

DAFTAR PUSTAKA

- [1] Ikhwan, F. 2019. Mobile Robot Pengikut Bola Menggunakan CMUCAM 3. Undergraduate thesis, Universitas Muhammadiyah Jember.
- [2] Haq, Rijalul., Rahmawati, E. 2017. Kendali Posisi Mobile Robot Menggunakan Sistem *Proportional Integral Derivative* (PID) Dengan Metode *Odometry*. *Jurnal Inovasi Fisika Indonesia (IFI)*. Vol 6, No 3.
- [3] Rochmanto, A.R., Muslim, A., & Rif'an, M. 2020. Rancang Bangun Robot *Three wheeled Omnidirectional* Dengan Konfigurasi Mikrokontroler *Master-Slave*. *Jurnal Fokus Elektroda Jurnal Fokus Elektroda*. Vol 5, No 4, Hal 7-10.
- [4] Rachman, A.A. 2020. Sistem Perencanaan Rute Gerak Pada Robot Sepakbola Beroda. *Tugas Akhir - TE 141599 Institut Teknologi Sepuluh November*.
- [5] Lutfika, A.I. 2020. Kendali Prediksi Berbasis Model untuk Robot Beroda Tipe *Holonomic* Tiga Roda. *Skripsi Jurusan Teknik Elektro Fakultas Teknologi Industri Universitas Islam Indonesia Yogyakarta*.
- [6] Abseno, A.P. 2019. Penerapan Kinematika untuk Lokalisasi pada Robot Sepak Bola Beroda. *Tugas Akhir Fakultas Teknologi dan Informatika Institut Bisnis dan Informatika STIKOM Surabaya*.
- [7] Wahab, F. 2022. Metode *Potential Field* sebagai Kendali Robot Roda Omni untuk Menuju Target dan Menghindari Rintangannya. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*. Vol 10, No 1, Hal 177-188.
- [8] Pambudi, A.D. 2015. Sistem Navigasi Dengan Kontrol PID Pada *Three Wheel Omni Directional Mobile Robot* Menggunakan Metode *Odometry*. *Tugas Akhir Universitas Dian Nusawantoro Semarang*.
- [9] Sihono., dkk. 2020. Robot Navigasi *Odometry* Dengan Antarmuka MYRIO. *Seminar Nasional Hasil Penelitian dan Pengabdian 2020*. Vol 8.
- [10] Kariyanto, J.D., Alasiry, A.H., Ardila, F., Hanafi, N. 2017. Navigasi Mobile Robot Berbasis Trajektori dan *Odometry* dengan Pemulihan Jalur Secara Otomatis. *Jurnal Jurusan Teknik Elektronika, Politeknik Elektronika Negeri Surabaya*.

- [11] Wibowo, I.K., dkk. 2018. Penentuan Posisi Robot Menggunakan *Odometry* Omniwheel. *The Indonesian Symposium on Robot Soccer Competition 2018*.
- [12] Fahmizal., Rijalussalam, D.U., Budiyanto, M., Mayub, A. 2019. *Trajectory Tracking* pada Robot Omni dengan Metode *Odometry*. *JNTETI*. Vol 8, No 1.
- [13] Pratama, E.O., Leamongga, C. 2019. Sistem Navigasi Pada Robot Sepak Bola Beroda Krsbi Menggunakan Metode *Odometry*. *Proyek Akhir Politeknik Manufaktur Negeri Bangka Belitung*.
- [14] Wardana, R.D.A., Koko, J., Wibisono, K.A., Ulum, M. 2021. Implementasi Sistem Kontrol pada Robot Penjaga Gawang Berbasis *Odometry* dan PID. *Journal Of ALINIER Artificial Intelligence & Applications*. Vol 2, No 1.
- [15] Prayoga, A.A. 2021 Penerapan *Odometri* Pada Robot Kiper KRSBI Beroda. *Tugas Akhir Politeknik Negeri Cilacap*.
- [16] Indrawan, R.W., Sulistijono I.A., Basuki, A. 2019. Pemetaan 3 Dimensi Untuk Menentukan Jalur Evakuasi Alternatif Pada Smart Robot Rescue. *Jurnal Inovtek Polbeng*, Vol 9, No 2.
- [17] Syahputra, M.F., Hardywantara, F., Andayani, U. 2020. *Augmented Reality Virtual House Model Using ARCore Technology Based on Android*. *Journal of Physics: Conference Series*.
- [18] Taufik, A.S. 2013. Sistem Navigasi *Waypoint* pada *Autonomous Mobile Robot*. *Jurnal Mahasiswa Teknik Elektro Universitas Brawijaya*.
- [19] Irfan, M., & Haryanto, D. 2020. Model Sistem Navigasi Inersial: Sebuah Tinjauan *Model Of Inertial Navigation System: A Review*. *Oceanika: Jurnal Riset dan Rekayasa Kelautan*. Vol 1, No 1.
- [20] Kadir, Abdul. (2017). *Pemograman Arduino & Android Menggunakan App Invector*. Jakarta: PT Elex Media Komputindo.
- [21] Simanjuntak, B.J.S., Susanto, E., Estanto. 2019. Perancangan Antarmuka Pendulum Terbalik Menggunakan Visual Studio. *Jurnal e-Proceeding of Engineering*. Vol 6, No 2.
- [22] Desnanjaya, I.G.M., Iswara I.B.A.I. 2018. *Trainer Atmega32* Sebagai Media Pelatihan Mikrokontroler dan Arduino. *Jurnal RESISTOR*. Vol 1, No 1.
- [23] Dr. Junaidi, S.Si., M.Sc, Yuliyani Dwi Prabowo. 2018. *Projek Sistem Kendali Elektronik Berbasis Arduino*. Bandar Lampung : AURA, CV Anugrah Utama Raharja.

- [24] Sitompul, F.O., Baqaruzi, S., Muhtar, A. 2021. Rancang Bangun Sistem Gerak Mekanik Menggunakan *MotorBrushless Direct Current* Dengan *Driver BTS7960* Pada *Renewable Energy Smart Trolley* (Resol). *Jurnal E- JOINT (Electronica and Electrical Journal of Innovation Technology)*. Vol 2, No 2.
- [25] Kholiqin, S.K., Siradjuddin, I, Sungkono. 2021. Sistem Kendali Kecepatan Putaran Motor Robot Omnidirectional dengan 4 Penggerak Mekanum Menggunakan PID Berbasis myRIO. *Jurnal Elko*. Vol 8, No 3.
- [26] Harahap, C.A., Manik, M.I. 2020. Rancang Bangun Robot Pemantau Ruangan Menggunakan Jaringan Nirkabel. *Jurusan Teknik Elektro, Politeknik Negeri Medan, Medan, Indonesia*. Vol 1, No 1. Hal 36-4.

DAFTAR LAMPIRAN

LAMPIRAN A

Listing Program

```
//program kinematika

double deg2rad(double d)
{
    return d * PI / 180.0 ;
}

void kinematik(float x_speed , float y_speed, float w_speed) {
    x_speed = x_speed * -1;
    y_speed = y_speed * -1;
    double a_inverse, b_inverse, c_inverse,
           d_inverse, e_inverse, f_inverse,
           g_inverse, h_inverse, i_inverse;
    double a, b, c, d, e, f, g, h, i;
    double det;

    // a = -0.5;    b = -0.5;    c = 1;
    // d = 0.87;    e = -0.87;    f = 0;
    // g = 0.054;   h = 0.054;    i = 0.054; // 1/L = 1/18,5 cm

    //----- inverse kinematik -----//
    // roda kanan  = 30 + 90 = 120 derajat
    // roda kiri   = 150 + 90 = 240 derajat
    // roda belakang = 270 + 90 = 360 derajat
    //-----//

    a = cos(deg2rad(120)); b = cos(deg2rad(240)); c = cos(deg2rad(360));
    d = sin(deg2rad(120)); e = sin(deg2rad(240)); f = sin(deg2rad(360));
    g = 1;    h = 1;    i = 1;

    det = a * e * i + b * f * g + c * d * h - c * e * g - a * f * h - b * d * i;
```



```

a_inverse = (e * i - f * h) / det; b_inverse = (h * c - i * b) / det;
c_inverse = (b * f - c * e) / det;
d_inverse = (g * f - d * i) / det; e_inverse = (a * i - g * c) / det;
f_inverse = (d * c - a * f) / det;
g_inverse = (d * h - g * e) / det; h_inverse = (g * b - a * h) / det;
i_inverse = (a * e - d * b) / det;

```

```

Vy = a_inverse * x_speed + b_inverse * y_speed + c_inverse *
w_speed;
Vx = d_inverse * x_speed + e_inverse * y_speed + f_inverse *
w_speed;
V0 = g_inverse * x_speed + h_inverse * y_speed + i_inverse *
w_speed;

```

```

// jika error Vx tukar dnegan Vy
// fungsinya untuk arah gerak (+) atau (-) kak

```

```

// Serial.print ("kecepatan roda kiri : ");
// Serial.println(Vy);
// Serial.print ("kecepatan roda kanan : ");
// Serial.println(Vx);
// Serial.print ("kecepatan roda tengah : ");
// Serial.println(V0);

```

```

d1 = 0;
d2 = 0;
d3 = 0;
d1 = Vy < 0 ? -1 : 1;
d2 = Vx < 0 ? -1 : 1;
d3 = V0 < 0 ? -1 : 1;

```

```

M1(Vy, d1);
M2(Vx, d2);
M3(V0, d3);
}

```

```

//Program PID
int target_x(int target, int odometry_x) {

int speed_x = max_speed_x;
  setPoin_x = target;
  float Kp_X = 22;
  float Ki_X = 0;
  float Kd_X = 0;
  int P_X = 0;
  int I_X = 0;
  int D_X = 0;
  float PID_X = 0;

  timePrev_x = time_x;
  time_x = millis();
  elapsedTime_x = (time - timePrev_x) / 500;
  uint16_t position = odometry_x ;
  float error = setPoin_x - position ; //setPoint-dataReal
  Serial.print("\n\n");
  Serial.print(setPoin_x);
  Serial.print(" ");
  Serial.print(odometry_x);
  Serial.print("\n\n");
  error_t_x = error;
  P_X = Kp_X * error;
  if (-1 < error < 1) {
    I_X = I_X + (Ki_X * error);
  }
  D_X = Kd_X * ((error - previous_error_x) / elapsedTime_x);
  PID_X = P_X + I_X + D_X;
  if (PID_X < - speed_x) {
    PID_X = - speed_x;
  }
  else if (PID_X > speed_x) {
    PID_X = speed_x;
  }
  previous_error_x = error_x;
  return PID_X;
}

```



```

int target_y(int target, int odometry_y) {
    int speed_y = max_speed_y;
    setPoin_y = target;
    float Kp_Y = 22;
    float Ki_Y = 0;
    float Kd_Y = 0;
    int P_Y = 0;
    int I_Y = 0;
    int D_Y = 0;
    float PID_Y = 0;

    timePrev_y = time_y;
    time_y = millis();
    elapsedTime_y = (time - timePrev_y) / 500;
    int position = setPoin_y;
    // Serial.print("\n\n");
    // Serial.print(setPoin_y);
    // Serial.print(" ");
    // Serial.print(odometry_y);
    // Serial.print("\n\n");
    float error = position - odometry_y; //setPoint-dataReal
    error_t_y = error;
    P_Y = Kp_Y * error;
    if (-1 < error < 1) {
        I_Y = I_Y + (Ki_Y * error);
    }
    D_Y = Kd_Y * ((error - previous_error_y) / elapsedTime_y);
    PID_Y = P_Y + I_Y + D_Y;
    if (PID_Y < -speed_y) {
        PID_Y = -speed_y;
    }
    else if (PID_Y > speed_y) {
        PID_Y = speed_y;
    }
    previous_error_y = error_y * -1; // global
    return PID_Y;
}

```



```

void headingArah() {
    Kp_H = 8; //12.5
    float Ki_H = 0;
    float Kd_H = 0.8;
    int P_heading = 0;
    int I_heading = 0;
    int D_heading = 0;
    timePrev = time;
    time = millis();
    elapsedTime = (time - timePrev) / 500;
    uint16_t position = yaw;
    error = setPoin_kompas - position; //setPoint-dataReal
    P_heading = Kp_H * error;
    if (-1 < error < 1) {
        I_heading = I_heading + (Ki_H * error);
    }
    D_heading = Kd_H * ((error - previous_error) / elapsedTime);
    PID_heading = P_heading + I_heading + D_heading;
    if (PID_heading < -255) {
        PID_heading = -255;
    }
    else if (PID_heading > 255) {
        PID_heading = 255;
    }
    previous_error = error;
}

//Program Odometri
void odometry()
{
    data_x = enc_belakang - (0.5 * enc_kanan) - (0.5 * enc_kiri);
    data_y = (0.87 * enc_kanan) - (0.87 * enc_kiri);
    data_heading = -(enc_kiri + enc_kanan + enc_belakang) / 3;
    nilaiY = 0.6236 * data_y*-1;
    nilaiX = 0.6236 * data_x*-1;
}

```



```

//Program komunikasi Bluetooth
void serialBT_kirim() {

    int kirimX = nilaiX;
    int kirimY = nilaiY;
    int kirimH = head_monitor;
    // int kirimJ = nilaiJ;
    Serial2.print(kirimX);
    Serial2.print(",");
    Serial2.print(kirimY);
    Serial2.print(",");
    Serial2.print(kirimH);
    Serial2.print(",");
    Serial2.println(jarak);

}

void serial_terima() {
    if (Serial2.available() > 0) {
        dataMasuk = Serial2.readStringUntil('\n');
        dataMasuk.trim();
        Serial.print(dataMasuk);
        Serial.print(" ");

        if (dataMasuk.startsWith("{") {
            byte buka = dataMasuk.indexOf('{');
            byte koma1 = dataMasuk.indexOf(',');
            byte koma2 = dataMasuk.indexOf(',', koma1 + 1);
            byte tutup = dataMasuk.indexOf('}');

            firstVal = dataMasuk.substring(buka + 1, koma1);
            secondVal = dataMasuk.substring(koma1 + 1, koma2);
            thirdVal = dataMasuk.substring(koma2 + 1, tutup);

            x_in = firstVal.toInt();
            y_in = secondVal.toInt();
            heading_in = thirdVal.toInt();
        }
    }
}

```



```

if (dataMasuk.startsWith("[") {

    int buka = dataMasuk.indexOf('[');
    int koma1 = dataMasuk.indexOf(',');
    int rampung = dataMasuk.indexOf(']');

    String firstVal = dataMasuk.substring(buka + 1, koma1);
    String secondVal = dataMasuk.substring(koma1 + 1, rampung);

    xAxis = firstVal.toInt();
    yAxis = secondVal.toInt();

}
else if (dataMasuk.startsWith(":")) {

    int buka = dataMasuk.indexOf(':');
    int tutup = dataMasuk.indexOf(';');

    String firstVal = dataMasuk.substring(buka + 1, tutup);

    putar = firstVal;

}

}
}
}

```


BIODATA PENULIS



Nama : Bagus Aulia Rahman
Tempat/Tanggal Lahir : Cilacap, 7 Agustus 2001
Alamat : Perum Papan Indah Blok i4 no 15, Kec.
Tambun Selatan, Kab.Bekasi.
Telepon/Hp : 082224311572
Moto : "Belajar Banyak Hari Ini Untuk Hari Esok"

Riwayat Pendidikan :

- SD Negeri Mangun Jaya 05 Tahun 2007-2013
- SMPIT Budi Luhur Tahun 2013-2016
- SMK Negeri 1 Cikarang Barat Tahun 2016-2019
- Politeknik Negeri Cilacap Tahun 2019-2022