

RANCANG BANGUN CONTROLLER GAME FPS MENGUNAKAN SENJATA MAINAN YANG DIMODIFIKASI BERBASIS ATMEGA 32U4

DESIGN OF FPS GAME CONTROLLER USING MODIFIED TOYS WEAPONS BASED ON ATMEGA 32U4

Adityo Candra Wirawan¹, Denny Darlis², Syahban Rangkti³

^{1,2,3}Universitas Telkom, Bandung

tyocandra@student.telkomuniversity.ac.id¹, dennydarlis@tass.telkomuniversity.ac.id²,
syahkti@tass.telkomuniversity.ac.id³

Abstrak

Salah satu cabang atau genre *game* yang paling populer yaitu FPS atau First Person Shooter, pada cabang tersebut pemain akan melakukan permainan dalam tim yang biasanya dalam satu tim berisikan empat atau lima orang, dimana setiap tim akan melakukan misinya seperti menjinakan bom, membebaskan sandra, dll. Untuk memainkan *game* dengan genre tersebut, biasanya pemain menggunakan beberapa perangkat seperti *mouse* dan *keyboard*. Akan tetapi, pemain tidak akan merasakan menjadi karakter yang ada dalam *game* yang dia mainkan. Pada proyek tingkat ini akan dibuat sebuah perangkat senjata yang dirancang menggunakan mikrokontroler berbasis ATmega32U4 yaitu Arduino *pro micro* dan menggunakan sensor gyro dan beberapa push *button*. ATmega32U4 dipilih karena mikrokontroler tersebut telah terintegrasi komunikasi USB sehingga dapat digunakan menjadi *mouse* dan *keyboard*. Sensor gyro digunakan sebagai pengganti *mouse* karena dapat mendeteksi kemiringan sudut sehingga dapat digunakan sebagai pengganti *mouse*. Selain itu push *button* digunakan sebagai pengganti *keyboard*. Alat ini dibuat dengan tujuan agar para pemain *game* aliran FPS ini dapat lebih merasakan feel saat memainkan *game* tersebut.

Kata kunci : *Game FPS, senjata mainan, ATmega32U4, Arduino pro micro*

Abstract

One of the most popular branches or game genres, namely FPS or First Person Shooter, in this branch players will play in teams which are usually in a team of four or five people, where each team will carry out missions such as defusing bombs, freeing hostages, etc. To play games with this genre, players usually use several devices such as a mouse and keyboard. However, the player will not feel like a character in the game he is playing. At this level project, a weapon device designed using an ATmega32U4-based microcontroller, the Arduino *pro micro*, will be built using a gyro sensor and several push buttons. ATmega32U4 was chosen because the microcontroller has integrated USB communication so that it can be used as a mouse and keyboard. The gyro sensor is used as a substitute for a mouse because it detects the tilt angle so it can be used as a replacement for a mouse. In addition, the push button is used as a replacement for the keyboard. This tool was made with the aim that the FPS flow game players can feel the feel more when playing the game.

Keyword : *FPS Game, toy gun, ATmega32U4, Arduino pro micro*

1. PENDAHULUAN

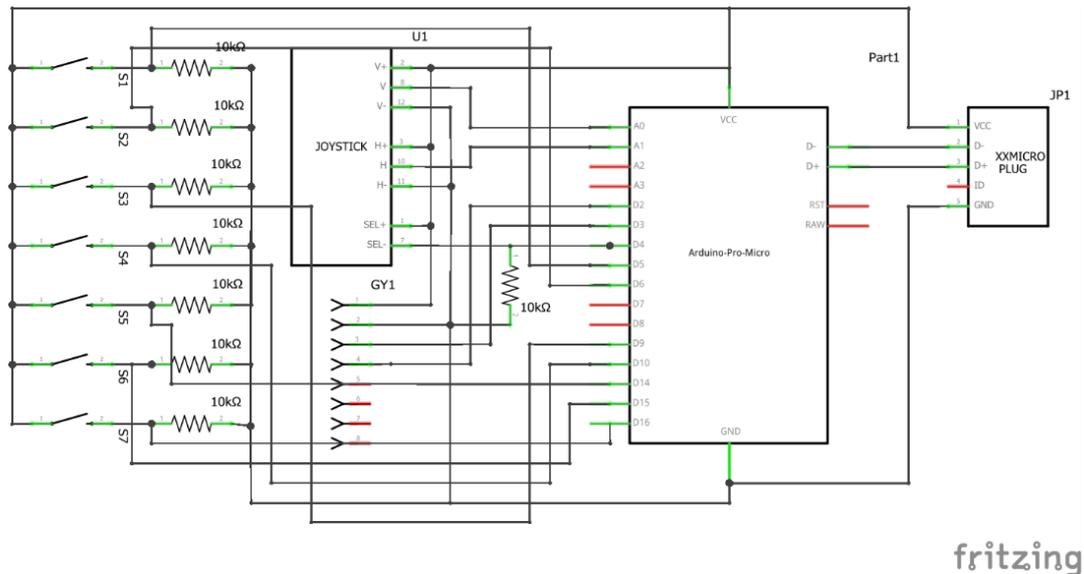
Dalam *game* berjenis FPS biasanya pemain akan menggunakan *mouse* dan *keyboard* untuk mengendalikan karakternya, tetapi selain menggunakan kedua alat tersebut para pemain juga dapat menggunakan sebuah kontroler yang bentuknya menyerupai senjata sehingga para pemain dapat merasakan pengalaman bermain yang lebih baik dan hanya menggunakan satu perangkat saja.

Karena para pemain *game* FPS menggunakan perangkat *mouse* dan *keyboard* untuk menggerakkan karakternya dimana akan membutuhkan banyak *port* USB dan kurangnya pengalaman bermain, maka dibuat sebuah kontroler

berbentuk senjata yang dapat berfungsi sebagai *mouse* dan juga *keyboard* dalam satu waktu, sehingga dapat menghemat *port* pada komputer dan pemain dapat merasakan pengalaman bermain yang lebih nyata daripada menggunakan *mouse* dan *keyboard*.

2. PERANCANGAN DAN SIMULASI

Pada proyek tingkat ini akan dibuat sebuah alat kontroler *game* FPS menggunakan ATmega 32U4. Alat ini dibuat untuk menggerakkan karakter dan menembak pada *game* FPS agar pemain lebih merasakan pengalaman bermain yang lebih nyata daripada menggunakan *mouse* dan *keyboard* yang saat ini digunakan kebanyakan orang untuk memainkan *game* FPS. Gambaran umum dari perancangan yang akan dibuat adalah sebagai berikut :

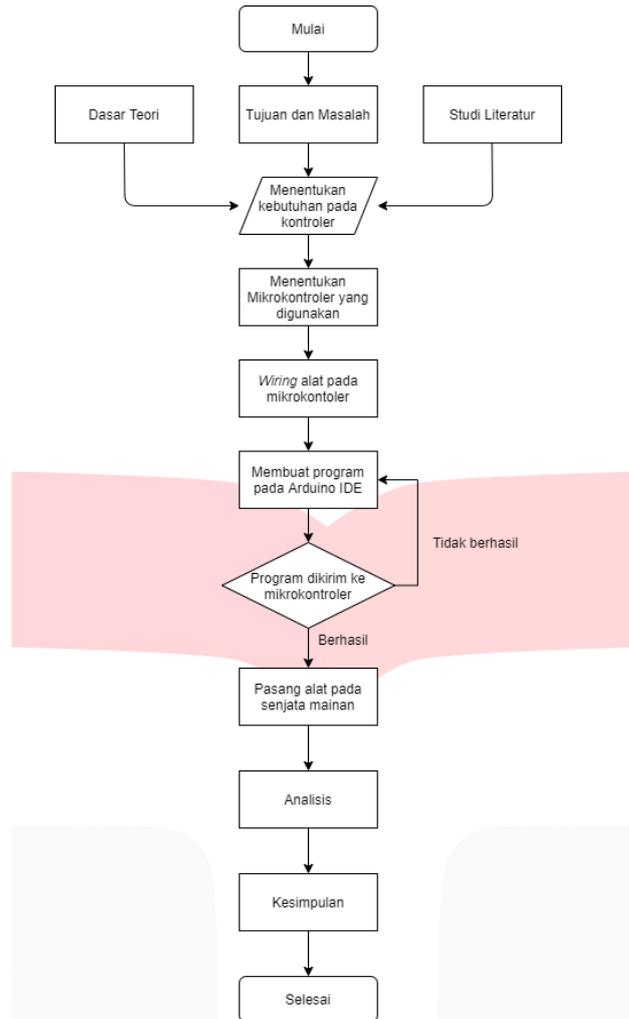


Gambar 2. 1 Skema rangkaian

Sensor yang terdiri dari *joystick* analog, *push button*, dan sensor *gyroscope* MPU6050 dirangkai ke Arduino *pro micro*. Sensor- sensor tersebut akan mengirim nilai yang nantinya akan diolah oleh mikrokontroler untuk mengirim perintah ke komputer. Mikrokontroler dan sensor-sensor tersebut dimasukkan ke dalam senjata mainan dan dihubungkan ke komputer menggunakan kabel data tipe Micro USB. Kabel data tersebut berfungsi sebagai jalur untuk mikrokontroler dan komputer berinteraksi.

2.1 Tahapan Pengerjaan

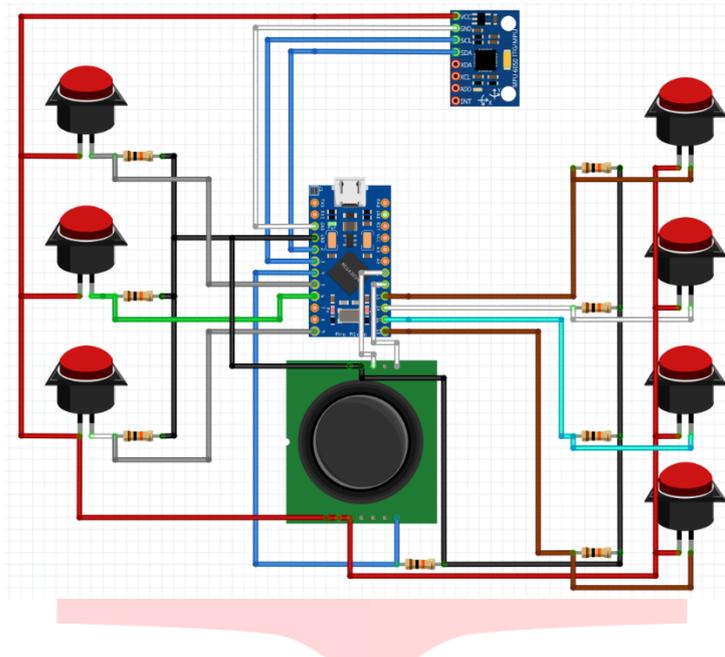
Tahapan pengerjaan alat ini dimulai dengan membaca literatur mengenai Arduino, ATmega 32U4, rangkaian Arduino *Pro micro*, sensor *gyroscope*, dan kode yang digunakan untuk memprogram alat-alat tersebut. Untuk alur pengerjaan proyek tingkat ini, dapat dilihat pada *flowchart* berikut ini :



Gambar 2. 2 Diagram Alir Tahapan Perancangan

Tahap awal dari pengerjaan proyek tingkat ini yaitu dengan menentukan tujuan dan rumusan masalah mengapa alat ini harus dibuat. Selain itu beberapa studi literatur dan dasar teori akan dikumpulkan dan dijadikan referensi untuk melakukan langkah selanjutnya yaitu menentukan kebutuhan pada kontroler. Setelah mengetahui apa saja yang dibutuhkan pada alat, langkah selanjutnya yaitu menentukan mikrokontroler apa yang akan dipilih. Pada tahap ini penentuan mikrokontroler harus sesuai dengan spesifikasi yang akan dipakai sehingga pin pada mikrokontroler tidak akan terbuang sia-sia. Selain itu ukuran mikrokontroler juga tidak boleh terlalu besar. Karena jika terlalu besar mikrokontroler tidak akan muat dimasukkan pada senjata mainan. Sehingga pada proyek tingkat ini dipilih *Arduino Pro micro* sebagai mikrokontroler yang akan digunakan karena memiliki cukup banyak pin input dan juga ukurannya yang relative kecil. Selain itu mikrokontroler ini mempunyai fitur High Speed USB 2.0 sehingga mikrokontroler dapat langsung diemulasikan sebagai *mouse* dan *keyboard* pada komputer. Hal ini dapat dilihat pada device manager pada OS *Windows* lalu expand pada bagian *keyboards* dan *Mice and others pointing devices*, maka akan terlihat perangkat *HID Keyboard Device* dan *HID-compliant mouse*. *HID (Human Interface Device)* merupakan protocol USB yang dibuat untuk menentukan perangkat yang berinteraksi sampai tingkat tertentu dengan manusia dan mentransfer data dalam sistem komputer [9]. Untuk tampilan pada Device Manager dapat dilihat pada Lampiran C.

Langkah selanjutnya yaitu merangkai *button*, sensor, dan *joystick* pada mikrokontroler. Jika semua alat sudah dipasang pada mikrokontroler, kemudian buat program untuk semua alat tersebut agar dapat bekerja dengan baik, lalu upload kode tersebut menggunakan software *Arduino IDE*. Untuk wiring diagram dapat dilihat pada gambar berikut :



Gambar 2. 3 *Wiring Diagram*

Setelah kode sukses diunggah alat bisa langsung dites dengan menghubungkan mikrokontroler ke komputer berbasis *Windows* (*Minimal Windows 7*) menggunakan kabel *micro USB*. Data yang dikirim mikrokontroler ke komputer merupakan bilangan HEX yang direpresentasikan dari program Arduino yang dibuat. Contoh bilangan HEX yang dikirim dapat dilihat pada tabel berikut :

Tabel 2. 1 Contoh kode HEX

Usage ID	Usage Name
04	<i>Keyboard a</i>
05	<i>Keyboard b</i>
16	<i>Keyboard s</i>
07	<i>Keyboard d</i>
1A	<i>Keyboard w</i>
2C	<i>Keyboard Spacebar</i>
E0	<i>Keyboard LeftControl</i>
E1	<i>Keyboard LeftShift</i>
15	<i>Keyboard r</i>

Jika alat sudah bekerja dengan baik, alat dapat dipasang pada senjata mainan yang sudah dimodifikasi. Namun jika belum, program harus dibuat ulang agar alat dapat bekerja dengan baik.

Apabila semua langkah pengerjaan sudah dilakukan, dapat dilakukan analisis pada pekerjaan yang sudah dilakukan, lalu membuat kesimpulan dari analisis yang telah dibuat.

2.2 Deskripsi Kebutuhan *Hardware* dan *Software*

Pada sub bab ini akan dijelaskan mengenai apa saja *hardware* dan *software* yang digunakan pada pengerjaan proyek tingkat ini.

2.2.1 *Hardware*

Beberapa *hardware* yang digunakan pada proyek tingkat ini adalah sebagai berikut :

1. *Arduino Pro micro*

Spesifikasi *Arduino Pro micro* yang digunakan adalah sebagai berikut :

Tabel 2. 2 Spesifikasi *Arduino Pro micro*

<i>Non-volatile Program and Data Memories</i>	<ul style="list-style-type: none">- 32KB of In-System Self-Programmable Flash- 2.5KB Internal SRAM- 1KB Internal EEPROM- Data retention: 20 years at 85°C/ 100 years at 25°C
<i>USB 2.0 Full-speed</i>	<ul style="list-style-type: none">- Complies fully with Universal Serial Bus Specification Rev 2.0- Supports data transfer rates up to 12Mbit/s and 1.5Mbit/s
<i>I/O and Packages</i>	All I/O combine CMOS outputs and LVTTL inputs
<i>Operating Voltage</i>	2.7 - 5.5V
<i>Operating Temperature</i>	Industrial (-40°C to +85°C)
<i>Maximum Frequency</i>	16MHz at 4.5V - Industrial range

2. *MPU6050*

MPU6050 merupakan sebuah modul *gyroscope* yang dapat digunakan pada perangkat *Arduino*. Pada proyek tingkat ini, modul ini digunakan untuk membaca gerakan senjata yang akan diterjemahkan kedalam bidang vertikal dan horizontal pada layer monitor komputer.

3. *Joystick Analog*

Joystick analog merupakan sebuah modul yang mempunyai dua buah nilai yang memiliki rentang sebanyak 10 bit yang dapat dibaca pada pin analog input. Pada proyek tingkat ini, modul ini digunakan untuk menggerakkan karakter pada *game*.

4. *Push Button*

Pada proyek tingkat ini, *push button* digunakan sebagai input perintah pada *game* yang fungsinya dapat dimodifikasi sesuai pada kebutuhan *game* tertentu.

5. Kabel data Micro USB

Digunakan untuk menghubungkan alat yang sudah dibuat dengan komputer.

2.2.2 Software

Beberapa *software* yang digunakan pada proyek tingkat ini adalah sebagai berikut :

1. Arduino IDE
Arduino IDE adalah *software open source* yang digunakan untuk memprogram mikrokontroler.
2. Counter Strike
Counter strike adalah sebuah *game* PC berbasis *Windows* yang bergenre *First Person Shooter* (FPS).

2.3 Blok Diagram Sistem

Secara garis besar, pada perancangan ini terdapat tiga komponen utama, yaitu input yang meliputi *button*, sensor, dan *joystick*. Lalu ada ATmega32U4 sebagai IC mikrokontroler yang terdapat pada mikrokontroler Arduino *pro micro*, dan juga PC sebagai penerima data dari ATmega32U4 yang akan digunakan untuk menggerakkan karakter pada *game* FPS. Hubungan ketiga komponen tersebut dapat diilustrasikan sebagai berikut:



Gambar 2. 4 Blok Diagram Sistem Alat

2.4 Tahapan Perancangan Alat

Tahapan perancangan kontroler *game* FPS ini dilakukan dengan metode try and error, ada beberapa tahapan yang dilakukan dalam perancangan kontroler ini yaitu sebagai berikut :

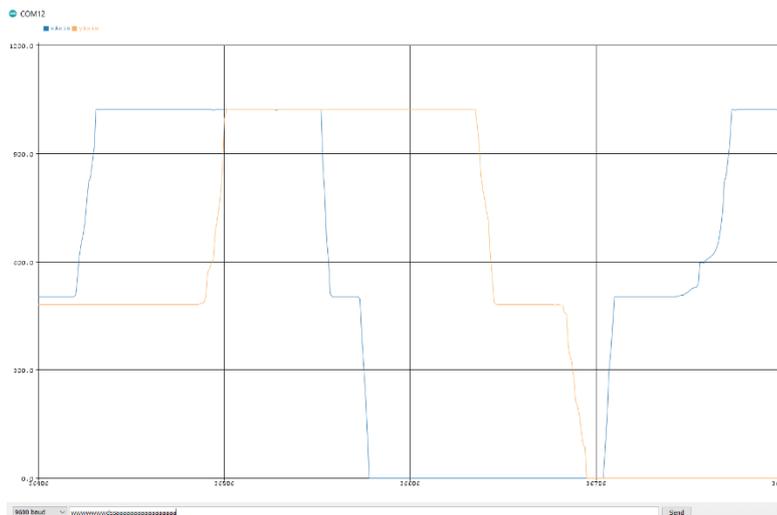
1. Menentukan spesifikasi alat sesuai kebutuhan, penentuan spesifikasi harus sesuai dengan kebutuhan alat sehingga pin tidak terbuang dengan sia-sia.
2. Menyambung *button*, sensor, dan *joystick*, dalam proses ini dapat dilakukan seperti menyambung komponen pada umumnya, tetapi menggunakan kabel jumper yang lebih panjang karena nantinya akan dipasang pada senjata mainan.
3. Memasang alat dan kontroler ke senjata mainan, pada tahap ini alat dan bahan dipasang pada senjata mainan dan ditempatkan sesuai dengan penempatan yang sudah direncanakan sebelumnya.
4. Membuat program untuk mikrokontroler (Arduino Leonardo), pada tahap ini dibuat sebuah program untuk memproses sinyal yang masuk dari alat yang sudah dipasang pada poin nomor 2. Pembuatan program digunakan untuk menentukan hal apa yang akan terjadi jika suatu pin mendapat nilai HIGH atau sebaliknya yang selanjutnya akan dikirim ke komputer. Sehingga komputer mengerti akan menekan tombol apa atau menggeser *mouse* ke titik mana.
5. Mengunggah program ke mikrokontroler, setelah kode selesai dibuat, program diunggah ke mikrokontroler menggunakan kabel data micro USB. Jika sudah terkirim dengan baik maka alat sudah siap digunakan pada komputer dengan sistem operasi *Windows* (Minimal *Windows 7*).

3. ANALISIS SIMULASI DAN PENGUJIAN ALAT

Simulasi dan pengujian alat bertujuan untuk mengetahui apakah alat yang sudah dibuat sudah berjalan dengan baik atau tidak. Hasil yang didapatkan adalah sebagai berikut :

3.1 Hasil Simulasi Program

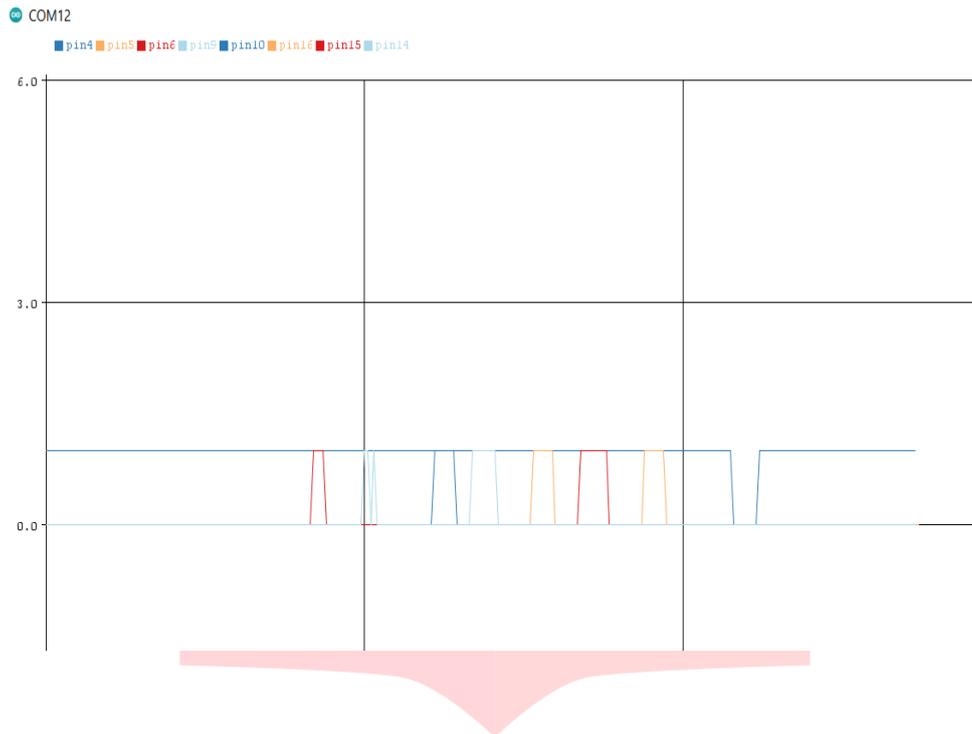
Sebelum melakukan pengujian alat pada *game*, langkah awal yang harus dilakukan yaitu menguji pada serial plotter terlebih dahulu agar dapat dilihat nilai yang dikeluarkan oleh sensor dan untuk mengetahui apakah alat sudah sesuai dengan yang diharapkan atau belum. Beberapa gambar di bawah menunjukkan hasil dari simulasi program berupa grafik *serial plotter*.



Gambar 3. 1 Grafik *serial plotter joystick* analog

Gambar 3.1 diatas merupakan gambar grafik serial plotter untuk *joystick* analog. Pada grafik tersebut terdapat dua grafik berwarna biru dan jingga. Grafik berwarna biru menunjukkan nilai xAxis dan grafik berwarna jingga menunjukkan nilai yAxis pada *joystick*.

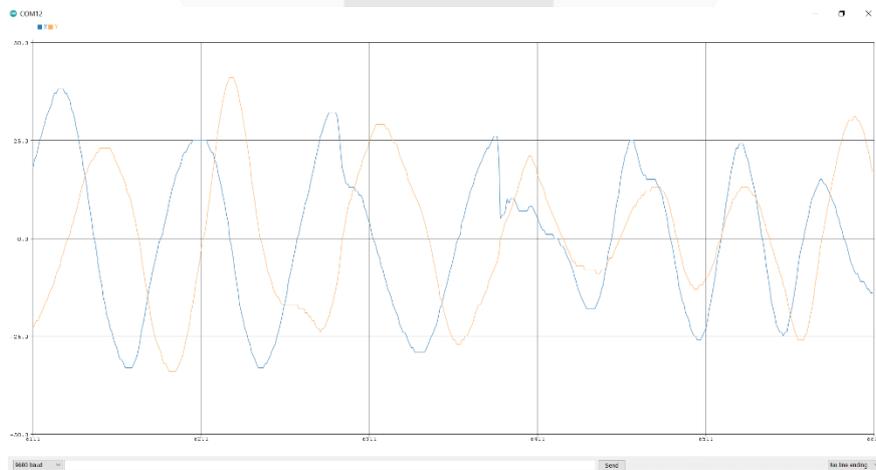
Dapat dilihat pada gambar diatas kedua grafik tersebut nilainya naik dan turun dikarenakan *joystick* digerakan ke segala arah (360°). Saat *joystick* diarahkan ke depan maka grafik warna biru akan berada di atas atau nilainya 1023 (nilai tertinggi). Lalu saat *joystick* berputar kearah antara depan dan atas, maka grafik biru dan grafik jingga akan naik dan akan berpapasan. Sedangkan apabila *joystick* diarahkan ke atas maka grafik biru akan turun dan grafik jingga akan tetap berada diatas. Lalu saat *joystick* digerakkan kearah diantara atas dan belakang, maka grafik jingga akan tetap berada di atas, tetapi grafik biru akan berada di titik bawah (0). Sedangkan pada saat *joystick* diarahkan ke belakang, maka grafik jingga akan kembali ke tengah, dan grafik berwarna biru akan kearah bawah. Kemudian saat *joystick* digerakkan kearah diantara bawah dan belakang, grafik biru dan jingga akan ke bergerak ke bawah. Saat *joystick* diarahkan ke bawah, maka grafik berwarna biru akan berada di tengah dan grafik berwarna jingga akan tetap berada di bawah. Lalu saat *joystick* diarahkan kearah diantara bawah dan depan maka grafik berwarna biru akan berada di atas dan grafik berwarna jingga berada di bawah.



Gambar 3. 2 Serial Plotter Button

Gambar 3.2 diatas merupakan gambar grafik serial plotter untuk *button*. Pada gambar tersebut terdapat delapan plot garis yang menunjukkan beberapa pin input, yaitu pin4 (klik kanan *mouse*), pin5 (klik kiri *mouse*), pin6 (spasi), pin9 (tombol Shift), pin10 (tombol Ctrl), pin16 (tombol 'r'), pin15 (tombol 'b'), dan pin14 (scroll *mouse*).

Dapat dilihat pada gambar diatas bahwa semua pin hanya memiliki dua kondisi yaitu 0 dan 1 (high dan low). Hal ini dikarenakan semua pin tersebut merupakan pin input digital. Pada gambar tersebut urutan untuk menekan tombol yaitu pin6, pin9, pin10, pin14, pin5, pin15, pin16, dan yang terakhir pin4. Semua pin tersebut memiliki kondisi default active low, artinya saat tombol tidak ditekan, maka nilai yang dibaca adalah 0, tetapi berbeda kasusnya pada pin4, dimana pin tersebut berasal dari modul *joystick* analog yang memiliki kondisi default active high. Sehingga pada saat tombol tidak ditekan nilai awalnya akan menjadi 1 dan saat ditekan nilainya akan menjadi 0.



Gambar 3. 3 Grafik serial plotter sensor gyroscope

Gambar 3.3 diatas merupakan gambar grafik serial plotter untuk pergerakan *mouse* hasil dari sensor *gyroscope*. Pada gambar tersebut terdapat dua plot garis yaitu garis berwarna biru untuk sumbu X dan garis berwarna jingga untuk sumbu Y.

Grafik tersebut menunjukkan banyaknya jumlah pixel yang harus cursor gerakkan secara real time. Grafik tersebut diambil dengan cara menggerakkan senjata mainan secara acak yang kemudian sensor gyro akan mengirim banyaknya data yang harus digerakkan.

3.2 Hasil Pengujian Alat

Setelah melakukan simulasi program pada serial *plotter*, langkah selanjutnya adalah melakukan pengujian alat pada in *game* secara langsung. Untuk pengujian ini menggunakan salah satu *game* FPS yang paling banyak dimainkan saat ini, yaitu Counter Strike. Pengujian ini dilakukan untuk melihat apakah alat yang sudah dibuat sudah sesuai dengan yang diharapkan atau belum. Cara pengujian alat ini dilakukan dengan menekan beberapa *button* pada senjata dan melihat apakah karakter pada *game* sudah bergerak sesuai dengan semestinya atau belum. Berikut beberapa hasil yang didapatkan :



Gambar 3. 4 Tombol *trigger* sebelum ditekan

Pada gambar diatas terlihat bahwa jari belum menekan *trigger* pada senjata dan dapat terlihat pada layar monitor senjata tersebut tidak menembakkan apa-apa, hal tersebut dapat dilihat dari senjata yang tidak mengeluarkan percikan api.



Gambar 3. 5 Tombol *trigger* setelah ditekan

Pada gambar diatas memperlihatkan senjata pada *in game* sudah menembakkan peluru saat jari sudah menekan *trigger* pada kontroler senjata, hal ini terlihat ada percikan api yang keluar dari moncong senjata pada *in game* tersebut.



Gambar 3. 6 Kondisi saat *joystick* belum digerakkan

Pada gambar diatas terlihat bahwa jempol kiri belum menggerakkan *joystick* tersebut dan karakter belum berpindah atau tidak bergerak.



Gambar 3. 7 Kondisi saat *Joystick* digerakkan

Berbeda dengan gambar 3.6, pada gambar 3.7 di atas terlihat bahwa posisi karakter sudah berpindah kearah belakang saat *joystick* sudah digerakkan.



Gambar 3. 8 Kondisi saat *button* pin6 belum ditekan

Pada gambar di atas terlihat bahwa tombol pada pin *input* pin6 (tombol spasi) yang ada pada kelingking kanan belum ditekan, dan karakter pada *in game* belum bergerak atau masih dalam keadaan diam.



Gambar 3. 9 Kondisi saat *button* pin6 sudah ditekan

Pada gambar 3.9 terlihat pada *in game* bahwa karakter sedang melompat saat *button* pin6 (tombol spasi) ditekan. Hal ini dapat dilihat dari sudut pandang karakter yang lebih tinggi daripada sudut pandang karakter yang ada di gambar 3.8.



Gambar 3. 10 Kondisi saat *buton* pin10 sebelum ditekan

Pada gambar 4.10 di atas terlihat bahwa pin input pin10 (tombol Ctrl) yang ada pada jari jempol tangan kanan belum ditekan. Terlihat pada karakter di in *game* tidak melakukan apapun atau karakter diam.



Gambar 3. 11 Kondisi saat *buton* pin10 sedang ditekan

Pada gambar 4.11 di atas, terlihat bahwa jempol tangan kanan sudah menekan dan menahan pin input pin10 (tombol Ctrl). Hal tersebut memicu karakter yang ada pada in *game* untuk melakukan posisi jongkok selama *buton* pin10 ditekan.

4. KESIMPULAN

Berdasarkan hasil analisis dan pengujian yang telah dilakukan dapat disimpulkan bahwa senjata mainan yang telah dimodifikasi telah dapat bekerja dengan baik dengan parameter seperti variabel yang sudah berubah pada tiap variabelnya, hal ini dapat dilihat pada serial monitor atau serial plotter. Seperti nilai *joystick* yang memiliki dua sumbu, sumbu X dan sumbu Y yang memiliki rentang nilai dari 0 s/d 1023 saat digerakkan. Lalu ada sensor *gyroscope* yang memiliki sumbu X dan sumbu Z yang memiliki nilai -300 s/d -400 saat tidak digerakkan, dan dapat berubah menjadi -1023 s/d 1023 ketika digerakkan. Lalu ada beberapa push *button* yang memiliki nilai default 0 saat tidak ditekan, ketika ditekan nilai akan berubah menjadi 1. Kemudian angka-angka tersebut akan mengatur mikrokontroler untuk mengirim perintah klik kiri, klik kanan, memencet tombol 'B', dsb.

REFERENSI

- [1] N. Dianty, "Digital Repository Unila," Universitas Lampung, 2012. [Online]. Available: <http://digilib.unila.ac.id/12811/>. [Diakses March 2020].
- [2] Y. Nuruzahha, "Open Library Telkom University," Universtas Telkom, 2016. [Online]. Available: <https://openlibrary.telkomuniversity.ac.id/pustaka/121916/aplikasi-game-fps-horror-of-campus-.html>. [Diakses March 2020].
- [3] A. E. R. E. Praja Irwandi, "PERANCANGAN *GAME* FIRST PERSON SHOOTER (FPS) "BOAR HUNTER" BERBASIS VIRTUAL REALITY," *Jurnal Rekursif*, vol. 04, p. 69, 2016.
- [4] A. Kurniawan, "UPT Perpustakaan Universitas Sebelas Maret," Universitas Sebelas Maret, 2011. [Online]. Available: <https://digilib.uns.ac.id/dokumen/detail/19281/Perancangan-Sistem-Kendali-Otomatis-pada-Lantai-Getar-Berbasis-Mikrokontroler-Avr-Atmega32-Studi-Kasus-Laboratorium-Perancangan-Sistem-Kerja-dan-Ergonomi>. [Diakses November 2020].
- [5] Admin, "Immersa Lab," February 2018. [Online]. Available: <https://www.immersa-lab.com/pengertian-gyroscope-dan-cara-kerjanya.htm>. [Diakses March 2020].
- [6] Admin, "Microchip," Microchip, [Online]. Available: <https://www.microchip.com/wwwproducts/en/ATmega32u4>. [Diakses 2020].
- [7] Atmel, "ATmega16U4/ATmega32U4 Datasheet," Atmel, 2016.
- [8] Admin, "EEPowerr," EETech Media, [Online]. Available: https://eepower.com/resistor-guide/resistor-applications/pull-up-resistor-pull-down-resistor/#Pull-up_resistor_definition. [Diakses March 2020].
- [9] S. Labs, "HUMAN INTERFACE DEVICE TUTORIAL," Silicon Labs, Austin , 2011.
- [10] USB.org, "HID Usage Tables for Universal Serial Bus (USB)," USB Implementers Forum, 2020.