Laporan Teknis

Bad RearView Mirror Rejection System

**Chapter 1. Development**

**1.1 Platform.**

Dalam proyek ini menggunakan platform yang bersifat opensources, kecuali dalam mendesain bagian mekanis.Keuntungan dalam menggunakan platform yang bersifat opensources antara lain:

* Gratis dan Legal.
* Dukungan penuh komunitas.
* Kemudahan modifikasi software.
* Kemampuan software opensources untuk development.

Untuk pihak pengembang lain yang berencana ikut mengembangkan dapat mengambil seluruh proyek di repository proyek berbasis Git di alamat:

<https://github.com/mekatronik-achmadi/spion_selector/>

**1.2 Software.**

Berikut adalah daftar aplikasi yang digunakan dalam proyek ini:

* LinuxMint adalah operating system yang berbasis kernel Linux dan paket manajemen Debian/Ubuntu. Disarankan untuk menggunakan sistem Graphical User Interface (GUI) jenis KDE, Cinnamon, maupun MATE. LinuxMint dapat di download di alamat:

<http://www.linuxmint.com/download.php>

* Selain menggunakan LinuxMint, alternatif operating system lain yang disarankan adalah:
  + Lubuntu dengan GUI berjenis LXDE. Dapat didownload

<https://help.ubuntu.com/community/Lubuntu/GetLubuntu>

* + Raspbian yang dijalankan di board RaspberryPi. Dapat di didownload di

<http://www.raspberrypi.org/downloads/>

Operating system Raspbian dikhususkan untuk dijalankan di sistem tertanam (Embedded System) berbasis RaspberryPi, bukan untuk komputer/laptop umum. Raspbian merupakan operating system berbasis Debian sehingga secara teknis instalasinya akan sama dengan operating system lainnya. Namun tidak disarankan menggunakan Debian maupun turunan langsungnya (kecuali Raspbian) mengingat versi software yang tersedia kurang update.

Untuk mengupdate daftar paket dari repository yang telah diatur, hubungkan LinuxMint ke internet dan gunakan perintah berikut di terminal:

**sudo apt-get update**

**1.3 Octave**

GNU Octave adalah bahasa interpretasi (kebalikan dari kompilasi) level tinggi,utamanya digunakan untuk komputasi numerik. GNU Octave menyediakan kemampuan untuk menemukan solusi numerik baik untuk masalah linear maupun non-linier, termasuk juga untuk experimen numerik. Selain itu juga menyediakan kemampuan grafis yang memadai untuk visualisasi dan manipulasi data. Octave secara umum digunakan melalui interface terminal yang interaktif, tapi dapat pula digunakan untuk program non-interaktif. Bahasa interpretasi Octave sangat mirip dengan Matlab sehingga skrip dari kedua software tersebut dapat saling bertukar. Octave memiliki kemampuan setara dengan Matlab,sekalipun belum memiliki fitur seperti Simulink dan GUI.

Octave pertama kali ditulis oleh John W. Eaton ditahun 1988 dalam bahasa C++. Octave dirilis oleh GNU Project, sebuah proyek international pengembangan software gratis dan opensource yang dipimpin oleh R. Stallman.Kini Octave telah menjadi software yang dapat dijalankan di Linux, Windows, Mac,maupun Android.Octave tersedia dalam 19 bahasa berbeda dengan English US sebagai bahasa default.Octave dirilis sebagai software OpenSource dengan lisensi GNU GPL sehingga bersifat legal dan gratis.Penjelasan lebih lanjut akan dilanjutkan di bab Octave tersendiri.

**1.4 OpenCV**

Pustaka OpenCV (Open Computer Vision) merupakan pustaka pemrograman berbasis C/C++/Python yang berisi fungsi-fungsi untuk akuisisi dan pengolahan citra.Pustaka OpenCV juga merupakan proyek opensource yang bersifat gratis.Saat ini pustaka OpenCV telah diterapkan di banyak website dan aplikasi mobile untuk deteksi wajah dan penjejak warna.Dalam proyek ini, OpenCV tidak digunakan untuk mengolah gambar namun sebatas untuk mengambil gambar. Bersama dengan Qt (C++), OpenCV ini dibentuk menjadi satu aplikasi untuk berkomunikasi dengan kamera melalui jalur USB dan mengambil gambar. Penjelasan lebih lanjut akan dilanjutkan di bersama dalam bab Octave tersendiri.Lebih lanjut tentang opencv dapat dikunjungi di situs resmi:

<http://opencv.org/>

**1.5 GCC**

GCC (GNU Compiler Collection) merupakan koleksi kompiler yang di rilis oleh proyek GNU.Kompiler merupakan program yang mengkonversi kode sumber menjadi program siap jalan. Saat ini GCC mendukung bahasa C, C++, Java, dan banyak bahasa lain. Selain itu GCC juga mendukung banyak arsitektur seperti i386, amd64, armhf, arm-eabi, dan banyak arsitektur lain Dalam proyek ini digunakan dua jenis varian GCC, yaitu g++ dan arm-gcc-none-eabi. Varian g++ digunakan untuk mengkompilasi program berbasis OpenCV untuk pengambilan gambar. Sedangkan varian arm-gcc-none-eabi digunakan untuk mengkompilasi program yang akan ditanamkan di sirkuit pengendali tangan robot.Penjelasan lebih lanjut tentang arm-gcc-none-eabi akan dilanjutkan di bersama dalam bab RTOS tersendiri.Lebih lanjut tentang GCC dapat dikunjungi di situs resmi:

<https://gcc.gnu.org/>

**1.6 Qt Creator**

Qt Creator merupakan salah satu IDE (Integrated Development Environment) yang digunakan secara luas terutama untuk pengembangan software berbasis Qt.Qt sendiri adalah kerangka kerja (framework) pengembangan software berbasis opensource yang memiliki fitur terlengkap dibandingkan framework lain sperti GTK,FLTK,dst.Qt Creator merupakan IDE untuk proses penulisan dan pengecekan program, namun membutuhkan kompiler seperti GCC, g++, maupun arm-gcc untuk dapat mengkompilasi. Dalam proyek ini Qt Creator digunakan dalam penulisan program untuk pengambilan gambar dan program yang ditanamkan di dalam chip.Lebih lanjut tentang Qt Creator dapat dikunjungi di situs resmi:

<http://qt-project.org/wiki/Category:Tools::QtCreator>

**1.7 EAGLE**

EAGLE (Easy Applicable Graphical Layout Editor) adalah software untuk mendesain papan circuit.Desain yang dibuat di EAGLE berupa skema dan layout.Versi yang dapat di instal secara gratis adalah versi Light dengan batasan:

* Luas maximal papan adalah 100mmx80mm.
* Hanya dapat menggunakan 2 Lapisan, yaitu Top dan Bottom.
* Fitur AutoRoute tidak tersedia.

Namun diluar batasan tadi, fitur EAGLE dapat digunakan secara penuh, terutama ketersedian pustaka komponen yang sangat lengkap.Penjelasan lebih lanjut akan dilanjutkan di bab Elektronik tersendiri. Lebih lanjut tentang EAGLE dapat dikunjungi di situs resmi:

<http://www.cadsoftusa.com/>

**1.8 SolidWork**

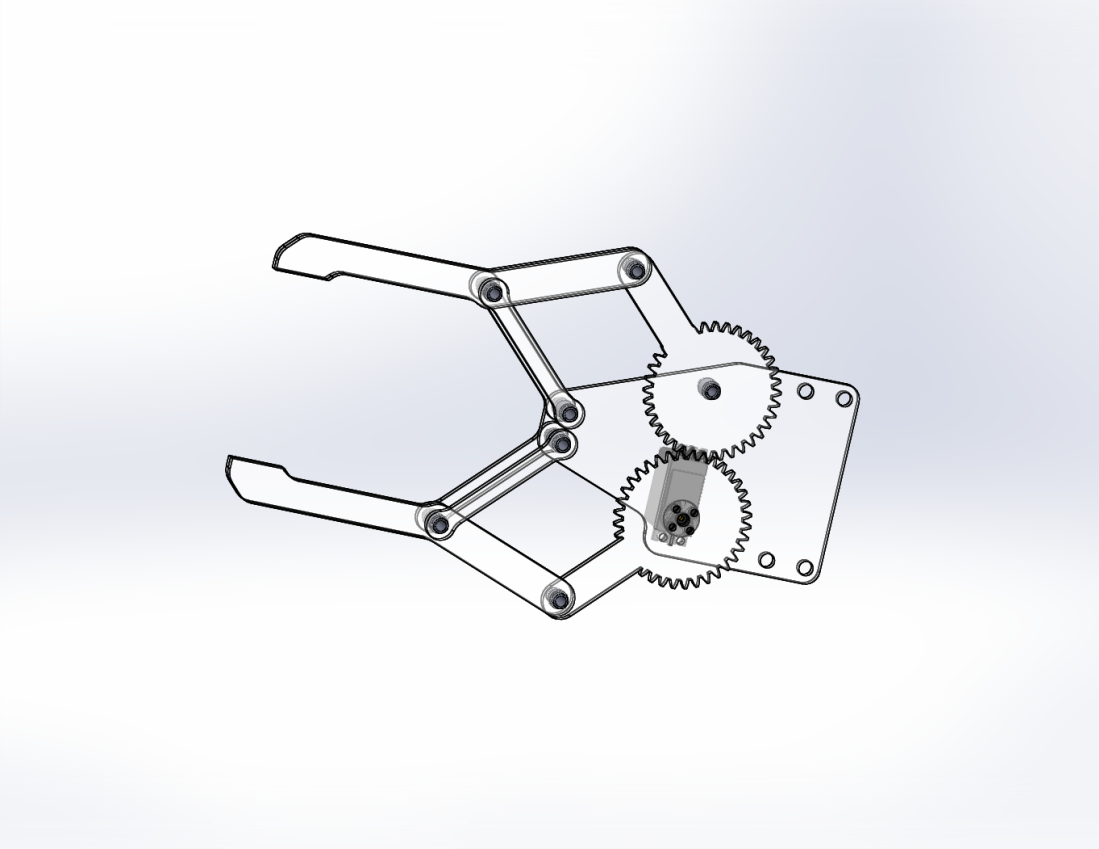
SolidWork merupakan salah satu software CAD (Computer Aided Design) yang telah populer di perusahan-perusahan besar.Software ini digunakan untuk mendesain sistem fisik dan mekanis dengan dukungan pustaka yang sangat lengkap. Dalam proyek ini, software ini digunakan untuk mendesain sistem mekanis tangan robot. Versi yang digunakan adalah 2013 dan SolidWork tergolong software yang tidak *Backward Compatible*. Software ini merupakan satu-satunya software berbayar yang digunakan dalam proyek ini. Pembahasan lebih lanjut hanya akan difokuskan pada desain mekanisnya. Namun tentang software tidak dibahas lebih jauh mengingat sifatnya yang tidak gratis dan dijalankan di platform Windows 7.Lebih lanjut tentang solidwork dapat dikunjungi di situs resmi: <http://www.solidworks.com/>

**1.9 License**

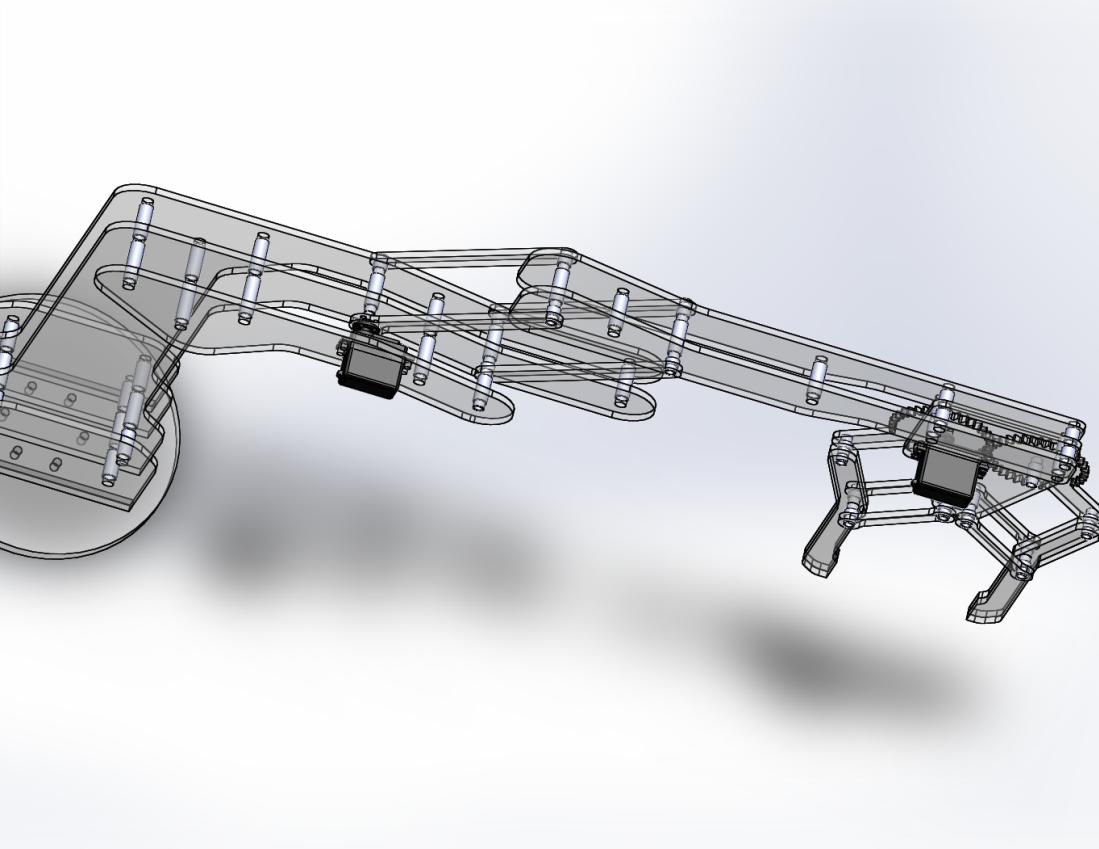
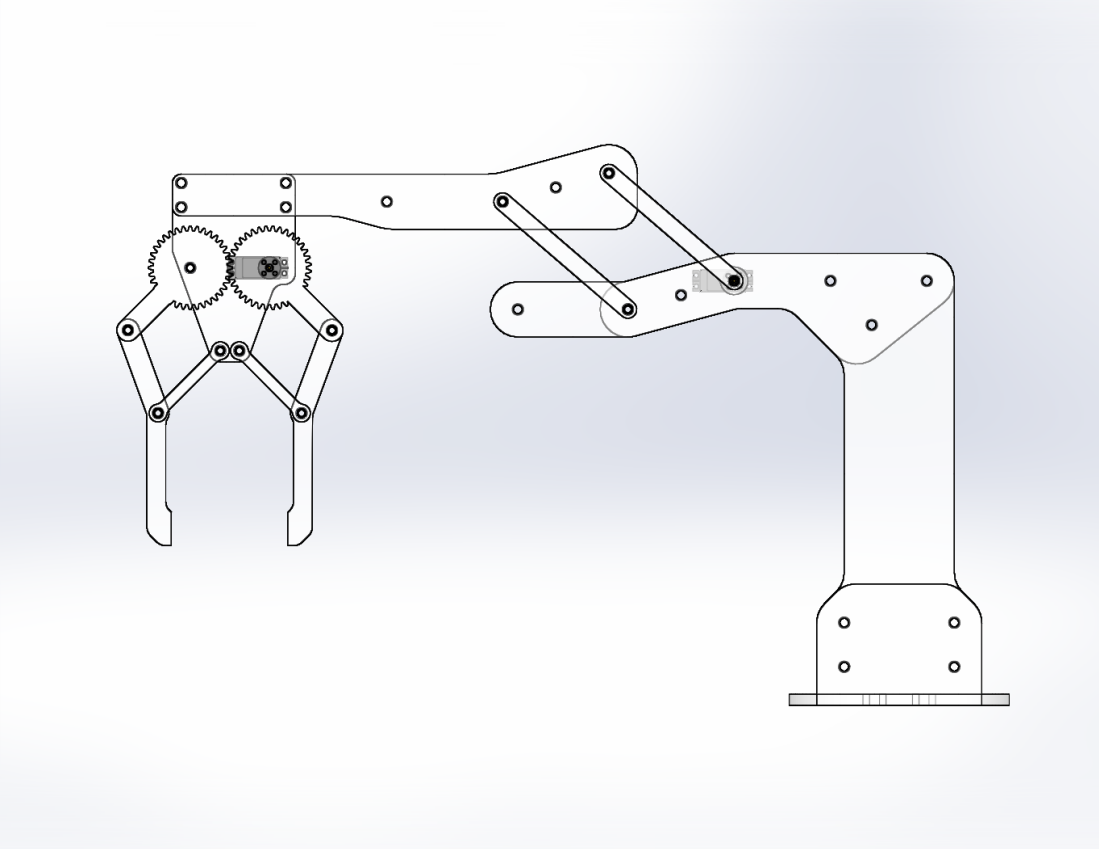
Proyek ini memiliki lisensi bervariasi mengingat seluruh resources yang digunakan merupakan turunan dari banyak proyek opensource lain.Namun mengingat kemiripan dan kompatibilitas antar jenis lisensi (terutama jenis MIT, Apache, GPL dan LPGL) maka pengembang menentukan lisensi dari seluruh proyek ini adalah GPL v2.Lebih lanjut tentang lisensi dapat dilihat di alamat: <http://opensource.org/licenses/>

**Chapter 2. Mekanik**

**2.1 Gripper**



**2.2 Arm**

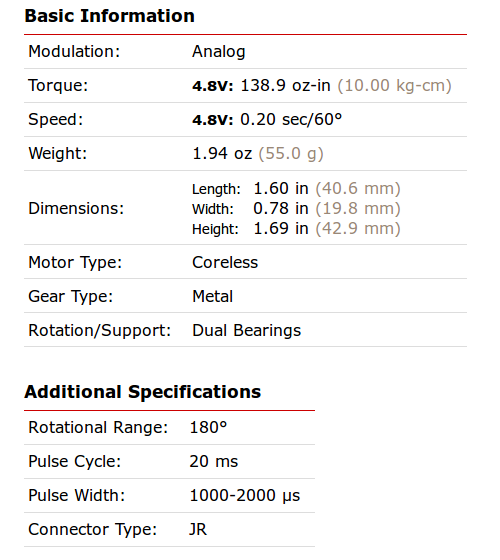
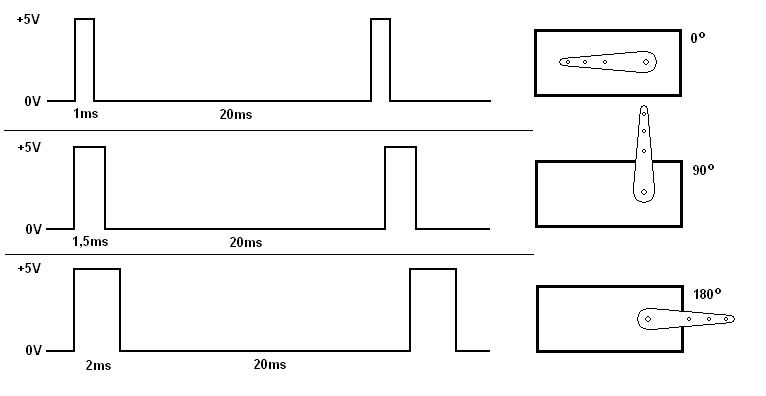


**2.3 Meja**

**Chapter 3. Elektronik**

**3.1 Servo**

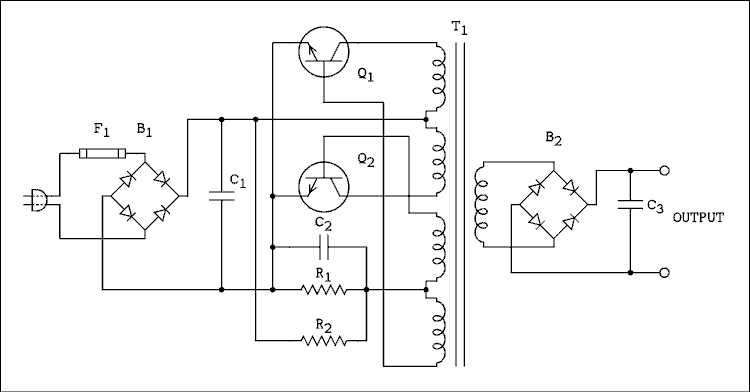
Servo adalah aktuator rotari yang memungkinkan untuk kontrol yang tepat dari posisi sudut, kecepatan dan percepatan. Terdiri dari motor DC dengan torsi besar plus gearbox yang digabungkan dengan sensor untuk umpan balik posisi (biasanya berupa potensiometer). Hal ini juga membutuhkan kontroler yang relatif canggih, seringkali modul khusus yang dirancang khusus untuk digunakan dalam mengontrol servo. Kabel menuju servo berjumlah 3 yaitu power positit, ground, dan sinyal. Sinyal disini berupa sinyal kotak dengan frekuensi dan duty-cycle tertentu untuk mengendalikan posisi putaran servo. Berikut adalah contoh sinyal kendali servo.



**3.2 SPS**

SPS adalah power supply dengan regulasi switching atau lebih dikenal sebagai power supply switching. Kelebihan power supply switching adalah efisiensi daya yang besar sampai sekitar 83% jika dibandingkan dengan power supply dengan regulasi biasa yang menggunakan LM78xx. Efisiensi yang rendah pada regulator LM78xx dikarenakan kelebihan tegangan input regulator akan dirubah menjadi panas sehingga sebagian besar daya input akan hilang karena dirubah menjadi panas tersebut. Bagaimanapun juga semua regulator harus mendapatkan tegangan input yang lebih tinggi daripada tegangan regulasi output untu mendapatkan tegangan yang teregulasi.

Tegangan regulasi dihasilkan dengan cara men-switching transistor seri ‘on’ atau ‘off ’. Dengan demikian duty cycle-nya menentukan tegangan DC rata-rata. Duty cycle dapat diatur melalui feedback negatif. Feedback ini dihasilkan dari suatu komparator tegangan yang membandingkan tegangan DC rata-rata dengan tegangan referensi.



**3.3 Wiring**

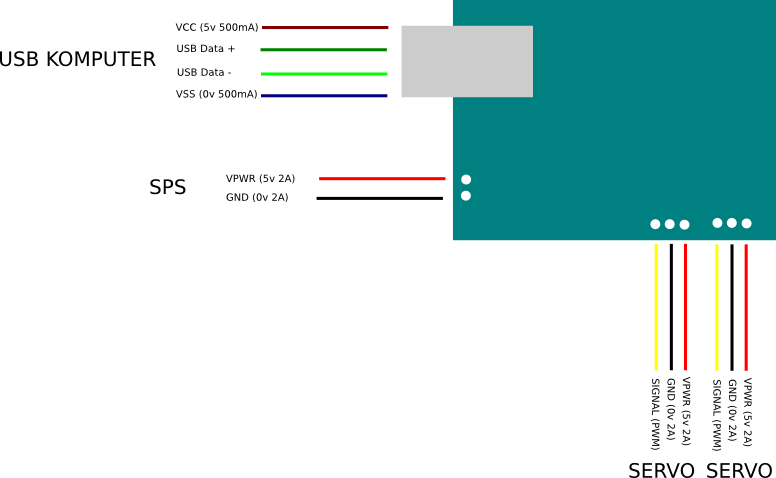
Secara global, sistem tangan robot yang digunakan memiliki bagian utama yaitu:

* Servo sebanyak 2 unit
* Controller sebanyak 1 unit
* SPS sebanyak 1 unit
* Komputer sebanyak 1 unit.

Untuk koneksi antara Controller dengan bagian-bagian lainnya dijelaskan seperti berikut:

* Wiring antara Controller dan masing-masing servo memiliki 3 jalur yaitu jalur positif 6 volt, ground, dan sinyal.
* Wiring antara Controller dan SPS memiliki 2 jalur yaitu jalur positif 6 volt dan ground.
* Wiring antara Controller dan Komputer memiliki 4 jalur yaitu jalur positif 5 volt, ground, USB data postif, dan USB data negatif.

Perlu diketahui bahwa ground SPS dan Servo terhubung, sedangkan dengan Komputer terpisah. Perlu diketahui pula bahwa arus di jalur Komputer hanya 500mA, sedangkan di jalur SPS dan servo mencapai 2 Ampere.Kamera dan Komputer terhubung secara langsung melalui jalur USB.Untuk memudahkan maka berikut skema dari penjelasan diatas:



**3.4 PCB**

PCB atau printed circuit board adalah papan yang telah dicetak jalur tembaga sebagai tempat komponen elektronik disolder dan jalur elektrisnya.Untuk dapat membuat PCB Controller yang digunakan dalam proyek ini maka diperlukan berikut ini:

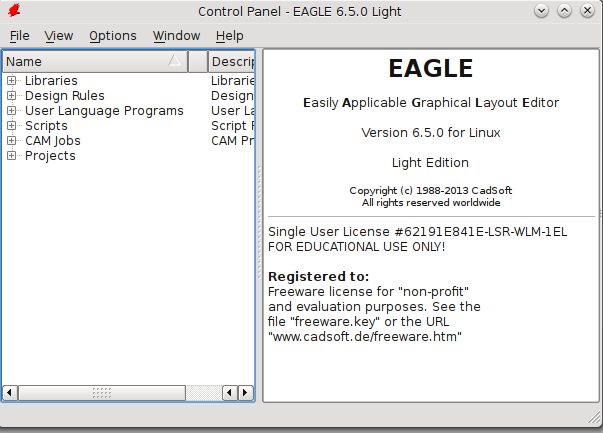
* Software CAD (Computer Aided Design)
* PCB Manufaktur.

Untuk PCB Manufaktur disini dipercayakan kepada CV. Maxtron Persada Indonesia.Perusahan ini merupakan perusahaan nasional yang bergerak di bidang manufaktur PCB kualitas tinggi dan penyedia mesin CNC untuk produksi PCB.Alasan mengapa dipilih perusahaan ini selain karena hasil manufakturnya yang berkualitas juga letaknya yang masih area Surabaya-Sidoarjo.Untuk informasi lebih lanjut dapat kunjungi situs resmi <http://www.maxtronpersada.com/>

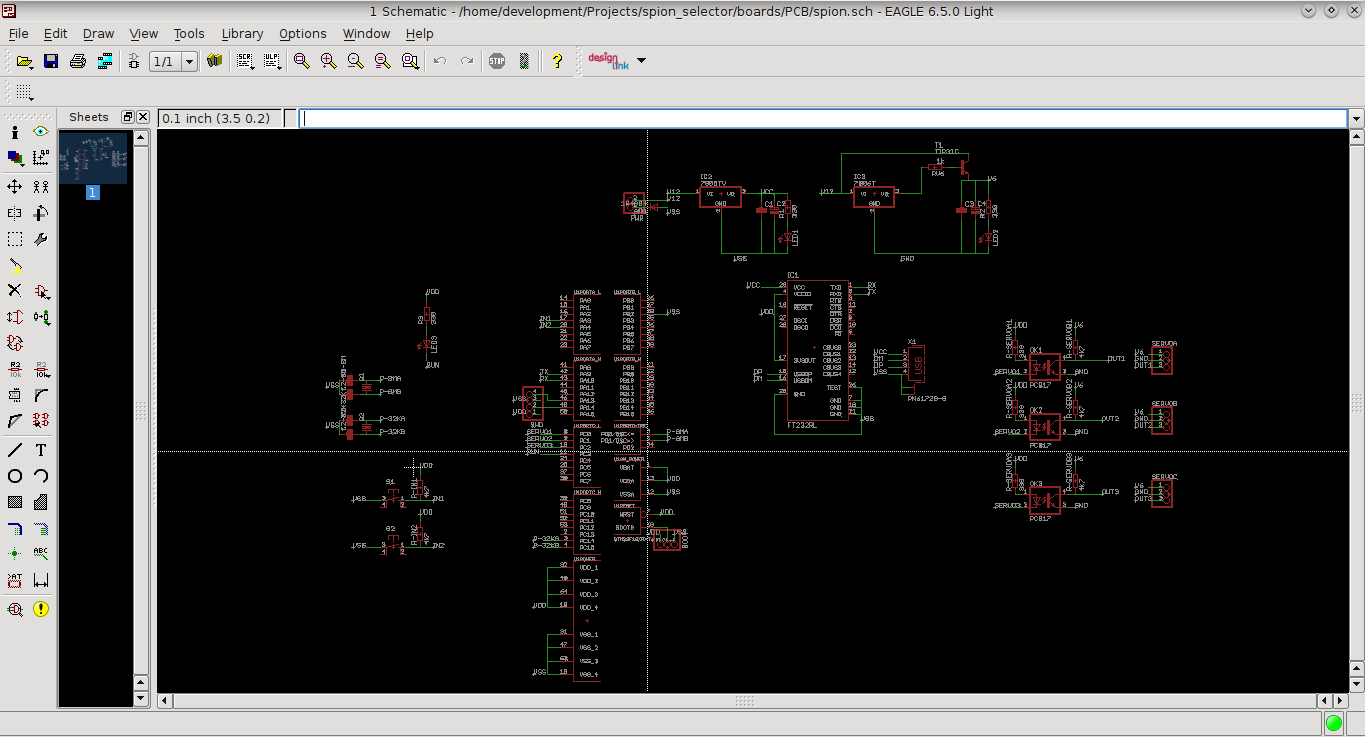
Selanjutnya adalah Software Computer Aided Design. Software CAD digunakan untuk mendesain baik skematik maupun layout sebelum di kirim ke perusahaan manufaktur PCB.Software CAD untuk PCB yang digunakan disini adalah EAGLE. Untuk menginstal EAGLE di LinuxMint dari internet dengan mengetikkan di terminal:

**sudo apt-get install eagle**

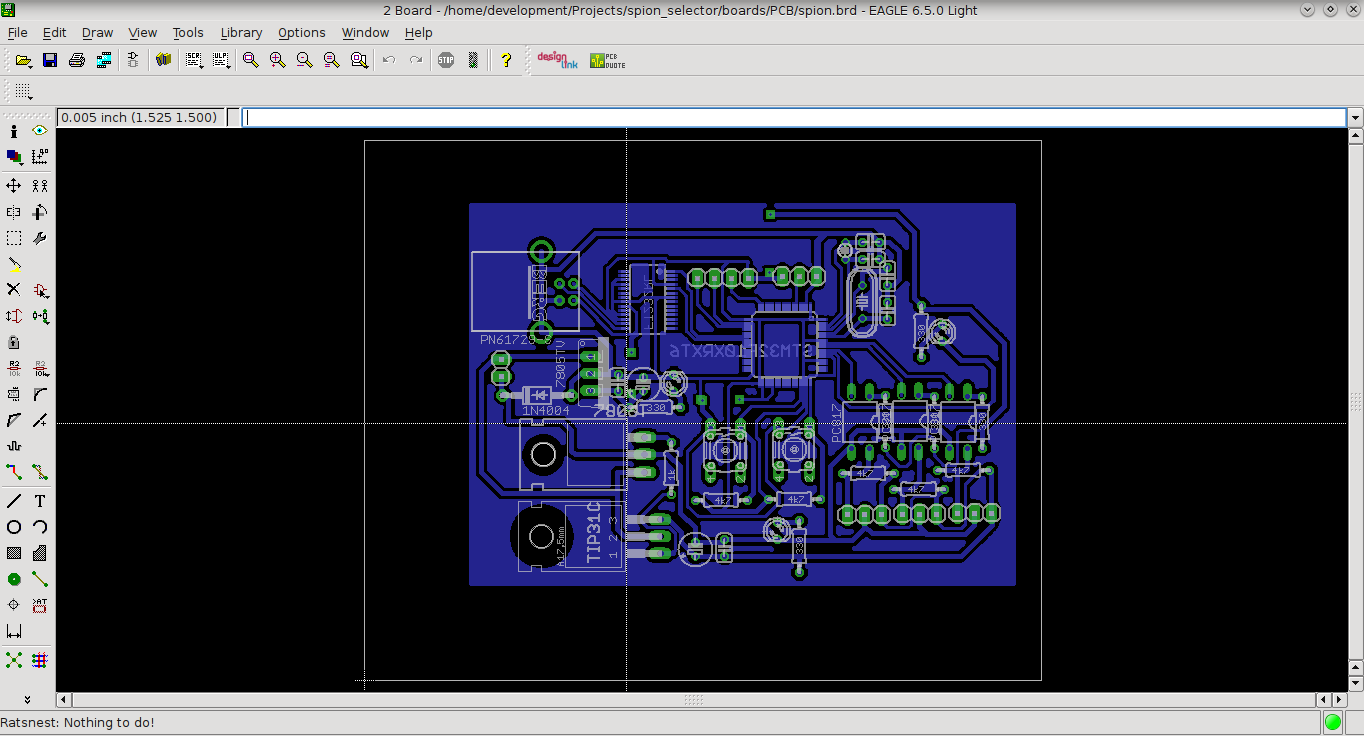
Berikut adalah tampilan software EAGLE:



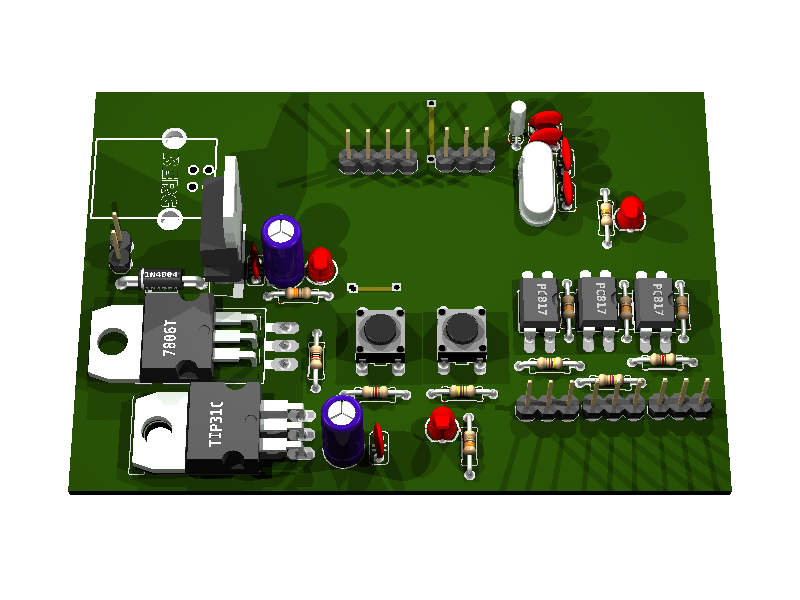
Berikut adalah gambar skema PCB Controller yang dibuka dengan EAGLE schematic:



Berikut adalah gambar skema PCB Controller yang dibuka dengan EAGLE boards:



Berikut ilustrasinya secara 3D di posisi TOP dan BOTTOM:



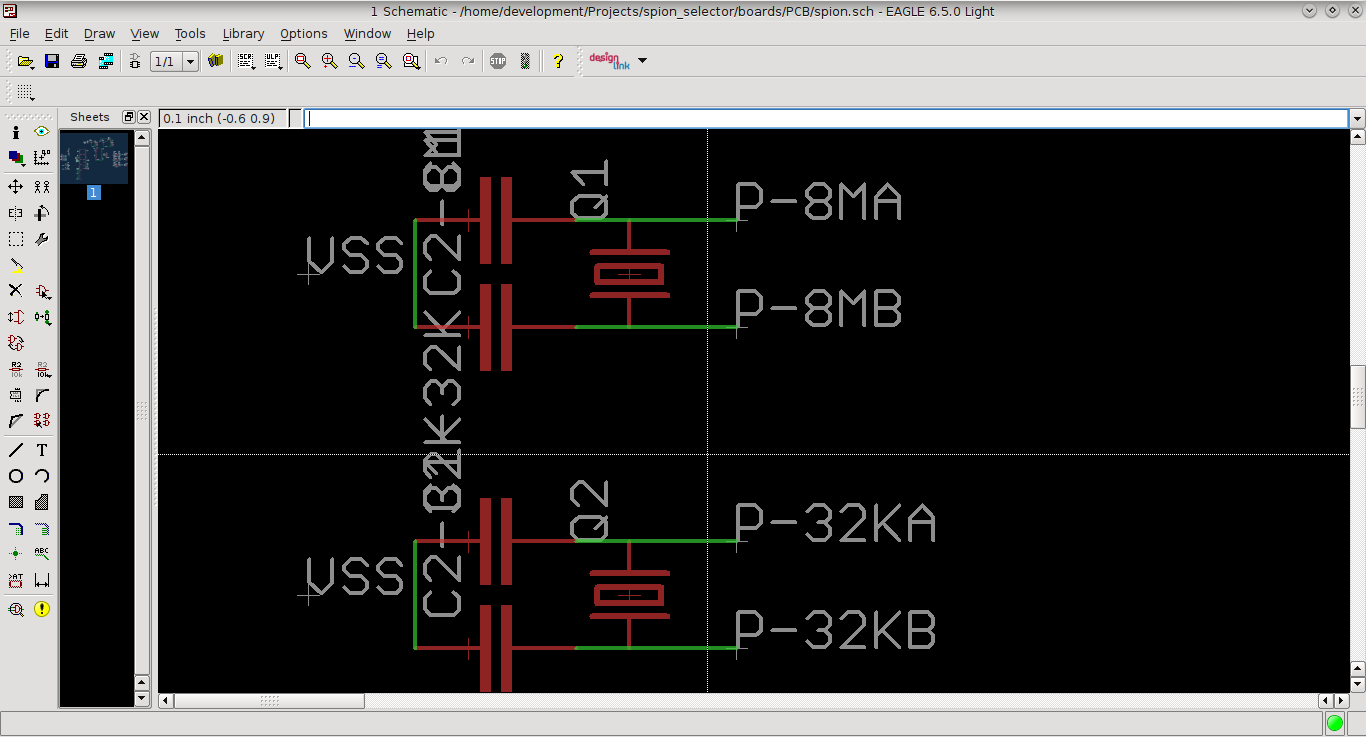
Selanjutnya adalah beberapa bagian penting dari skema PCB Controller meliputi:

1. Pembangkit Clock

Pembangkit clock ini digunakan untuk membangkitkan pulsa elektrik yang akan digunakan sebagai acuan chip kontroller bekerja.Terbagi dalam dua jenis frekuensi:

* 32khz digunakan ketika chip utama di program
* 8mhz digunakan ketika chip utama berproses

Berikut adalah potongan gambar skema PCB Controller yang menunjukkan bagian Pembangkit Clock

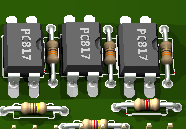
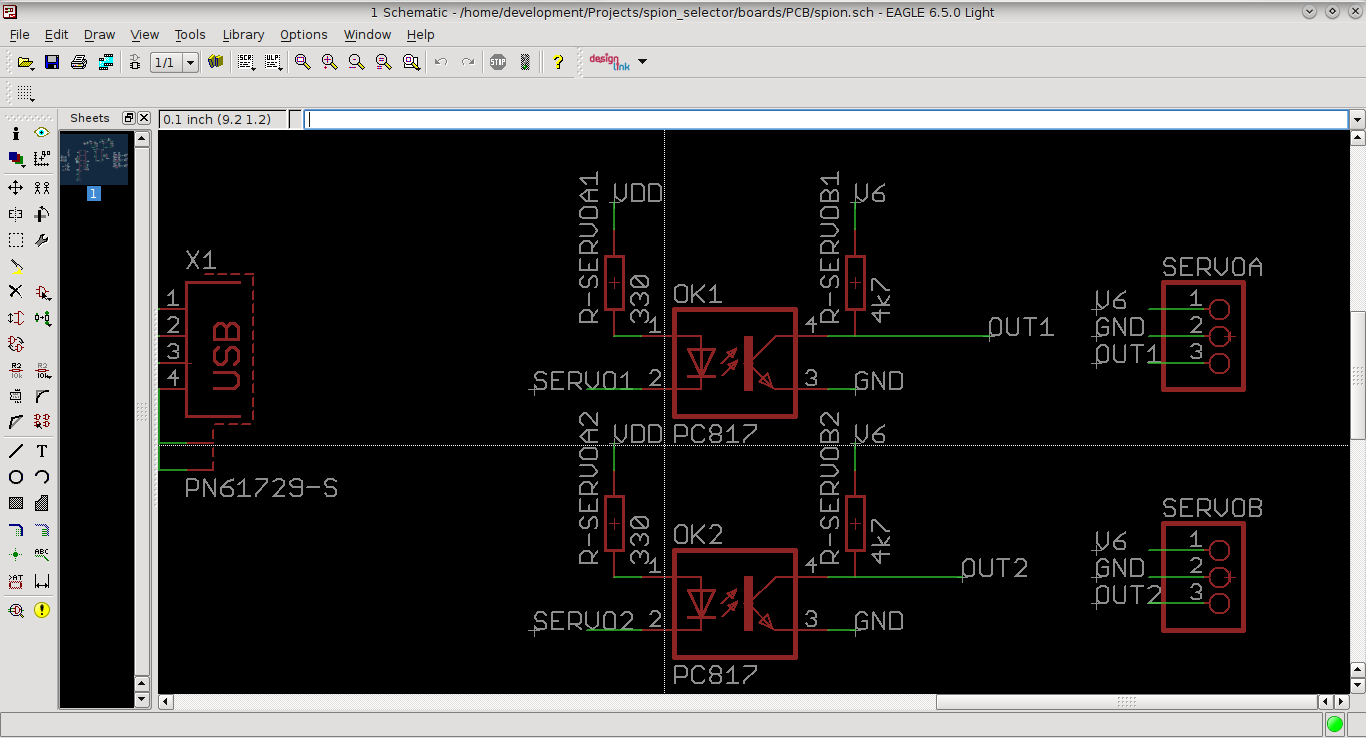


1. Servo Signal Brigde

Dalam mengoperasikan aktuator servo, diperlukan komponen sebagai jembatan sinyal antara chip utama dan servo.Beberapa alasan dibutuhkan jembatan sinyal adalah:

* Level tegangan chip utama adalah 3,3v sedangkan servo adalah 6 volt.
* Chip utama bekerja pada arus dibawah 500mA sedangkan servo diatas 1A.
* Servo menghasilkan induksi balik sehingga chip utama dan servo perlu memiliki acuan tegangan 0 terpisah.

Untuk memenuhi kebutuhan inilah digunakan IC PC817 sebagai jembatan antara chip utama dan servo.Komponen PC817 terdiri dari LED (untuk sisi chip) dan PhotoTransistor (untuk sisi servo).Mekanisme kerja PC817 adalah switch PhotoTransistor di kendalikan oleh sinyal cahaya dari LED.Oleh karena sinyal antara sisi LED dan PhotoTransistor bukan sinyal elektris (melainkan cahaya), maka LED dan PhotoTransistor dapat bekerja di level tegangan, arus dan acuan tegangan yang terpisah.Berikut adalah potongan gambar skema PCB Controller yang menunjukkan bagian Servo Signal Brigde:



1. Power Regulator merupakan bagian opsional yang berfungsi untuk mengatur tegangan tenaga yang masuk ke circuit agar sesuai dengan kebutuhan.Disini digunakan dua jenis IC regulator:

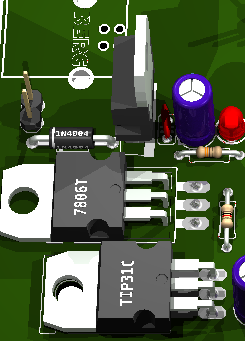
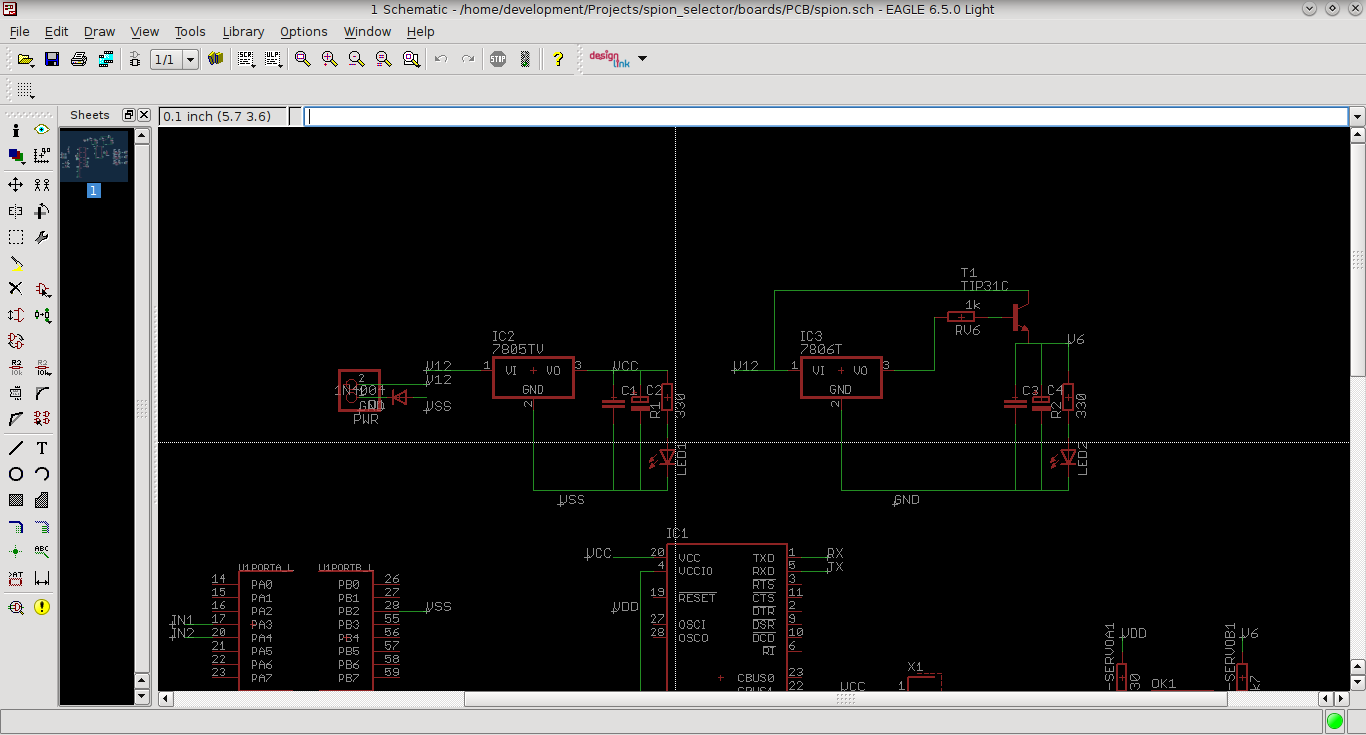
* 7806 untuk tenaga ke servo sebesar 6 volt.
* 7805 untuk rangkaian ke chip sebesar 5 volt.

Pada setiap segmen telah ditambahkan LED untuk indikator dan Electrolite Capacitor untuk menstabilkan tegangan. Arus maximal yang dikeluar di setiap IC adalah 1A.

Khusus untuk servo, digunakan pula Transistor TIP 31C untuk menambahkan arus ke servo dengan sumber arus asli.Namun dalam proyek ini, segmen ini tidak digunakan karena digantikan oleh:

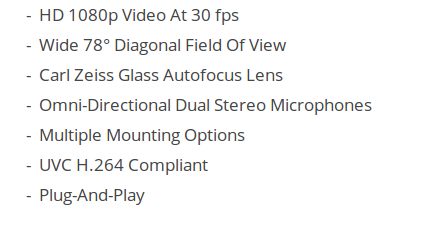
* Untuk servo tenaga diperoleh dari SPS sebesar 5 volt 2 Ampere.
* Untuk chip tenaga diperoleh dari USB Komputer sebesar 5 volt 500mA.

Berikut adalah potongan gambar skema PCB Controller yang menunjukkan bagian Power Regulator:



**3.5 Kamera**

Kamera yang digunakan adalah Logitech C920. Kamera ini digunakan untuk mengambil gambar dan selanjutnya diproses. Spesifikasi kamera ini adalah:

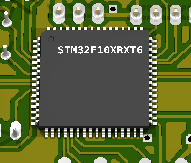
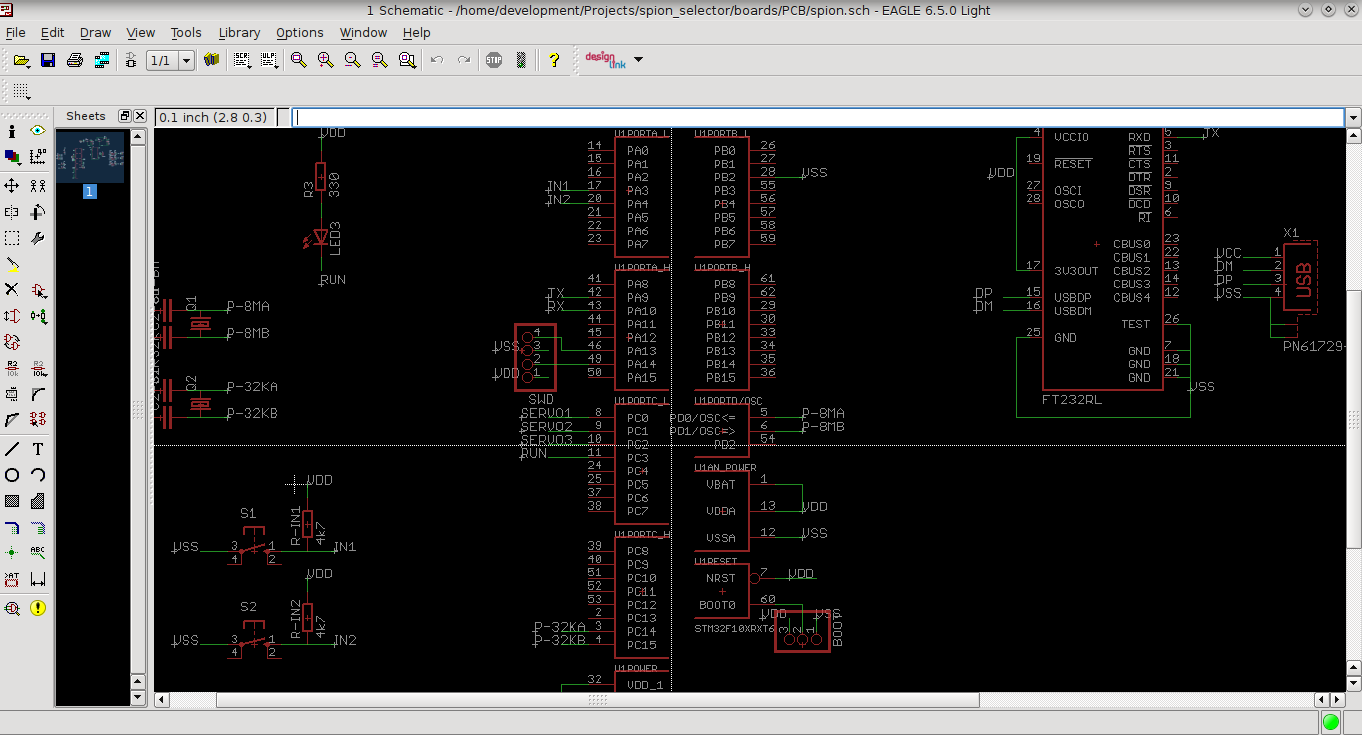
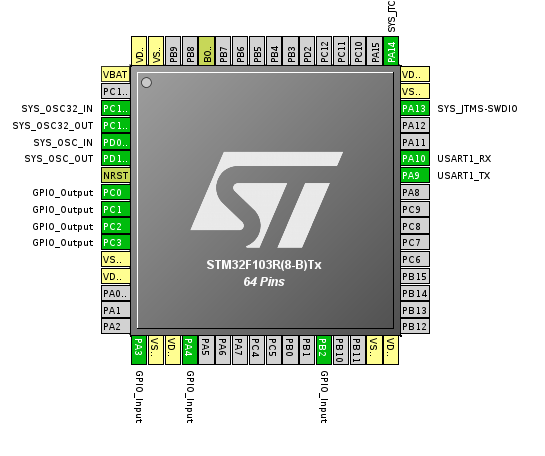


**Chapter 4. Controller**

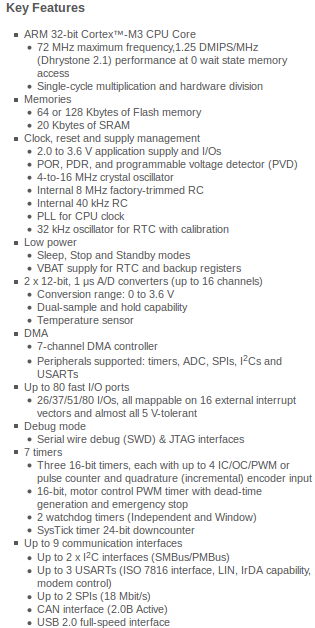
**4.1 STM32**

STM32 adalah microcontroller 32bit yang kini telah digunakan di hampir semua instrumen.STM32 adalah salah satu produk dari ST Microelectronic Inc, sebuah perusahaan semikonduktor asal Italia.STM32 memiliki fitur lengkap dengan CPU 32bit plus FPU danregister-register baik 32bit, 16bit, 12bit, maupun 8bit. Berikut beberapa data STM32 secara umum:

Untuk jenis STM32 yang digunakan dalam proyek ini adalah STM32F103RBT6.Berikut adalah gambar mapping pin yang digunakan:

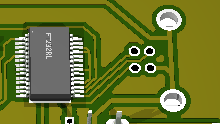
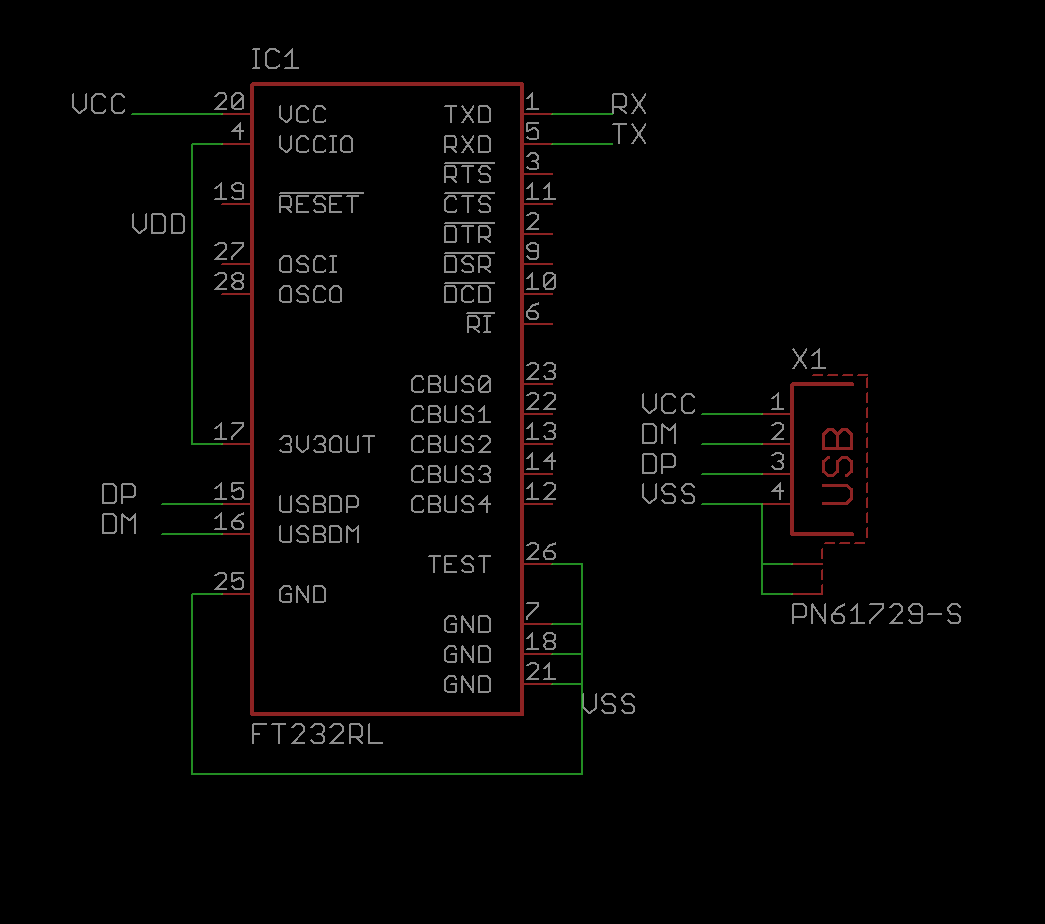


Berikut spesifikasi chip STM32F103RB:



**4.2 FT232RL**

FT232RL adalah chip konversi signal USB ke signal TTL/UART (*USB-to-TTL Converter*) yang andal dan praktis untuk digunakan pada rangkaian elektronika berbasis mikrokontroler, dengan demikian perangkat elektronika buatan Anda bisa berkomunikasi dengan perangkat lain lewat komunikasi standar USB. Tegangan konsumsi untuk FTDI adalah standar 5v dan menyediakan keluaran 3.3v sedangkan untuk tegangan IO dapat menggunakan 5 atau 3.3v. Berikut adalah skema penggunakan FT323RL:



Berikut fitur utama chip FT232RL:

· Single chip USB to asynchronous serial data transfer interface.

· Entire USB protocol handled on the chip - No USB-specific firmware programming required.

· UART interface support for 7 or 8 data bits, 1 or 2 stop bits and odd / even / mark / space / no parity.

· Fully assisted hardware or X-On / X-Off software handshaking.

· Data transfer rates from 300 baud to 3 Megabaud (RS422 / RS485 and at TTL levels) and 300 baud to 1 Megabaud (RS232).

· In-built support for event characters and line break condition.

· New 48MHz, 24MHz,12MHz, and 6MHz clock output signal Options for driving external MCU or FPGA.

· FIFO receive and transmit buffers for high data throughput.

· 256 Byte receive buffer and 128 Byte transmit buffer utilising buffer smoothing technology to allow for high data throughput.

· Integrated 1024 bit internal EEPROM for I/O configuration and storing USB VID, PID, serial number and product description strings.

· Integrated 3.3V level converter for USB I/O .

· True 5V / 3.3V / 2.8V / 1.8V CMOS drive output and TTL input.

· Fully integrated clock - no external crystal, oscillator, or resonator required.

· 3.3V to 5.25V Single Supply Operation.

· USB 2.0 Full Speed compatible.

· -40°C to 85°C extended operating temperature range.

**Chapter 5. Real-Time Operating System (RTOS)**

**5.1 HAL**

Pada bagian ini akan dijelaskan tentang HAL (Hardware Abstraction Layer) yang digunakan dalam proyek ini.Karena penjelasan ini adalah bagian dari RTOS (Real Time Operating System) yang akan ditanamkan di chip controller, maka diperlukan software Qt Creator untuk membuka file-file kode sumber yang ada di folder project.Untuk menginstal Qt Creator di LinuxMint dari internet dengan mengetikkan di terminal:

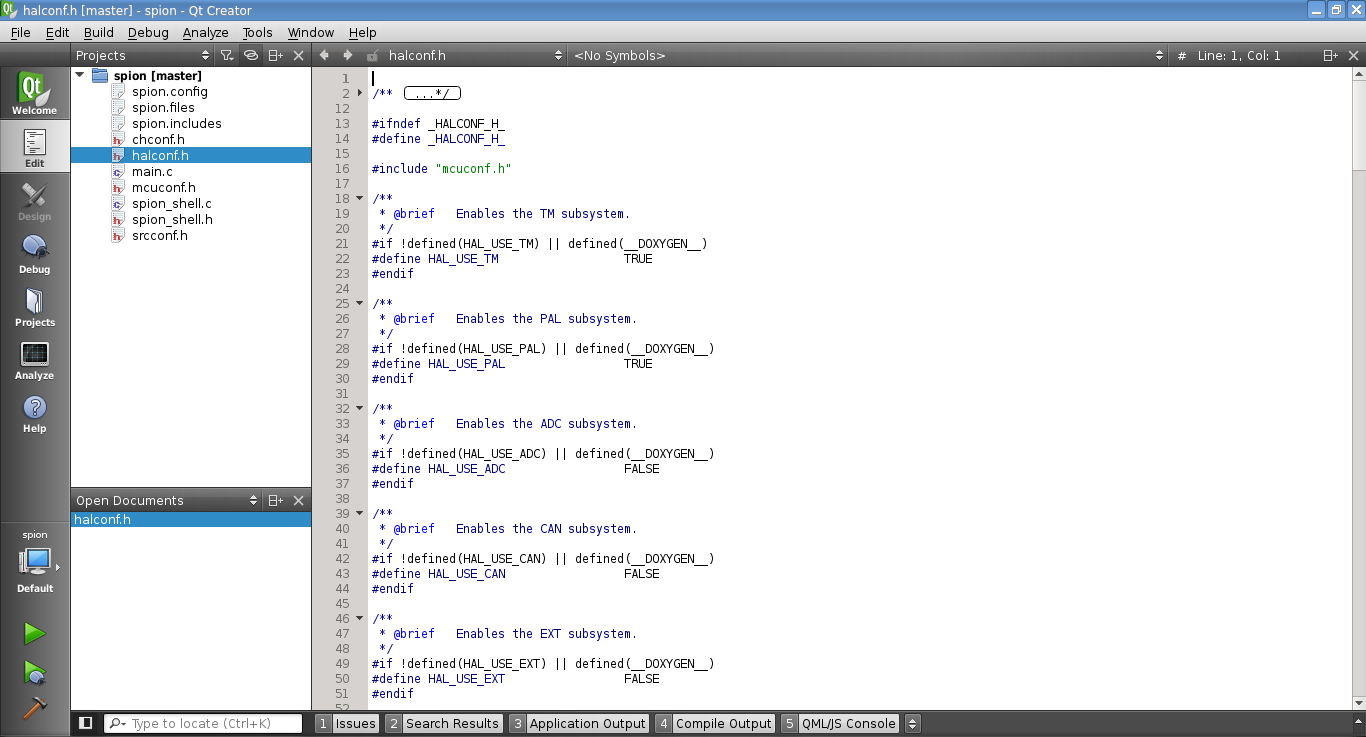
**sudo apt-get install qtcreator qtdeclarative5-qtquick2-plugin**

RTOS adalah kependekan dari Real Tipe Operating System. RTOS disini yang digunakan ChibiOS yang merupakan sekumpulan template kode sumber dalam bahasa C dan Makefile untuk membuat firmware multitask pada sistem ARM.Lebih lanjut tentang ChibiOS dapat dilihat di alamat: [www.chibios.org](http://www.chibios.org)

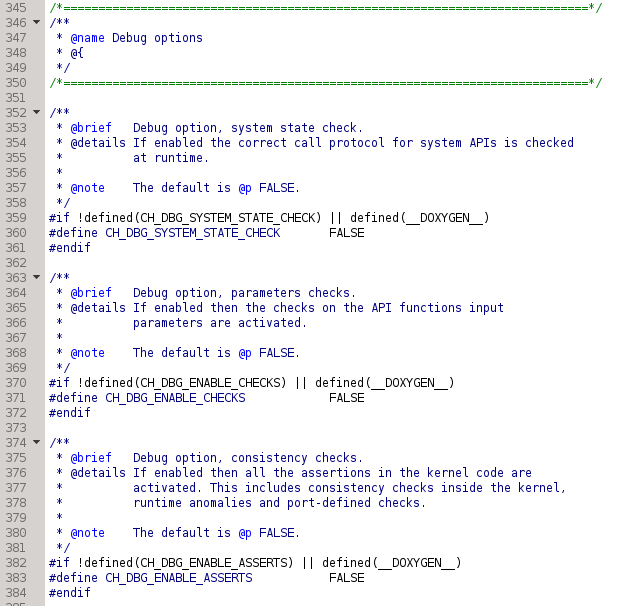
Sementara HAL itu sendiri adalah kependekan dari Hardware Abstraction Layer.HAL merupakan fitur pada RTOS yang memungkinkan pengembang tidak harus menggunakan register-register low level dalam membuat firmwarenya. Register low level sifatnya sangat bergantung kepada tipe dan merek chip sekalipun sama-sama memiliki arsitektur ARM.Untuk itulah dibuat fitur HAL sehingga pada level tertinggi dari hirarki fungsi, pengembang dapat menggunakan fungsi-fungsi yang sama untuk beragam tipe dan mereka chip selama telah RTOS memiliki HAL-nya. Karena berbasis ARM maka kompilasinya dapat dilakukan oleh software arm-gcc dan dapat diinstal dengan perintah:

**sudo apt-get install gcc-arm-none-eabi libnewlib-arm-none-eabi libnewlib-dev**

Selanjutnya adalah membuka file bertipe \*.creator.Maka akan muncul tampilan seperti berikut:



Pembahasan hanya difokuskan pada kode dasar (codebase) yang ditulis oleh pengembang.Sedangkan untuk kode sisanya dapat menggunakan template untuk seri STM32F103 yang telah disiap oleh pengembang ChibiOS/RT.Sebelum mengatur HAL pengembang menyarankan agar membuka dulu file chconf.c yang berisi pengaturan terhadap kernel ChibiOS/RT. Disarankan untuk mematikan seluruh opsi Debugging untuk mengurangi konsumsi memory flash dan SRAM kecuali jika memang dibutuhkan.



**5.2 Threads**

Threads adalah algoritma sub-RTOS yang mampu bekerja secara independen dengen memesan memory tersendiri (eksklusif) dan mampu saling berkomunikasi melalui variabel atau flag yang berskup global. Thread yang digunakan terbagi menjadi:

5.2.1 Thread Main

Thread ini adalah thread utama dan thread yang pertama untuk di ekskusi. Thread ini berfungsi start-up dan memanggil thread lain kemudian di akhiri dengan loop untuk kedip LED sebagai inikator sistem yang masih berjalan. Thread ini berada di file main.c pada fungsi main() dan berikut adalah kode sumbernya:

int main(void) {

halInit();

chSysInit();

palSetPadMode(GPIOC,3,PAL\_MODE\_OUTPUT\_PUSHPULL);

palSetPad(GPIOC,3);

vservo0=1500;

vservo1=1500;

vservo2=1500;

palSetPadMode(GPIOC,0,PAL\_MODE\_OUTPUT\_PUSHPULL);

palSetPadMode(GPIOC,1,PAL\_MODE\_OUTPUT\_PUSHPULL);

palSetPadMode(GPIOC,2,PAL\_MODE\_OUTPUT\_PUSHPULL);

chThdCreateStatic(wa\_servo0Thread, sizeof(wa\_servo0Thread), NORMALPRIO, servo0Thread, NULL);

chThdCreateStatic(wa\_servo1Thread, sizeof(wa\_servo1Thread), NORMALPRIO, servo1Thread, NULL);

chThdCreateStatic(wa\_servo2Thread, sizeof(wa\_servo2Thread), NORMALPRIO, servo2Thread, NULL);

Shell\_Setup();

while (TRUE) {

Shell\_Create();

palTogglePad(GPIOC,3);

chThdSleepMilliseconds(100);

}

}

5.2.2 Thread Servo

Thread ini berfungsi untuk menghasilkan sinyal untuk kendali servo. Thread ini menghasilkan sinyal dengan mengatur pin output dengan high dan low secara bergantian dengan waktu tunda tertentu. Thread ini berada di main.c dan terbagi menjadi 3 thread (masing-masing untuk 3 servo). Berikut adalah kode sumber salah satu thread:

#define loopmax 2000

uint16\_t vservo0,vservo1,vservo2;

static WORKING\_AREA(wa\_servo0Thread, 128);

static msg\_t servo0Thread(void \*arg) {

(void)arg;

while (TRUE) {

servo0\_on;

chThdSleepMicroseconds(vservo0);

servo0\_off;

chThdSleepMicroseconds(loopmax-vservo0);

}

return 0;

}

5.2.3 Thread Serial

Thread ini berfungsi untuk berkomunikasi dengan komputer melalui jalur serial UART. Thread ini menangani data string standar yang di definisikan di stdio.h. Di thread ini juga respon untuk servo di definisikan. Thread ini berada di file spion\_shell.c dan untuk perintah-perintahnya adakan dijelaskan pada sub-bab selanjutnya.Berikut adalah kode sumbernya:

static const ShellCommand commands[] = {

{"test",cmd\_ok},

{"set0",cmd\_set0},

{"set1",cmd\_set1},

{"set2",cmd\_set2},

{"val",cmd\_value},

{"grip",cmd\_grip},

{"ungrip",cmd\_ungrip},

{"take",cmd\_take},

{"untake",cmd\_untake},

{"rest",cmd\_rest},

{NULL, NULL}

};

static const ShellConfig shell\_cfg1 = {

(BaseSequentialStream \*)&SD1,

commands

};

**5.3 Commands**

Berikut akan dijelaskan perintah-perintah yang digunakan untuk komunikasi antara PC dan kontroller.Semua perintah ini berada di file spion\_shell.c. Perintah ini terbagi menjadi 4 yaitu:

5.3.1 Grip

Perintah untuk penjepit untuk mulai memegang (grip) spion. Berikut kode sumber

extern uint16\_t vservo0,vservo1,vservo2;

static void cmd\_grip(BaseSequentialStream \*chp, int argc, char \*argv[]) {

(void)argv;

if(argc!=0){

chprintf(chp,"bad commands\r\n");

return;

};

vservo0=1500;

return;

}

5.3.2 Ungrip

Perintah untuk penjepit untuk mulai melepas (ungrip) spion. Berikut kode sumber

extern uint16\_t vservo0,vservo1,vservo2;

static void cmd\_ungrip(BaseSequentialStream \*chp, int argc, char \*argv[]) {

(void)argv;

if(argc!=0){

chprintf(chp,"bad commands\r\n");

return;

};

vservo0=500;

return;

}

5.3.3 Take

Perintah untuk lengan untuk mulai mengambil spion. Berikut kode sumber

extern uint16\_t vservo0,vservo1,vservo2;

static void cmd\_take(BaseSequentialStream \*chp, int argc, char \*argv[]) {

(void)argv;

if(argc!=0){

chprintf(chp,"bad commands\r\n");

return;

};

Vservo1=1500;

return;

}

5.3.4 Untake

Perintah untuk lengan untuk mulai mengembalikan spion. Berikut kode sumber

extern uint16\_t vservo0,vservo1,vservo2;

static void cmd\_untake(BaseSequentialStream \*chp, int argc, char \*argv[]) {

(void)argv;

if(argc!=0){

chprintf(chp,"bad commands\r\n");

return;

};

Vservo1=500;

return;

}

**Chapter 6. Octave**

**6.1 Instalasi**

**6.2 Image Processing**

**6.3 Mekanisme**