutuhan memori yang rendah serta kemampuan yang handal dalam mengatasi *noise*. Metode ini menerapkan teknik estimasi keadaan (*state technique estimation*) yang dapat mengambil informasi dari data yang terpengaruh oleh *noise* [16]. Selain itu, Kalman Filter juga dikenal karena fleksibilitasnya dalam berbagai situasi dan kemampuannya beradaptasi terhadap perubahan lingkungan yang dinamis, menjadikannya alat yang sangat berguna tidak hanya dalam industri dan penelitian, tetapi juga dalam teknologi konsumen seperti sistem navigasi GPS. *automation* dan robotika, serta prediksi kondisi suatu lingkungan secara *real-time* yang membutuhkan ketepatan tinggi dan keandalan dalam kondisi yang dinamis.

2. METODE PENELITIAN

2.1 Alur Penelitian



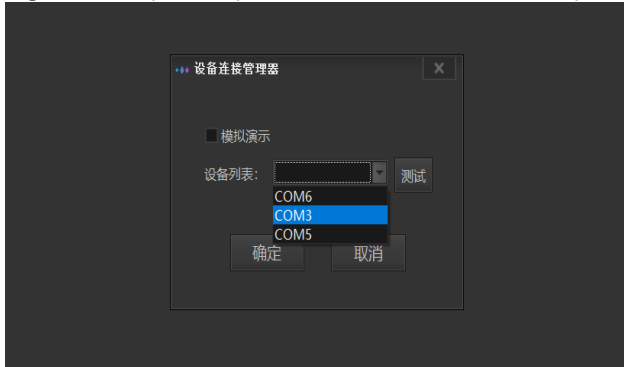
Gambar 1: Flowchart Pengaksesan Echosounder menggunakan Mikrokontroler

Penelitian ini diawali dengan mengakses sensor menggunakan *software* Underwater Explorer untuk memahami fungsi dan cara kerjanya. Sensor dapat dikatakan layak apabila memiliki rata-rata *%error* dibawah 10% dikomparasi dengan pengukuran aktual menggunakan meteran. Selanjutnya dilakukan pengkoleksian *output* data heksadesimal yang bersumber dari Underwater Explorer dan hasil pembacaan sensor untuk keperluan *accessing* sensor dan konversi data sensor menggunakan mikrokontroler. Kemudian diterapkan metode kalman filter pada data sensor dengan berbagai varian R (*Variance*) dan Q (*Process Covariance*) untuk mendapatkan respon terbaik dalam mereduksi *noise*.



2.1 Accessing Sensor Ping Sonar Echosounder pada Underwater Explorer

Akses sensor ping sonar echosounder dimulai dengan *connect device* sensor pada *software* kemudian mengkonfigurasi tipe pembacaan sensor menggunakan mode *non-wave* yang hanya menampilkan pengukuran kedalaman tanpa menampilkan grafik kontur. Pada *toolbar* terdapat fitur *run* dan *stop* untuk memulai dan menghentikan proses pembacaan data kedalaman perairan.



(a)



(b)

Gambar 2: Konfigurasi "Connect Device" (a) dan "Read Mode Non-Wave" (b)

2.2 Komparasi Data Sensor terhadap Pengukuran Aktual

Komparasi data sensor terhadap pengukuran aktual menggunakan meteran dilakukan di media kolam renang yang memiliki standar kedalaman tertentu agar data yang didapat tidak rancu. Tahap ini melibatkan beberapa kondisi kedalaman kolam renang yaitu pada kedalaman 1.400m atau 1400mm dan 2.200m atau 2200mm.



(a)



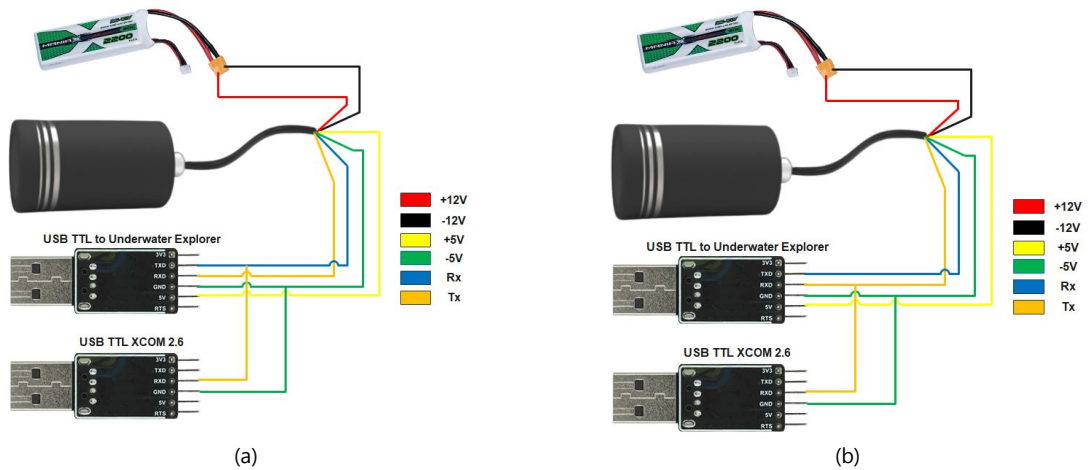
(b)

Gambar 3: Pengukuran Aktual (a) Pengukuran menggunakan Sensor (b)

2.3 Pengumpulan Data Heksadesimal *Output* Underwater Explorer dan Sensor Ping Sonar Echosounder

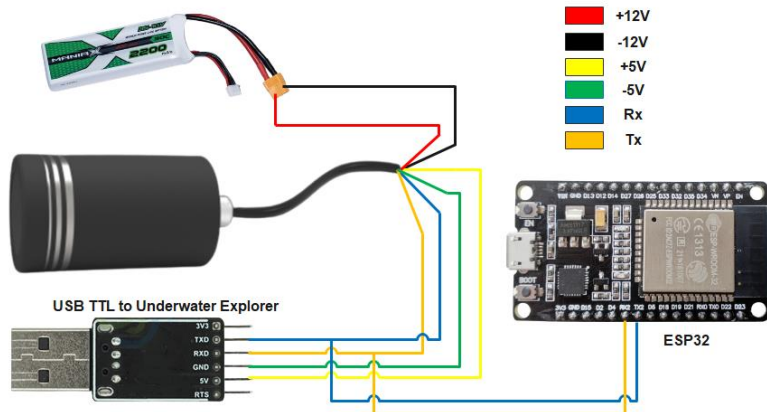
Pengumpulan data heksadesimal *output* dari *software* Underwater Explorer dilakukan dalam konfigurasi "Connect Device", "Read Mode Non-Wave", "Run", dan "Stop". Data heksadesimal dapat digunakan untuk mengakses ping sonar echosounder menggunakan mikrokontroler. Sedangkan data heksadesimal *output* dari hasil pembacaan sensor akan dikonversi ke dalam representasi data desimal yang dapat terbaca oleh pengguna.





Gambar 4: *Wiring Collect Data Heksadesimal dari Sensor* (a) *Wiring Data Collect dari Software* (b)

2.4 Accessing Sensor Ping Sonar Echosounder pada Mikrokontroler



Gambar 5: *Wiring Sensor Ping Sonar Echosounder dengan Mikrokontroler ESP32*

Pengaksesan sensor ping sonar echosounder menggunakan mikrokontroler dimulai dengan mengirim data heksadesimal konfigurasi "Connect Device" dan "Read Mode Non-Wave". Kemudian pembacaan sensor dapat dijalankan dan dihentikan dengan mengirim data heksadesimal "Run" dan "Stop".

```
void EchoConfig() {
    byte dataConnect[] = {0x12, 0x00, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE6};
    Serial1.write(dataConnect, sizeof(dataConnect));
    delay(1000);
    byte dataNonWave[] = {0x12, 0x00, 0x04, 0x04, 0x07, 0x01, 0xF0, 0x49, 0x02, 0x00, 0x32, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x6E};
    Serial1.write(dataNonWave, sizeof(dataNonWave));
}

void RunEcho() {
    byte dataRun[] = {0x12, 0x00, 0x06, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xC8};
    Serial1.write(dataRun, sizeof(dataRun));
}

void StopEcho() {
    byte dataStop[] = {0x12, 0x00, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xCC};
    Serial1.write(dataStop, sizeof(dataStop));
}
```

Gambar 6: Listing Program Konfigurasi Sensor menggunakan Mikrokontroler ESP32



2.5 Konversi Data Pembacaan Sensor Ping Sonar Echosounder menjadi Data Desimal

```

void readEcho() {
    curMillis4 = millis();
    if (curMillis4 - prevMillis4 >= 1000) {
        while (Serial1.available() > 0) {
            char inChar = Serial1.read();
            char hexChar[3];
            sprintf(hexChar, "%02X", inChar);
            String hexString = String(hexChar);
            if (hexString == "FC" && !started && !ended) {
                dataIndex = 0;
                started = true;
            } else if (hexString == "FC" && started) {
                ended = true;
                break;
            } else {
                if (dataIndex < 40) {
                    inData[dataIndex] = inChar;
                    dataIndex++;
                    inData[dataIndex] = '\0';
                }
            }
        }
        if (started && ended) {
            depth = (inData[5] << 8) | inData[4];
            Serial.println(depth);
            started = false;
            ended = false;
            dataIndex = 0;
            inData[dataIndex] = '\0';
        } prevMillis4 = curMillis4;
    }
}
    
```

Gambar 7: Listing Program Konfigurasi Sensor menggunakan Mikrokontroler ESP32

Pada Gambar 7 bagian program pada kotak biru merupakan program parsing data heksadesimal untuk memisahkan satu paket data dengan paket data lain. Setiap paket data terdiri dari "Start Code (FC)", "Main Data", dan "End Code (FC)". Ketika menemui Start Code program akan mulai mengoleksi "Main Data" dan ketika menemui "End Code" program akan berhenti mengoleksi data. "Main Data" yang telah disimpan pada variabel inData indeks ke 5 dan 6 akan dikonversi menggunakan program pada kotak putih dengan perhitungan matematis seperti dalam (1).

$$\text{Decimal Data} = (\text{High Byte} \times 256) + \text{Low Byte} \tag{1}$$

2.5 Noise Reduction Data Sensor Ping Sonar Echosounder menggunakan Metode Kalman Filter

Dalam mereduksi *noise* data sensor menggunakan metode kalman filter melalui 2 tahap yaitu tahap *prediction* dan tahap *update*. Pada tahap *prediction* terdapat perhitungan untuk memprediksi *state* (keadaan) seperti dalam (2) dan memprediksi *error* seperti dalam (3).

$$X_{t|t-1} = X_{t-1|t-1} \tag{2}$$

$$P_{t|t-1} = P_{t-1|t-1} + Q_t \tag{3}$$

X adalah *estimated state* (kondisi yang diestimasi), P adalah kovarin *state*, Q adalah kovarian *noise* proses, R adalah kovarian *noise* pengukuran, Y adalah data sensor waktu sekarang, R adalah kovarian *noise* pengukuran, $t|t-1$ adalah *intermediate steps*, $t-1|t-1$ adalah periode waktu sebelumnya, $t|t$ adalah periode waktu sekarang. Kemudian pada tahap *update* terdapat perhitungan untuk *update* nilai *state* seperti dalam (4), menghitung kalman *gain* seperti dalam (5) dan *update* nilai *error* seperti dalam (6).

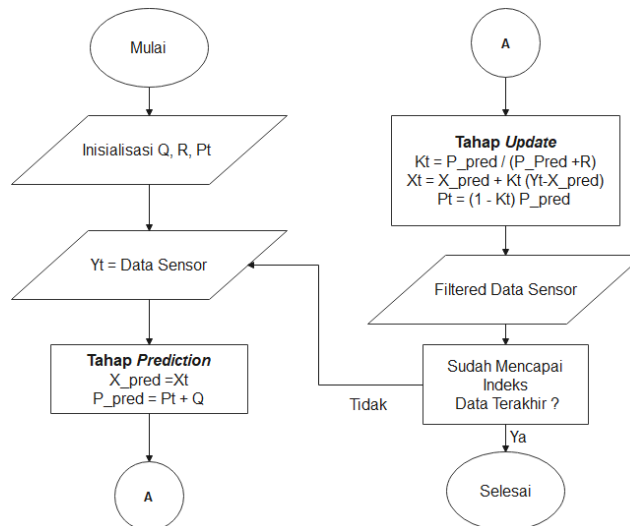
$$X_{t|t} = X_{t|t-1} + K_t(y_t - X_{t|t-1}) \tag{4}$$

$$K_t = P_{t|t-1}(P_{t|t-1} + R)^{-1} \tag{5}$$



$$P_{t|t} = (1 - K_t)P_{t|t-1} \tag{6}$$

Perhitungan seperti dalam (2), (3), (4), (5), dan (6) akan diadaptasi kedalam program mikrokontroler ESP32 untuk mereduksi *noise* data sensor ping sonar echosounder. Terdapat 2 varian kedalaman perairan di kedalaman 1.4 meter dan 2.2 meter yang masing-masing memiliki 20 sampel data sensor yang akan melalui proses *filtering* menggunakan metode kalman filter. Untuk mendapatkan hasil *filter* yang terbaik maka dilakukan 4 kali eksperimen dalam menentukan rasio antara nilai Q (Kovarian *noise* proses) dan R (Kovarian *noise* pengukuran).



Gambar 6: Flowchart Algoritma Pemrograman Kalman Filter pada Mikrokontroler

Langkah awal pada pemrograman mikrokontroler yaitu inisialisasi nilai Q,R, dan Pt kemudian memberi *input* data sensor disimpan ke variabel Yt. Setelah itu masuk ke tahap *prediction* untuk menghitung *prediction state* (X_pred) dan prediksi kovarian *noise* proses (P_pred). Kemudian masuk ke tahap *update* untuk menghitung kalman *gain* (Kt), memperbarui nilai *state* (Xt) dan memperbarui nilai *error* (Pt). Program *noise reduction* data sensor ping sonar echosounder seperti pada Gambar 7.



```

const int TotalSample = 20;
double sensorData[Samples] = {
    1440, 1521, 1372, 1404, 1429, 1442, 1375, 1405, 1357, 1452,
    1438, 1407, 1399, 1466, 1454, 1504, 1480, 1467, 1363, 1412
};

// Inisialisasi variabel-variabel untuk Kalman filter
double Q = 1; // Kovarian noise proses
double R = 1; // Kovarian noise pengukuran
double Xt = 1400; // Estimated State
double Pt = 1; // Kovarian state
double Kt; // Kalman gain
double X_pred, P_pred, Yt;

void setup() {
    Serial.begin(115200);
    Serial.println("Data Sensor, Filtered Data Sensor");
    for (int i = 0; i < numSamples; i++) {
        Yt = sensorData[i];

        // Tahap Prediction
        X_pred = Xt; // Prediction state
        P_pred = Pt + Q; // Prediksi kovarian noise proses

        // Tahap Update
        Kt = P_pred / (P_pred + R); // Calculate Kalman gain
        Xt = X_pred + Kt * (Yt - X_pred); // Update nilai state
        Pt = (1 - Kt) * P_pred; // Update nilai error
        Serial.print(Yt);
        Serial.print(",");
        Serial.println(Xt);
        delay(100);
    }
}

void loop() {
}
    
```

Gambar 7: Listing Program *Noise Reduction* menggunakan Metode Kalman Filter

3. HASIL DAN PEMBAHASAN

Tahapan pertama yaitu mengakses sensor echosounder menggunakan *software* Underwater Explorer untuk dapat menampilkan pembacaan data kedalaman perairan. Kemudian dari hasil pembacaan sensor echosounder perlu dibandingkan dengan kondisi aktual dari kedalaman suatu perairan menggunakan meteran sebagai validasi rata-rata *%error*. Hasil validasi seperti pada Tabel 1.

TABEL I: VALIDASI PEMBACAAN SENSOR ECHOSOUNDER TERHADAP KONDISI AKTUAL 1400 MM DAN 2200 MM

Pengujian Ke-	<i>Real Depth</i> (mm)	Echosounder Depth (mm)	<i>%Error</i>	<i>Real Depth</i> (mm)	Echosounder Depth (mm)	<i>%Error</i>
1	1400	1440	2,78	2200	2293	4,06
2	1400	1521	7,96	2200	2144	2,61
3	1400	1372	2,04	2200	2159	1,90
4	1400	1404	0,28	2200	2194	0,27
5	1400	1429	2,03	2200	2200	0,00
6	1400	1442	2,91	2200	2266	2,91
7	1400	1375	1,82	2200	2106	4,46
8	1400	1405	0,36	2200	2115	4,02
9	1400	1357	3,17	2200	2135	3,04
10	1400	1452	3,58	2200	2124	3,58
11	1400	1438	2,64	2200	2166	1,57



12	1400	1407	0,50	2200	2188	0,55
13	1400	1399	0,07	2200	2193	0,32
14	1400	1466	4,50	2200	2167	1,52
15	1400	1454	3,71	2200	2174	1,20
16	1400	1504	6,91	2200	2141	2,76
17	1400	1480	5,41	2200	2273	3,21
18	1400	1467	4,57	2200	2210	0,45
19	1400	1363	2,71	2200	2274	3,25
20	1400	1412	0,85	2200	2250	2,22
Rata-rata %Error			2,94	Rata-rata %Error		2,195

Berdasarkan data yang tercantum pada Tabel I terdapat selisih antara nilai kedalaman aktual dengan hasil pembacaan sensor echosounder. Terdapat *error* sebesar 2.94% pada pembacaan sensor di kedalaman 1400mm dan *error* sebesar 2.195% di kedalaman 2200mm. Hal ini dapat disimpulkan bahwa performa sensor echosounder dinyatakan layak karena memiliki rata-rata %*error* dibawah 5%.

TABEL II: DATA OUTPUT SOFTWARE UNDERWATER EXPLORER

No	Mode Konfigurasi	Data Hexadecimal
1	Connect to Device	12 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 E6
2	Read Non-Wave Mode	12 00 04 04 07 01 F0 49 02 00 32 00 00 00 00 00 00 00 6E
3	Run	12 00 06 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 C8
4	Stop	12 00 06 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 CC

Pada Tabel II merupakan hasil pengkoleksian data *output software* Underwater Explorer dalam berbagai mode konfigurasi. Data-data *hexadecimal* tersebut yang akan digunakan untuk mengakses sensor echosounder menggunakan ESP32.

TABEL III: DATA OUTPUT PEMBACAAN SENSOR ECHOSOUNDER

No	Depth (mm)	Hexadecimal Data
1	1450	FC 90 D0 03 00 AA 05 00 00 FC
2	1500	FC 90 D0 03 00 DC 05 00 00 FC
3	1800	FC 90 D0 03 00 08 07 00 00 FC
4	1900	FC 90 D0 03 00 6C 07 00 00 FC
5	2150	FC 90 D0 03 00 66 08 00 00 FC

Berdasarkan Tabel III data *output* pembacaan sensor echosounder memiliki struktur yang terdiri atas "Start Code (FC)", "Main Data" yang terletak diantara *Start* dan *End Code*, dan "End Code (FC)" seperti yang telah dijelaskan pada bab 2. Perhitungan konversi data heksadesimal ke data decimal dengan kedalaman 1900mm seperti pada (7).

Diketahui:

High Byte : 07 (*Hexadecimal*) ≈ 7 (*Decimal*)

Low Byte : 6C (*Hexadecimal*) ≈ 108 (*Decimal*)

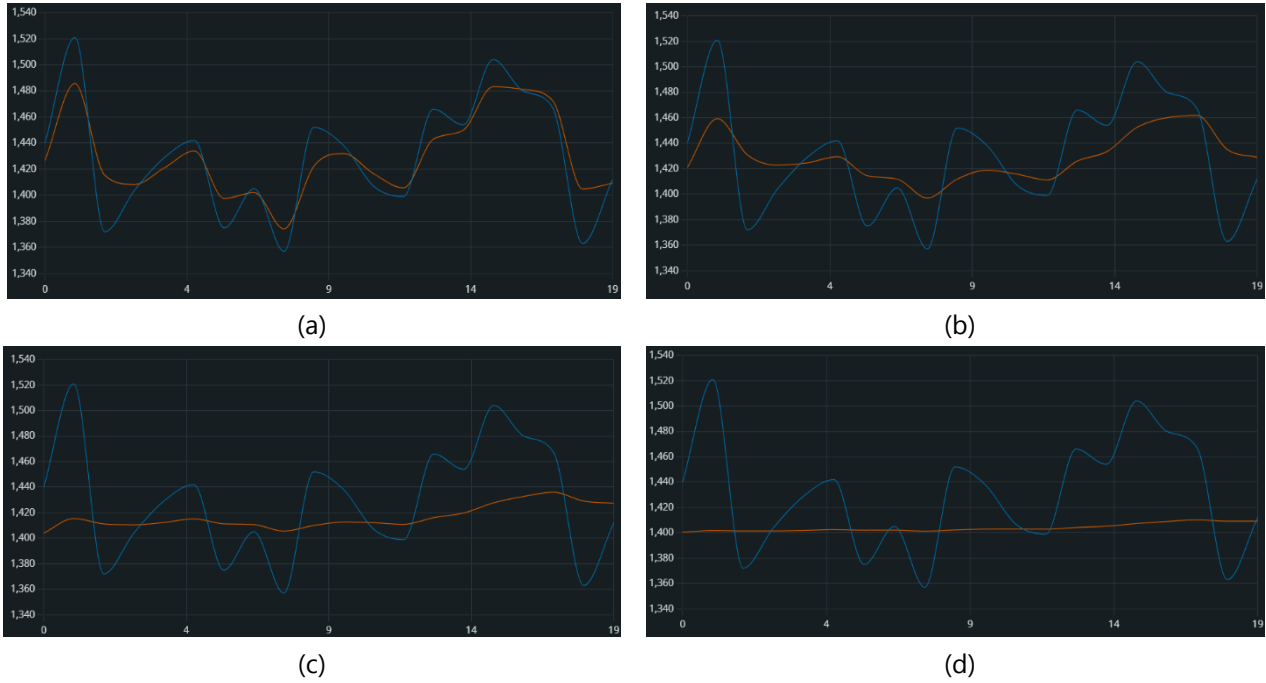
Jadi:

Decimal Data = (High Byte x 256) + Low Byte

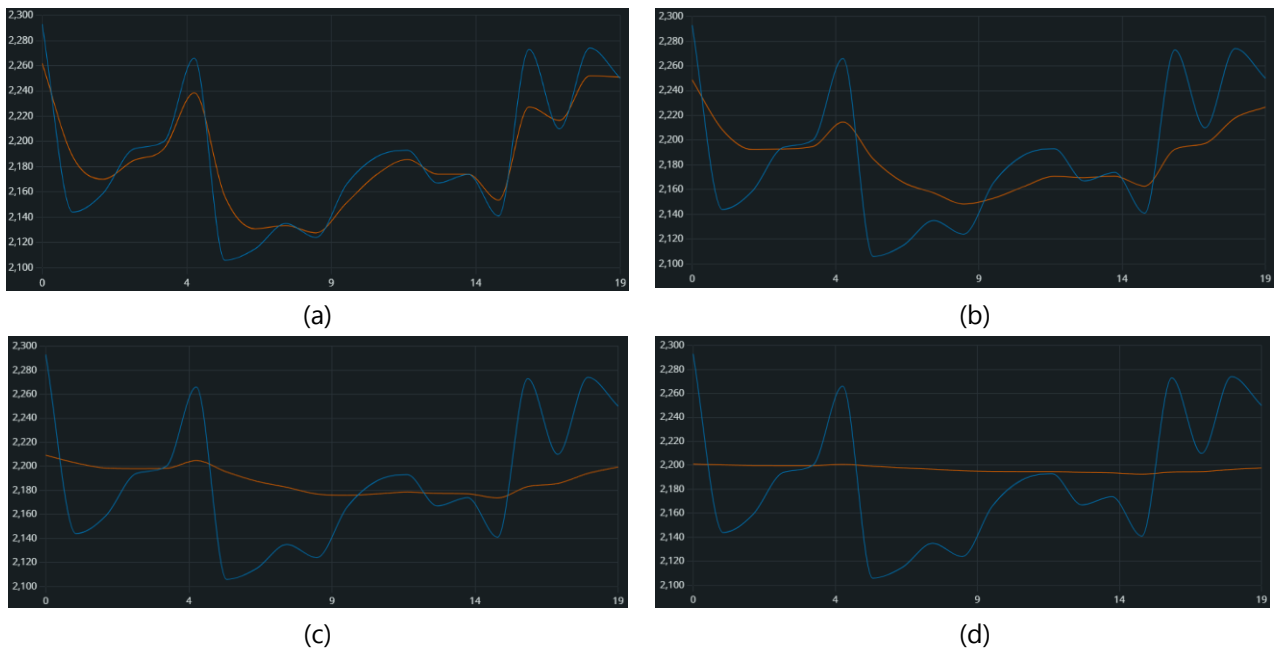
Decimal Data = (7 x 256) + 108 = 1900 mm ≈ 1.90m (7)



Noise reduction data sensor menggunakan metode Kalman filter dilakukan eksperimen terhadap rasio antara nilai Q dan R untuk menghasilkan respon *filtering* yang terbaik. Hasil penerapan metode kalman filter ditampilkan pada serial plotter Arduino IDE. Grafik perbandingan antara data sensor sebelum di *filter* (Biru) dan sesudah di *filter* (merah) seperti pada Gambar 8, Gambar 9 dan keseluruhan hasil dirangkum pada Tabel IV.



Gambar 8: R = 1 dan Q = 1 (a) R = 1 dan Q = 0,1 (b) R = 10 dan Q = 0,1 (c) R = 100 dan Q = 0,1 (d) pada kedalaman 1400mm



Gambar 9: R = 1 dan Q = 1 (a) R = 1 dan Q = 0,1 (b) R = 10 dan Q = 0,1 (c) R = 100 dan Q = 0,1 (d) pada kedalaman 2200mm



TABEL IV: PERHITUNGAN RMSE (ROOT MEAN SQUARE ERROR) HASIL *NOISE REDUCTION* KEDALAMAN 1400MM

Experiment	Kalman Filter Parameter Value		Rasio (R dan Q)	RMSE (Root Mean Square Error)
	Ke-	R		
1		1	1	42,23
2		1	10	33,33
3		10	100	18,79
4		100	1000	5,05

TABEL V: PERHITUNGAN RMSE (ROOT MEAN SQUARE ERROR) HASIL *NOISE REDUCTION* KEDALAMAN 2200MM

Experiment	Kalman Filter Parameter Value		Rasio (R dan Q)	RMSE (Root Mean Square Error)
	Ke-	R		
1		1	1	42,43
2		1	10	29,45
3		10	100	15,74
4		100	1000	3,99

4. KESIMPULAN

Penelitian ini berhasil mengembangkan sistem survei batimetri otomatis menggunakan sensor ping sonar echosounder ROVMAKER yang dipasang pada kendaraan *unmanned surface vehicle* (USV) dan diakses menggunakan mikrokontroler ESP32. Sensor ping sonar echosounder yang digunakan memiliki rata-rata kesalahan pembacaan kedalaman sebesar 2,94% di kedalaman 1400mm dan 2,195% di kedalaman 2200mm. Hal ini menunjukkan bahwa sensor ini cukup akurat dan layak digunakan untuk survei batimetri. Dengan menggunakan mikrokontroler ESP32, sensor ping sonar echosounder dapat diakses dan dikonfigurasi tanpa perlu menggunakan *software* bawaan dari ROVMAKER. Data heksadesimal dari hasil pembacaan sensor ping sonar echosounder berhasil dikonversi menjadi data desimal yang memudahkan pengguna untuk membaca data. Untuk mereduksi *noise* data sensor dapat diatasi dengan baik menggunakan metode kalman filter dengan bereksperimen terhadap rasio (R/Q) dari rasio 1 hingga 1000. *Noise* dapat tereduksi dengan baik pada rasio (R/Q) 1000 yang memiliki RMSE (*Root Mean Square Error*) diangka 5,05 pada kedalaman 1400mm dan 3,99 pada kedalaman 2200mm. Pengembangan sistem ini diharapkan dapat membantu dalam mitigasi bencana banjir di Indonesia dengan menyediakan data batimetri yang akurat dan dapat diakses dengan mudah, sehingga memudahkan dalam pengelolaan dan pemeliharaan tempat penyimpanan air seperti sungai dan waduk.

DAFTAR PUSTAKA

- [1] G. Nikawanti, "Ecoliteracy: Membangun Ketahanan Pangan Dari Kekayaan Maritim Indonesia," *Jurnal Kemaritiman: Indonesian Journal of Maritime*, vol. 2, no. 2, pp. 149–166, 2021.
- [2] D. SAFITRI, "KARAKTERISTIK ALIRAN DAN DEBIT BANJIR PADA BEBERAPA SUNGAI DI INDONESIA: KAJIAN LITERATUR," *JICE (Journal of Infrastructural in Civil Engineering)*, vol. 2, no. 02, p. 1, Aug. 2021, doi: 10.33365/jice.v2i02.1322.
- [3] D. Setiawan, "Analisis curah hujan di Indonesia untuk memetakan daerah potensi banjir dan tanah longsor dengan Metode Cluster Fuzzy C-Means dan Singular Value Decomposition (SVD)," *Engineering, Mathematics and Computer Science Journal (EMACS)*, vol. 3, no. 3, pp. 115–120, 2021.
- [4] T. P. E. Pratama, W. P. Prihadita, V. P. Yuliatama, S. P. Ramadhani, W. Safitri, and H. N. Syifa, "Analisis Index Overlay Untuk Pemetaan Kawasan Berpotensi Banjir di Gowa, Provinsi Sulawesi Selatan," *Jurnal Geosains dan Remote Sensing*, vol. 1, no. 1, pp. 52–63, 2020.
- [5] M. B. Al Amin, *Pemodelan sistem drainase perkotaan menggunakan SWMM*. Deepublish, 2020.
- [6] R. D. V. Rut, "Rancang Bangun Peringatan Dini Banjir Berbasis Arduino Uno," *Jurnal Portal Data*, vol. 1, no. 2, 2021.
- [7] R. NUGRAHA, "SURVEI BATIMETRI DI AREA VOID PIT DENGAN MENGGUNAKAN WAHANA TELEDYNE ODOM SINGLEBEAM ECHOSOUNDER SINGLE FREQUENCY," *Prosiding Temu Profesi Tahunan PERHAPI*, pp. 191–200, 2020.



- [8] Y. Yuwono, D. G. Pratomo, and M. I. F. Al-Azhar, "Analisis posisi kerangka kapal terhadap keselamatan alur pelayaran menggunakan data multibeam echosounder (Studi kasus: Alur pelayaran barat Surabaya)," *Geoid*, vol. 14, no. 1, pp. 13–21, 2018.
- [9] Oktafari and Daffa Naufal, "Rancang Bangun Konfigurasi Sistem Pengambilan Data Batimetri Kontur Dasar Laut dengan Unmanned Surface Vehicle (USV)," Institut Teknologi Sepuluh Noverber, 2023.
- [10] A. H. DWIANTO and D. T. T. LAUT, "ANALISIS POLA OPERASI DAN PERANCANGAN KAPAL PINISI UNTUK WISATA BAHARI: STUDI KASUS KEPULAUAN SELAYAR".
- [11] A. Ma'arif, I. Iswanto, A. A. Nuryono, and R. I. Alfian, "Kalman filter for noise reducer on sensor readings," *Signal and Image Processing Letters*, vol. 1, no. 2, pp. 50–61, 2019.
- [12] M. Safitri, A. Cahyadi, E. Firmansyah, and J. Grafika, "Estimasi Posisi UAV dengan Kalman Filter," *no. September*, pp. 240–245, 2015.
- [13] I. S. H. Sujoto and H. Sutiksno, "Studi Analisa Kalman Filter Sebagai State Estimator Untuk Meningkatkan Akurasi Pengukuran Kecepatan Motor DC," *Journal of Information System, Graphics, Hospitality and Technology*, vol. 3, no. 01, pp. 9–18, Apr. 2021, doi: 10.37823/insight.v3i01.96.
- [14] W. S. Pambudi and I. Suhendra, "Perbaikan respon output menggunakan implementasi kalman filter pada simulasi pembacaan sensor beban load cell," in *Seminar Nasional Sains Dan Teknologi Terapan*, 2015, pp. 141–150.
- [15] M. A. Bahris, F. Nasrudien, B. Ferdiansyah, D. E. Yulian, M. Thaifur, and W. S. Pambudi, "Implementasi Kalman Filter Pada Pengukuran Ketinggian Air Menggunakan Sensor Ultrasonik," in *Prosiding Seminar Nasional Teknik Elektro, Sistem Informasi, dan Teknik Informatika (SNESTIK)*, 2022, pp. 109–115.
- [16] Y. Pei, S. Biswas, D. S. Fussell, and K. Pingali, "An elementary introduction to Kalman filtering," *Commun ACM*, vol. 62, no. 11, pp. 122–133, 2019.

