



REPUBLIK INDONESIA
KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

SURAT PENCATATAN CIPTAAN

Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan : EC00202020345, 30 Juni 2020

Pencipta

Nama : **Dr.Eng. Romy Budhi Widodo, Evans Jahja dkk**
Alamat : Jl. Diponegoro 12, Girimulyo, Karangploso, Malang, -, Jawa Timur, -
Kewarganegaraan : Indonesia

Pemegang Hak Cipta

Nama : **Universitas Ma Chung**
Alamat : Villa Puncak Tidar N-1 Malang, -, Jawa Timur, 65151
Kewarganegaraan : Indonesia
Jenis Ciptaan : **Program Komputer**
Judul Ciptaan : **Wifi-based Gyro Mouse**
Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia : 1 Juni 2020, di Malang
Jangka waktu perlindungan : Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor pencatatan : 000194155

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.
Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.



a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA
DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL

Dr. Freddy Harris, S.H., LL.M., ACCS.
NIP. 196611181994031001

LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Dr.Eng. Romy Budhi Widodo	Jl. Diponegoro 12, Girimulyo, Karangploso, Malang
2	Evans Jahja	Green Garden BLK X-2/8 RT.14 RW.03 Kedoyo Utara, Kebon Jeruk, Jakarta Barat



2020

**Software
Manual
Book:**

**“Wifi-based
Gyro Mouse”**

**Oleh:
Romy Budhi Widodo & Evans Jahja**

Universitas Ma Chung



Petunjuk sistem Gyro ESP32

Keterangan singkat

Sistem ini menangkap orientasi perangkat dan men-transmit ke PC melalui WiFi. Microcontroller yang digunakan adalah ESP32, yang terhubung ke Modul Gyro (Accelerometer) dan dua tombol.

Koneksi

Koneksi ESP32 ke modul Gyro

Notasi menunjukkan dari modul Gyro *ke* ESP32

Gyro	ESP32
VIN	3.3V
GND	GND
TX	RX
RESET	GND(*)

(*) Koneksi ini hanya digunakan sebelum *upload*. Saat *upload*, gyro tidak boleh terhubung dengan ESP32, karena akan mengganggu proses *upload*. Pin TX -> RX perlu dilepas, atau Gyro dapat dinonaktifkan dengan menyambungkan: Reset pada Gyro ke Ground.

Koneksi ESP32 dengan tombol

Dua pin digunakan oleh ESP32 untuk membaca tombol dari pengguna. Pin yang digunakan untuk pembacaan tombol adalah pin 34 dan pin 35. Untuk mengubah pin ini, silahkan ubah:

```
// gyro.ino
#define BTN_0 34
#define BTN_1 35
```

sesuai dengan kebutuhan.

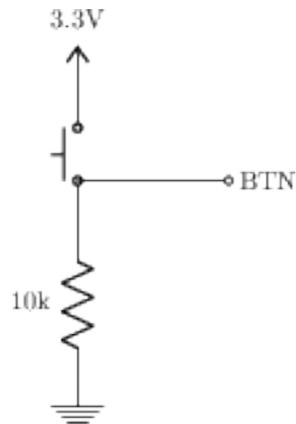


Figure 1: Rangkaian tombol

Gambar di atas adalah contoh salah satu tombol. Resistor ($R=10k$) digunakan sebagai *pull down*, agar BTN bernilai 0 ketika tombol tidak ditekan. Ketika tombol ditekan, BTN bertegangan VCC, atau 3.3V. Kedua tombol (BTN_0 dan BTN_1) memiliki rangkaian yang sama.

Pemrograman ESP32

ESP32 (Arduino IDE)

ESP32 diprogram menggunakan Arduino IDE (<https://www.arduino.cc/en/main/software>).

Setelah diinstall, Arduino IDE membutuhkan 'Library' untuk berkomunikasi dengan ESP32. Ini bisa didapatkan dengan membuka Boards Manager (Tools -> Board... -> Boards Manager).

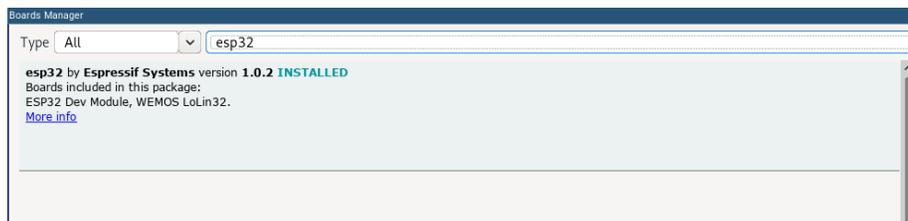


Figure 2: Boards Manager

Pada Boards Manager, ketik "esp32" di kotak pencarian, dan install 'Library' esp32.

Setelah terinstall, pada pilihan Boards, pilihlah "ESP32 Dev Module".

Wifi-based Gyro Mouse

Berikut beberapa informasi relevan pada menu Tools yang digunakan:

Upload Speed: 921600
CPU Frequency: 240MHz
Flash Frequency: 80MHz
Flash Mode: QIO
Flash Size: 4MB (32Mb)
Partition Scheme: Default 4MB with spiffs

Pastikan juga Port dari menu Tools terpilih dengan benar. Ini bergantung pada komputer yang digunakan, dan dapat berubah setiap kali ESP32 disambungkan ke komputer melalui USB.

Penjelasan kode

Kode Arduino terdapat dalam file `gyro.ino`. Berikut kode Arduino beserta penjelasannya.

Bagian inisialisasi

```
#include <WiFi.h>
#include <WiFiClient.h>
#include <WiFiAP.h>
#include <WiFiUDP.h>

#define BTN_0 34
#define BTN_1 35

const char *ssid = "gyro";
//const char *password = "yourPassword";

WiFiUDP udp;

const char * udpAddress = "192.168.4.255";
const int udpPort = 3333;
unsigned char buf[50];
```

Bagian ini merupakan bagian inisialisasi. Baris-baris `#include` di atas memuat file-file terkait fungsi WiFi milik ESP32.

`#define BTN_0` dan `BTN_1` digunakan untuk men-set pin manakah pada ESP32 yang digunakan untuk fungsi klik kiri (`BTN_0`) dan klik kanan (`BTN_1`). Pada kode ini, pin 34 digunakan untuk klik kiri dan pin 35 digunakan untuk klik kanan.

Wifi-based Gyro Mouse

`const char *ssid` digunakan untuk menyetel ssid atau nama akses poin WiFi yang akan dibuat oleh ESP32. ESP32 berfungsi sebagai akses poin, dan komputer yang hendak mendapatkan informasi sensor butuh terkoneksi ke akses poin yang dibuat ini. Password dapat digunakan dengan cara meng-*uncomment* `const char password`, mengisi password yang diinginkan, dan mengubah kode `WiFi.softAP()` di bagian `setup()`.

`udpAddress` dan `udpPort` digunakan sebagai tujuan paket UDP yang berisi data sensor. Library WiFi milik ESP32 ketika bertindak sebagai Access Point (AP), akan secara *default* menggunakan Network Address 192.168.4.0/24. Perubahan ini dapat diatur menggunakan konfigurasi yang di luar *scope* dokumentasi ini. Paket UDP ditujukan ke *broadcast address* dari network ini, yaitu 192.168.4.255. Port yang dipilih (3333) boleh diubah sesuai kebutuhan, selama aplikasi penerima di PC menggunakan port yang sama.

`unsigned char buf[50]` merupakan *buffer* sementara untuk mengolah data sensor yang diterima dari modul gyro untuk ditransmisi ulang melalui UDP.

Bagian `setup()`

```
void setup() {
  // put your setup code here, to run once:
  Serial.begin(57600);

  // Wifi
  WiFi.softAP(ssid);

  Serial.setTimeout(2);

  // Setup buttons
  pinMode(BTN_0, INPUT);
  pinMode(BTN_1, INPUT);
}
```

`Serial.begin(57600)` memulai komunikasi serial dengan modul gyro, dengan *baud rate* 57600bps, sesuai dengan konfigurasi awal dari modul gyro.

`WiFi.softAP()` merupakan fungsi yang digunakan untuk membuat ESP32 bertindak sebagai *Access Point*. Fungsi ini dapat dipanggil dalam dua bentuk:

1. `WiFi.softAP(ssid)`
2. `WiFi.softAP(ssid, password)`

tergantung apakah kita ingin AP memiliki password atau tidak.

`Serial.setTimeout(2)` digunakan untuk mengkonfigurasi pembacaan serial dari modul gyro untuk menunggu maksimal 2 millisecond, sehingga pembacaan String berhenti ketika 2 millisecond terlampaui, atau ketika karakter terminasi *carriage return* (`\r`) ditemukan.

Wifi-based Gyro Mouse

`pinMode(BTN_0, INPUT)` dan `pinMode(BTN_1, INPUT)` digunakan untuk mengatur kedua pin tersebut ke dalam mode `INPUT` atau pembacaan.

Bagian `loop()`

```
void loop() {
  while(1) {
    String str = Serial.readStringUntil('\r') + getButtons() + "\n";
    //String str =
    str.getBytes(buf, 50);
    if (str.length() > 10) {
      udp.beginPacket(udpAddress, udpPort);
      udp.write(buf, str.length());
      udp.endPacket();
    }
    delay(1);
  }
}
```

Pada tahap ini, pengulangan dilakukan. Pertama, kita coba baca data dari modul gyro menggunakan perintah `Serial.readStringUntil('\r')`. Pembacaan ini akan menghasilkan suatu string panjang yang berisi informasi *Yaw*, *Pitch*, dan *Roll* (YPR). Data yang diberikan oleh sensor berbentuk seperti ini:

```
#YPR=-108.52,41.19,6.15
```

Terdapat tiga angka yang dipisahkan oleh koma, yang masing-masing merupakan nilai *Yaw*, *Pitch*, dan *Roll* secara berurutan.

Selain Ketiga nilai ini, kita menambahkan fungsi `getButtons()` setelah nilai YPR tersebut. `getButtons()` dibahas pada bagian setelah ini.

Pada tahap ini, kita memiliki variabel `str` yang berisikan informasi YPR dan *state* tombol yang diwakilkan dengan dua angka – untuk tombol kiri (`BTN_0`) dan tombol kanan (`BTN_1`). Kedua angka ini dapat bernilai 0, yang artinya tidak ditekan, atau 1, yang berarti tombol tersebut ditekan. Variabel `str` dalam keadaan kedua tombol tidak ditekan tampak sebagai berikut:

```
#YPR=-108.52,41.19,6.15,0,0
```

`str.getBytes` merupakan fungsi yang memindahkan isi dari *Object String* yang merupakan kelas C++ ke bentuk `byte array` yang dapat ditransmisi dalam paket UDP. Isi dari variabel `str` dimasukkan ke variabel `buf` dengan jumlah *bytes* maksimal 50.

`if` di sini dipakai untuk mendeteksi ketika data yang kita terima panjangnya di bawah 10 karakter, yang menandakan bahwa komunikasi antara ESP32 dan modul gyro terputus. Paket UDP hanya dikirim ketika jumlah data lebih dari 10 karakter. Tiga baris setelahnya mengirimkan paket UDP ke *address* dan *port* tujuan dengan *buf* sebagai datanya.

Kita lakukan `delay` selama 1 millisecond agar sistem lebih stabil.

**)Function `loop()` seharusnya berisi kode yang dijalankan berulang-ulang. Tetapi karena alasan teknis, menggunakan `while(1)` di dalam fungsi `loop()` menghasilkan performa yang lebih baik.*

Bagian `getButtons()`

```
String getButtons() {  
    char buf[8];  
    sprintf(buf, "%d,%d", digitalRead(BTN_0), digitalRead(BTN_1));  
    String result = String(buf);  
    return result;  
}
```

`getButtons` berfungsi untuk mengetahui apakah tombol sedang ditekan atau tidak melalui fungsi `digitalRead`, dan hasilnya dimasukkan ke dalam `String` menggunakan perintah `sprintf`, yang akan memasukkan angka 0 atau 1, beserta karakter koma `,` yang memisahkan angka-angka tersebut.

Pemrograman C#

Aplikasi di PC dibuat dalam bahasa pemrograman C# dengan Visual Studio 14 (VS 2015). Aplikasi ini berfungsi untuk menghubungkan PC dengan sistem gyro melalui WiFi, menangkap data sensor dari sistem gyro, mengubah informasi sensor ke pergerakan kursor mouse dan *mouse click*, dan merekam pergerakan kursor ke *file .csv*.

Berikut tampilan aplikasi C#.

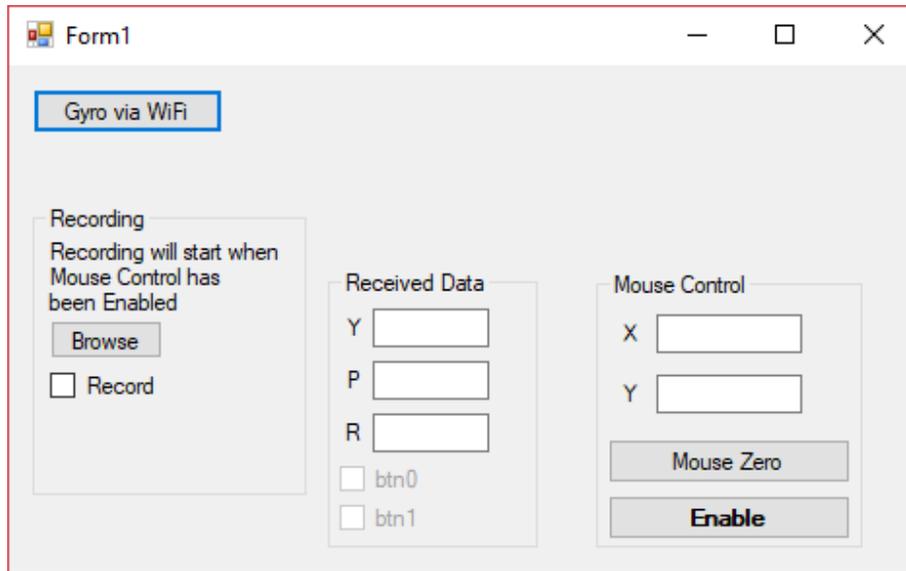


Figure 3: Tampilan awal aplikasi

Tampilan di atas merupakan tampilan awal ketika aplikasi baru dijalankan. Cara menggunakan aplikasi adalah sebagai berikut:

1. Hubungkan PC dengan *Access Point* 'gyro'.
2. Tekan tombol 'Gyro via WiFi'. *Received Data* akan terisi oleh data seperti pada "Tampilan setelah aplikasi terhubung dengan sistem gyro"
 Jika PC berhasil terhubung, maka *Received Data* akan terisi angka-angka yang didapatkan dari sensor. Jika data tidak muncul, kemungkinan koneksi antara Gyro dan ESP32 terputus, terindikasi dengan lampu di Gyro kedip-kedip atau mati. Pada modul gyro, seharusnya lampu menyala terus.
3. Atur sehingga posisi gyro berada pada posisi yang nyaman, dan tekan tombol *Mouse Zero*. Tombol ini akan me-reset nilai YPR ke nilai nol pada posisi tersebut, seperti pada "Tampilan setelah tombol *Mouse Zero* ditekan"
4. Tombol *Enable* dapat ditekan dan kursor akan mengikuti pergerakan sensor gyro seperti pada "Tampilan setelah tombol *Enable* ditekan". Jika *recording* diharapkan, maka **sebelum** tombol *Enable* ditekan, kita perlu memilih lokasi penyimpanan dengan menekan tombol *Browse* dan men-cek *checkbox Record*, seperti pada "Tampilan *Browse*, untuk memilih lokasi penyimpanan *Recording*" dan "Tampilan setelah *checkbox Record* ditekan".
5. Program dapat dihentikan dengan menekan tombol *Enable*, atau dengan menekan tombol *Esc* di keyboard ketika layar aplikasi terfokus. Jika *record* dilakukan, maka ketika menghetikan program, data akan tersimpan.

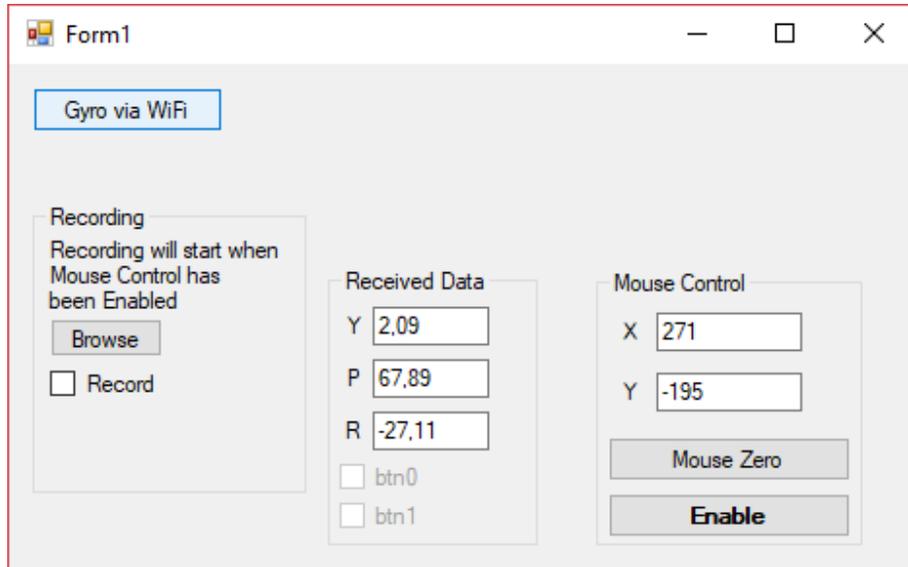


Figure 4: Tampilan setelah aplikasi terhubung dengan sistem gyro

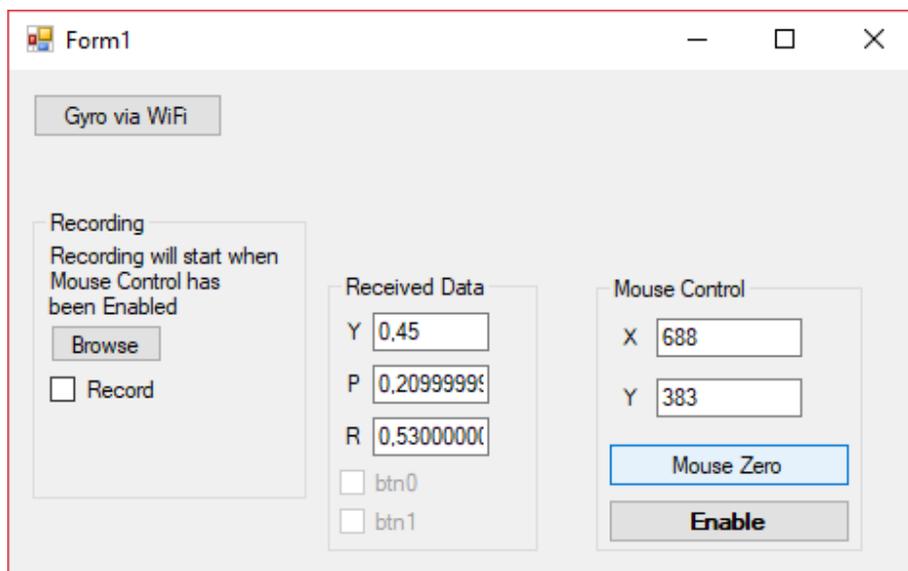


Figure 5: Tampilan setelah tombol Mouse Zero ditekan

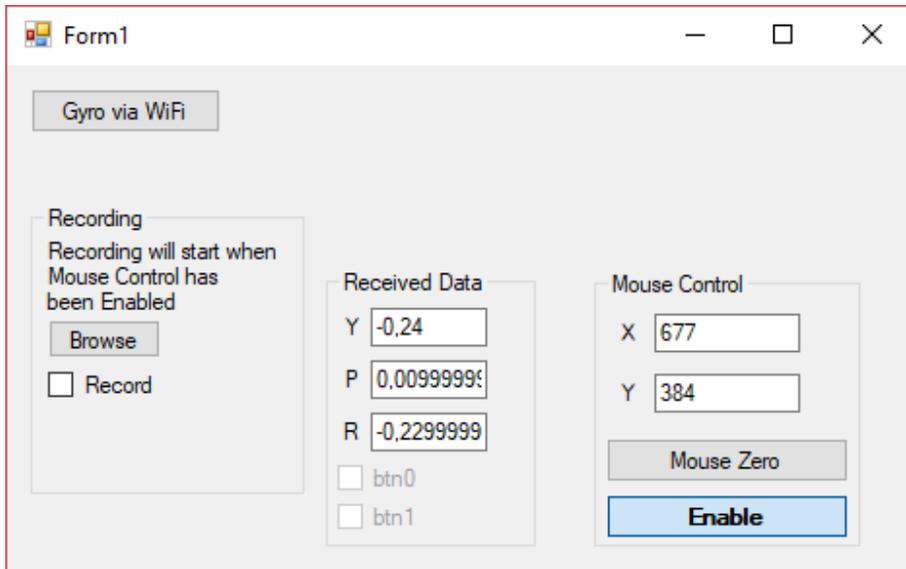


Figure 6: Tampilan setelah tombol Enable ditekan

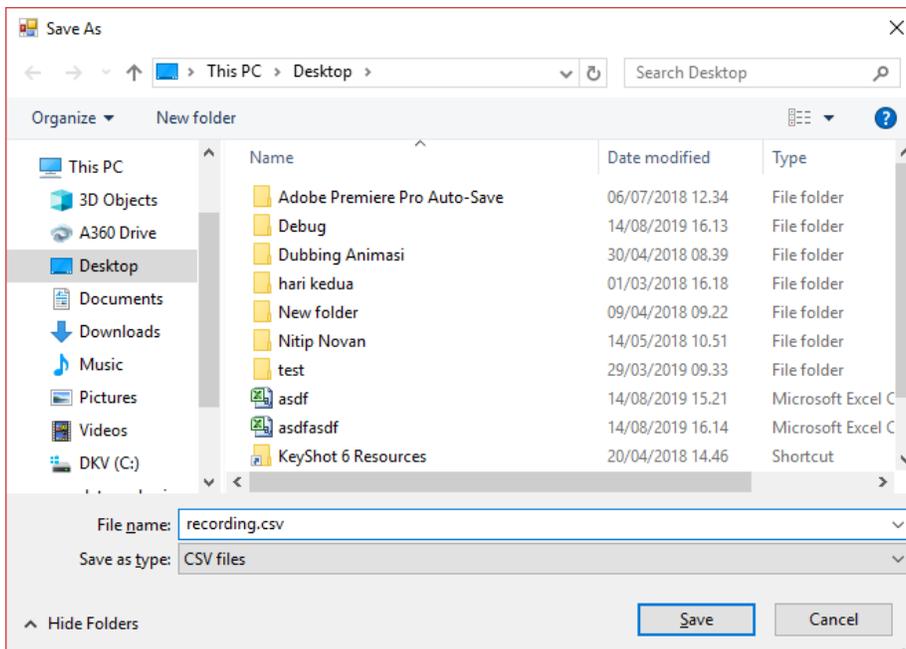


Figure 7: Tampilan Browse, untuk memilih lokasi penyimpanan Recording

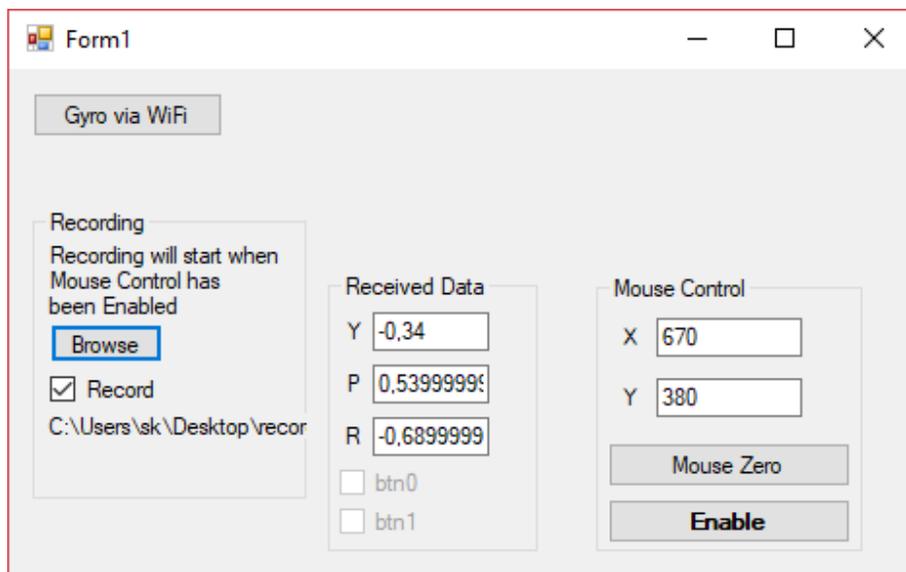


Figure 8: Tampilan setelah checkbox Record ditekan