**247-GEE-LG / 247-GED-LG**

**Réalisation et conception de projet**

**Rapport technique**

Projet supervisé par:

[Raoudha Jaouadi](https://clg.moodle.decclic.qc.ca/mod/resource/view.php?id=143839) et Gilles Deschenes

Projet effectué par:

Guillaume Bergeron et Julie Foucault

Date:

22 Mai 2018

Sommaire

[Résumé du projet 3](#_Toc514775502)

[Objectifs du projet 3](#_Toc514775503)

[Problèmes rencontrés 5](#_Toc514775504)

[Améliorations possibles 7](#_Toc514775505)

[Conclusion 8](#_Toc514775506)

[Annexe 9](#_Toc514775507)

[Code source 9](#_Toc514775508)

[Schéma(s) électrique(s) 9](#_Toc514775509)

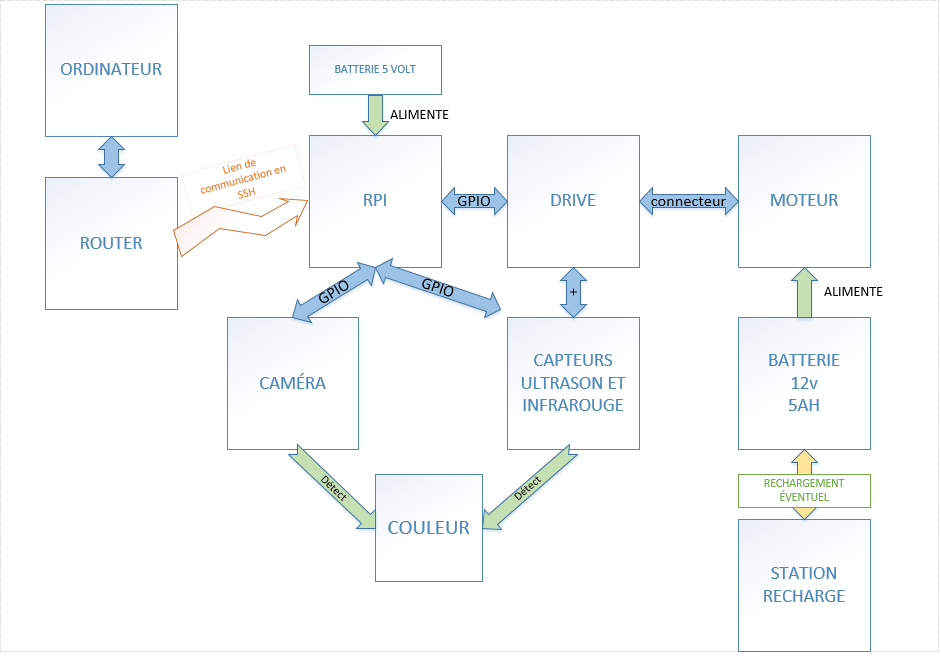
[Schéma mécanique 17](#_Toc514775510)

[Référence : 19](#_Toc514775511)

# Résumé du projet

Le Robot Suiveur est un petit robot sur deux roues qui nous suit, peu importe où nous allons à l’aide d’un tague de couleur. Il peut aussi éviter les obstacles à l’aide de capteurs de distance et aller se recharger par lui-même en trouvant sa station de charge. Les modes du robot peuvent être commandés par un site internet à l’aide d’une communication par SSH.

Voici un schéma bloc du projet:



# Objectifs du projet

Voici les objectifs que nous devions atteindre :

Pour le Robot :

1. Trouver les composantes essentielles pour sa construction : de bonnes batteries 12v 4000mah minimum, un contrôleur pour tout gérer, une caméra pour capturer l'image, les capteurs de distance pour éviter les obstacles et connaître la distance entre le robot et l'objet, les composantes électroniques pour la carte de contrôle de puissance et les lectures analogiques, puis la structure du robot avec les deux roues
2. Modifier la partie mécanique d’un ancien projet pour l’adapter à notre projet.
3. Construire une carte de contrôle de puissance pour les moteurs du robot qui supporte 5A par moteur «peak» et qui est capable d’accepteur 3.3V comme niveau haut dans ces entrées.
4. Faire fonctionner électroniquement:
5. Les moteurs
6. La carte de contrôle de puissance   
   *Atteint, mais nous avons eu un problème électrique qui n’a pas été réglé. Ceci n’a toutefois pas empêché le contrôle de nos moteurs durant nos tests de programmation et durant la présentation finale.*
7. La caméra
8. Les capteurs de distance : ultrasons et infrarouge
9. Le convertisseur analogique/digital
10. Les plaques de recharge
11. Programmer :
12. Les moteurs
13. La caméra pour détecter un objet à l’aide d’une forme ou couleur.

*Atteint, mais au lieu d’une forme c’est une couleur.*

1. Les capteurs
2. Le convertisseur analogique/digital
3. Une interface Web
4. Intégrer les modes de fonctionnement du robot : Suiveur, recharge, stop.
5. Assembler le site internet pour commander le robot
6. La station de recharge. *Complété à 80% dû aux problèmes électriques. Le robot est capable 1 fois sur 3 de bien se stationner.*

Pour la Station de charge :

1. Construire une carte de recharge de batterie acide
2. Construire la structure de la station de charge
3. Tester le fonctionnement à l’aide d’une batterie
4. Tester le fonctionnement sur le robot.

*Pas atteint, car nous n’avions pas réservé de moment dans notre planification pour tester le circuit avec le robot et lorsqu’un problème s’est présenté, on a été beaucoup retardé. Donc, à cause d’un manque de temps et de test, nous avions peur d’endommager le circuit du contrôle des moteurs lorsque le robot se connecterait au chargeur. De plus, étant donné qu’il nous restait deux semaines de préparation avant la présentation, on n’avait plus le droit à l’erreur.*

\*\*\*Tous les objectifs ont été atteints sauf ceux qui sont spécifiés le contraire.

# Problèmes rencontrés

Voici les problèmes rencontrés énumérés par section :

Programmation :

Installer les librairies pour faire la cross-compilation était l'un des plus gros défis au niveau de la programmation. Après 2 semaines et beaucoup d'essaies erreurs, on a finalement trouvé un bon site internet qui montrait comment installer les librairies de Opencv sur l'architecture Linux. Cependant, on n'avait pas fini de rencontrer des problèmes, il fallait également installer les librairies pour avoir accès au gpio, la caméra et mqtt.

Le fait de ne pas être capable de retirer une hauteur lors de l'analyse de l'image nous a compliqué la tâche et l'on a dû ajouter un capteur infrarouge à l'avant du robot, car on était incapable de détecter des pieds.

Le capteur infrarouge ne détectait pas les objets noirs et il avait de la difficulté à nous détecter lors de la présentation, car on portait des souliers noirs. Si l'on avait constaté ce problème plus tôt, on aurait porté des souliers d’une couleur différente.

On a eu quelques problèmes de performance avec la tâche Hsv. Le bût était de faire fonctionner la tâche le plus rapidement possible afin d’avoir un temps de réaction moins élevé et de suivre l’objet le plus fluidement possible. Malheureusement, on n’arrivait pas à avoir un taux de rafraichissement assez rapide. On a trouvé plusieurs solutions pour améliorer les performances :

1. Ouvrir le Raspberry Pi sans interface graphique « desktop ».
2. Réserver un cœur du processeur uniquement pour la tâche Hsv.
3. Réduire la résolution de la caméra à 640px par 480px.

Électronique :

On a eu quelques problèmes de routage causé par des erreurs d'inattentions lors de la consultation du schéma électrique de la «drive». La solution était de mettre des fils de cuivre pour corriger les erreurs. On a aussi mis les sorties d’un amplificateur opérationnel sur le GND qui n’était pas utilisé. Ceci a augmenté sa température et l'a brûlé. Pour corriger le problème, il a fallu qu’on coupe les pattes qui faisaient de mauvaises connexions pour le bon fonctionnement du circuit intégré.

De plus, on a accidentellement soudé une diode du mauvais sens sur notre PCB de station de recharge. La solution était de mettre une nouvelle diode sur le bon sens et de remplacer le transistor qui a été sauté par cette erreur.

On a forcé manuellement les moteurs en voulant vérifier les lectures des encodeurs. En fessant cela, le voltage qui les alimentaient à grimper à plus de 16V et les à brûler les amplis-op. Pour remédier au problème, on les a alimentés à l’aide du « switching power supply » de 5V.

On a également eu beaucoup de problèmes électriques avec les circuits intégrés hip4081. Durant un moment inattendu, le courant monte plus haut que 5 A et brûle le circuit intégré du Hip4081. Malheureusement, on a passé les deux dernières semaines à essayer de trouver la cause de ce problème sans jamais le trouver. On a vérifié les soudures beaucoup de fois, on a vérifié les entrées et les sorties des circuits intégrés et l'on a ajouté des temporisations dans la programmation pour éviter de changer de sens trop brusquement.

Juste avant la journée de la présentation, nous avons brûlé un Raspberry Pi en voulant regarder les tentions de sortie à cause d’une mauvaise manipulation avec les sondes de l’oscilloscope. Marc Fortin a pris la gentillesse d’acheter un nouveau Raspberry PI pour continuer à programmer.

Site internet : On a eu beaucoup de problèmes avec la position des titres par rapport au Canvas et au tableau. C’était difficile de bouger un élément sans influencer les autres, mais j’ai réussi à trouver une combinaison qui est belle esthétiquement et qui fonctionne sur différente résolution d’écran.

On a voulu afficher la position de l’objet afin d’avoir un outil de plus lors du débogage. Cependant, afficher l’image de la caméra ralentissait terriblement l’analyse de l’image et n’était pas une bonne méthode. Heureusement, on a trouvé une solution pour afficher notre objet sans réduire les performances. On envoie les coordonnées X et Y de l’objet par MQTT. Ensuite, on les affiche dans un carré sur le site internet à la position qui représente les coordonnées.

Mécanique :

Nous voulions finir la mécanique dès le mois de février. Malheureusement, par la demande de Guillaume, la mécanique a été reportée au mois d’avril ou, mai parce qu’il n’était pas sur des emplacements des capteurs et de la caméra. Ce qui a ralenti la programmation du robot de 2 jours. Finalement, on a diminué nos attentes dans la programmation du robot pour avoir un robot fonctionnel le jour de la présentation.

L’une des deux roues commençait toujours ces mouvements avant l’autre. Elle était mal serrée et il y avait beaucoup de mouvement rotationnel permis. En la resserrant, on a réglé le problème.

Après avoir complété l’installation de plaquettes de cuivre à l’avant du robot. On a voulu poser un capteur infrarouge au même emplacement, mais il n’avait plus de place à cause des plaquettes de cuivre. On a décidé de faire une insertion afin de poser le capteur à l’intérieur du robot pour que le robot soit encore capable de se connecter à la station de recharge.

On a aussi eu des problèmes au niveau de l’isolation de la boîte. Nous avons cru que se manque d’isolation créait des courts-circuits dans le circuit où le HIP4081 brûlerait. La bonne nouvelle est que l’isolation a été réglée en ajoutant du ruban électrique dans toutes la boîte. La mauvaise nouvelle est que cela n’a pas réglé le problème du HIP4081.

Planification :

Presque qu’à chaque rencontre, nous étions en retard dans notre échéancier. Nous avons surestimé nos capacités à construire le robot dans un temps raisonnable. Nous n’avions pas prévu des jours pour déboguer les problèmes si nous en avions.

Solution : nous avons fait des heures supplémentaires durant le soir ou le matin pour nous permettre d’arriver à temps sur l’échéancier. Nous avions aussi diminué nos attentes dans la programmation du robot. Par exemple, le robot peut seulement suivre une couleur au lieu d’une forme déterminée.

# Améliorations possibles

Voici les améliorations que nous voudrions apporter au projet si cela était à refaire :

Julie : J’aurai aimé mettre des lumières au robot pour lui donner plus de personnalité. J’aurai aussi voulu mettre plus de temps dans la conception électronique de la carte pour nous permettre d’ajouter des lumières et des fusibles protecteurs. De plus, j'aurais aimé refaire le circuit électronique à l’extérieur avec les corrections apportées après les tests. Finalement, j’aurais utilisé seulement des capteurs infrarouges pour remplacer les capteurs ultrasons.

Pour une meilleure flexibilité dans le projet, j’avais pensé de mettre un système Bluetooth qui permettrait au robot de nous suivre à l’aide d’un signal Bluetooth au lieu d’une caméra, mais il faudrait faire des recherches pour savoir si c’est possible.

Guillaume : Premièrement, la période d'installation de librairies aurait été beaucoup moins pénible si j'avais fait plus de recherche avant de choisir un logiciel de programmation. J'aurais dû faire le projet sur Windows avec un logiciel qui supporte mieux la cross compilation afin de nous faciliter la tâche. De plus, si j'avais à le refaire, j'utiliserais une plateforme qui supporte deux caméras en les utilisant comme l'œil humain pour connaître l'angle des objets et leur distance. Un grand avantage d'utiliser cette méthode est la précision lors du stationnement à la station de charge, en sachant l'angle, on aurait pu s'orienter pour être centré de loin plutôt qu'utiliser une technique qui ressemble à essaie erreur. Finalement, j'aurais fait faire le circuit électrique à l'extérieur à cause de tous les problèmes électriques qu'on a eus.

# Conclusion

Finalement, nous avons développé nos capacités à régler les problèmes électriques et les problèmes de programmation, à construire un projet du début à la fin, à travailler en équipe, à gérer notre temps dans un projet et à lire les fiches techniques des composantes pour le bon fonctionnement des composantes électriques. Tout ceci nous permettra d’être de bon technicien dans notre carrière.

# Annexe

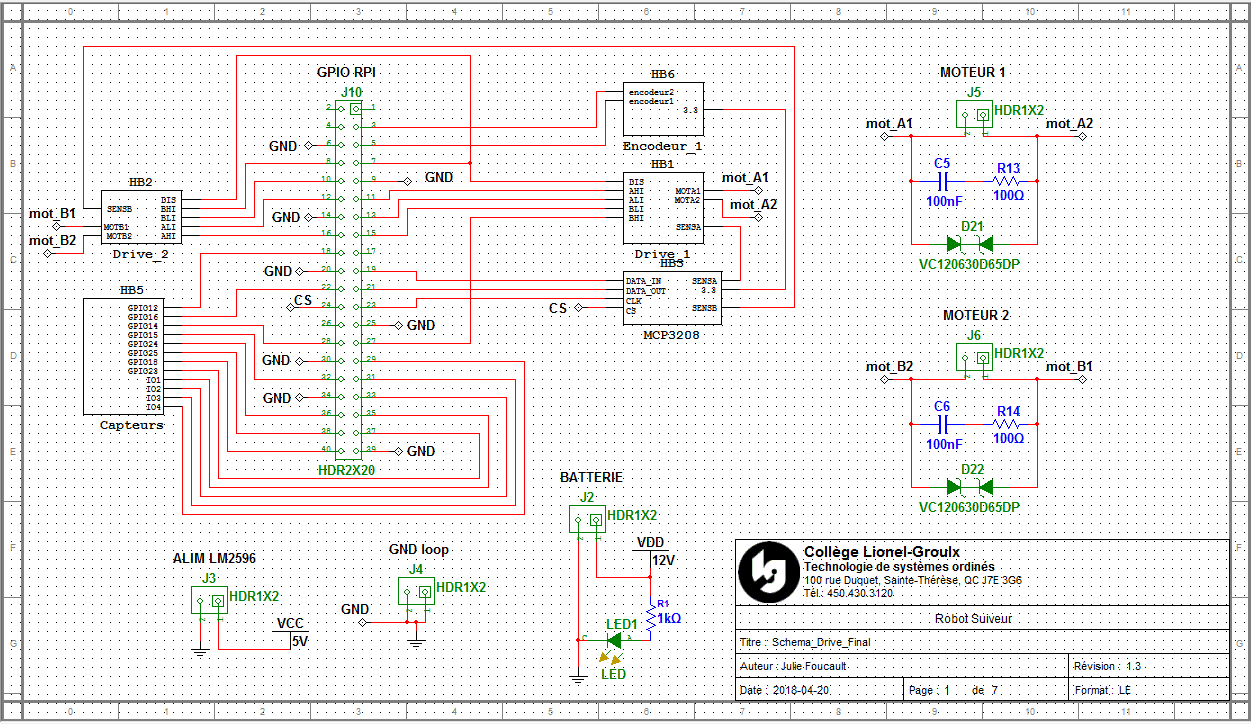
## Code source

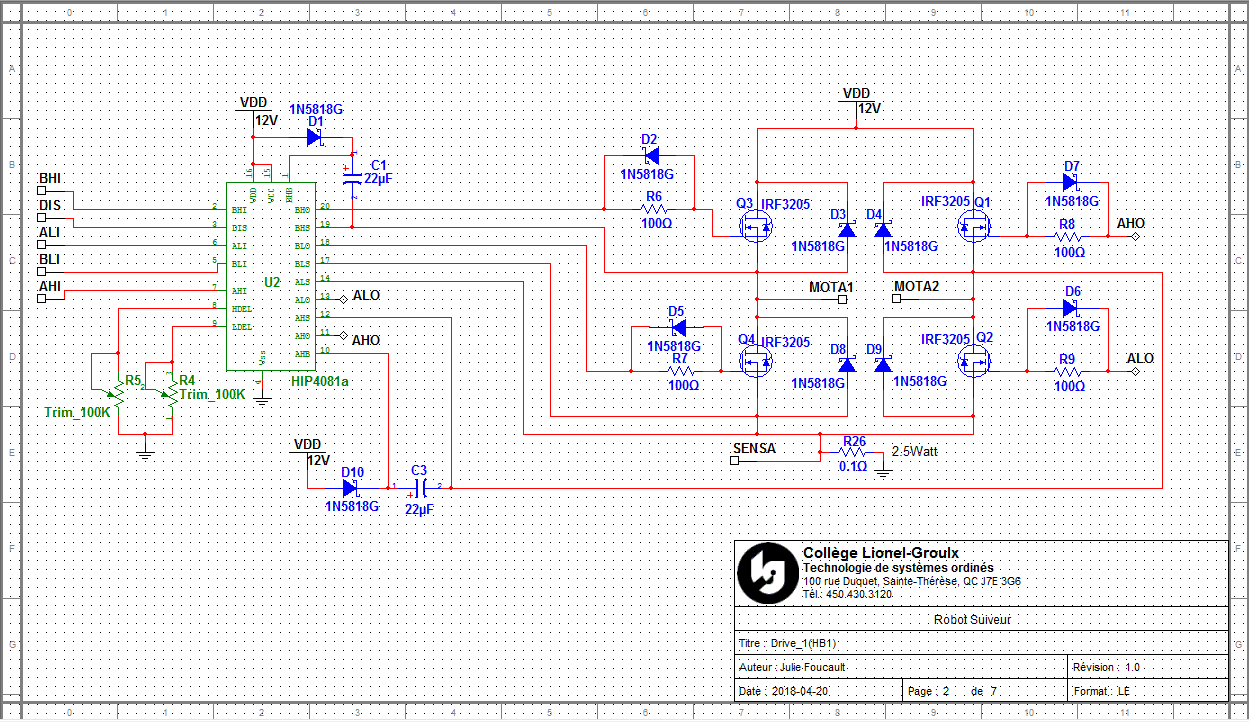
Tous les documents sont dans le fichier annexe.

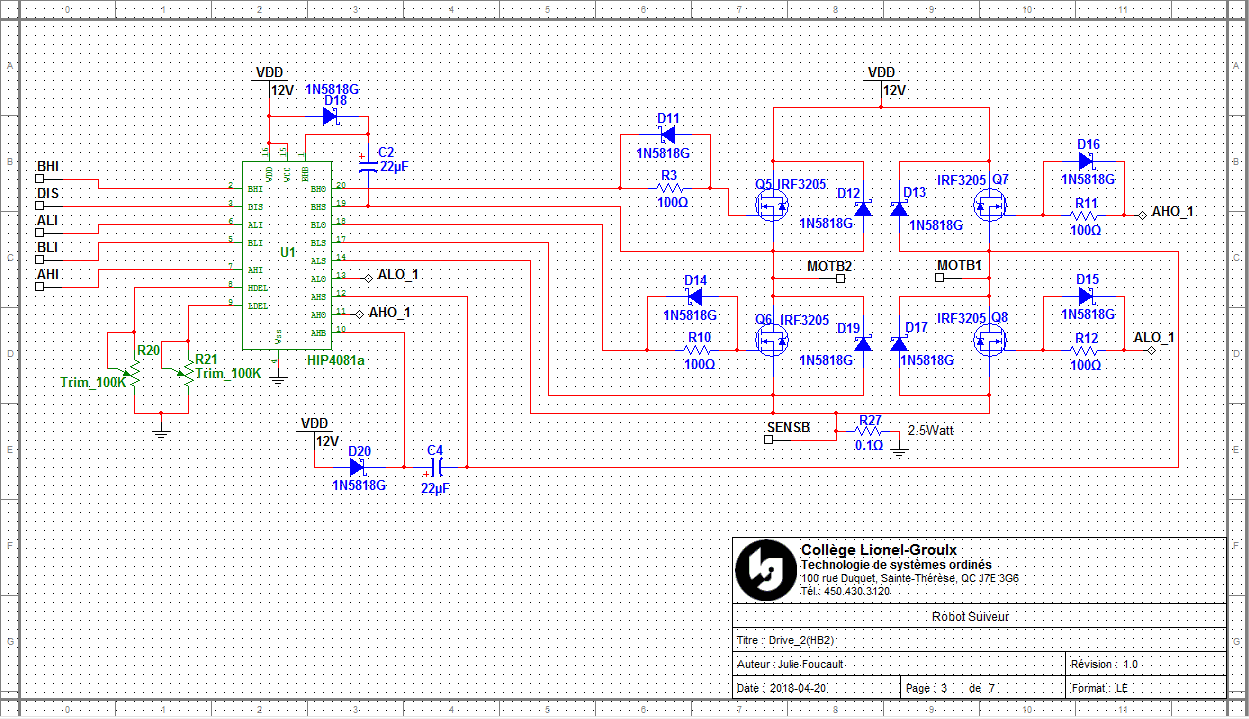
## Schéma(s) électrique(s)

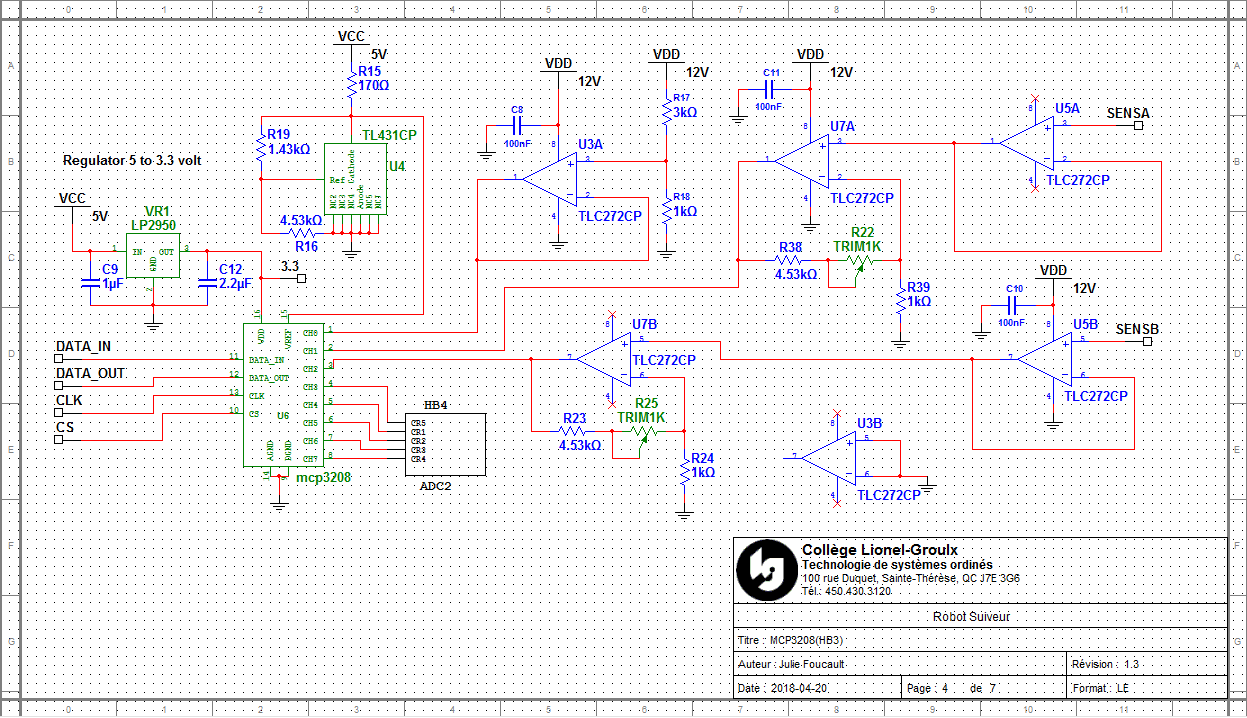
Multisim :

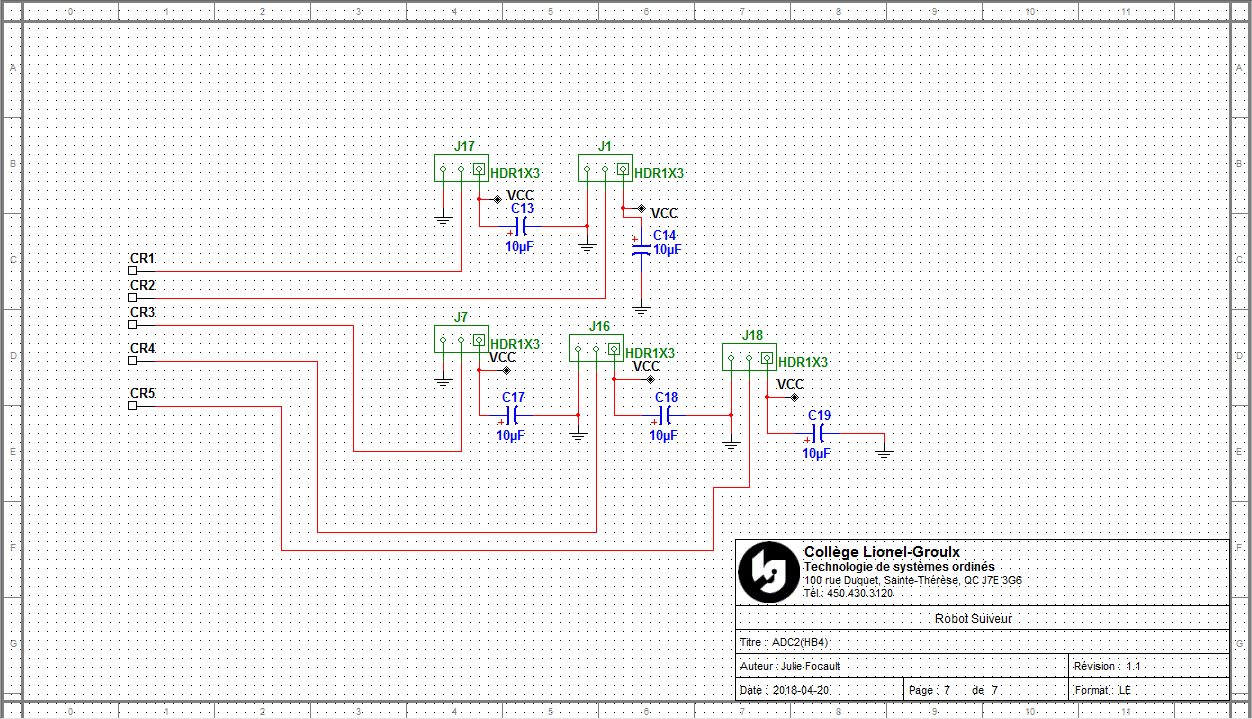
Robot :

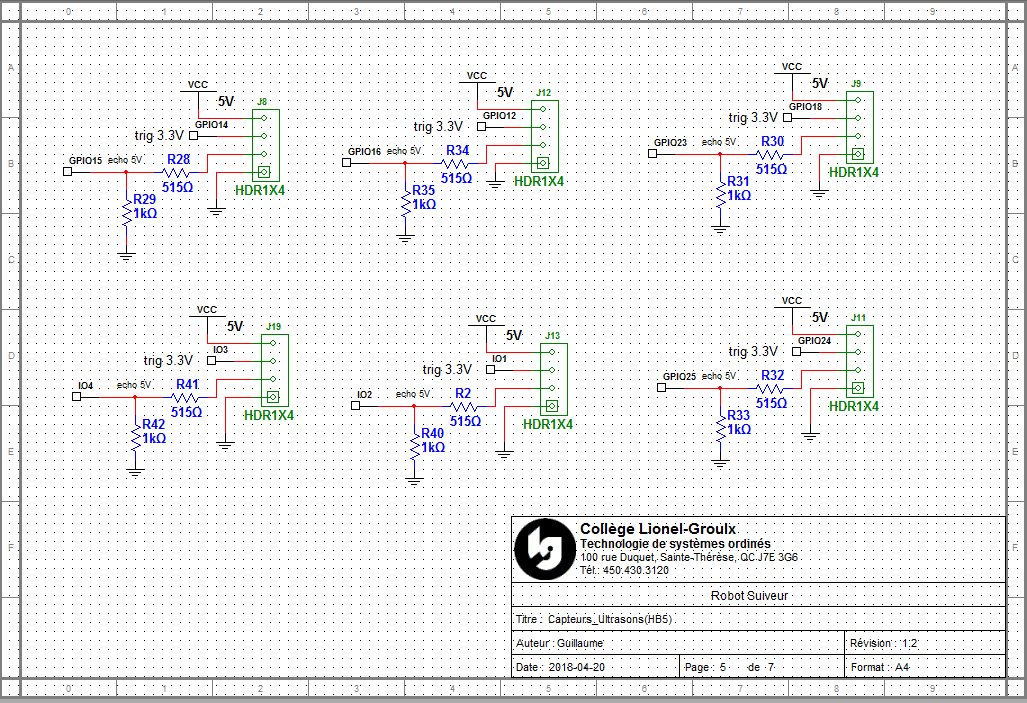


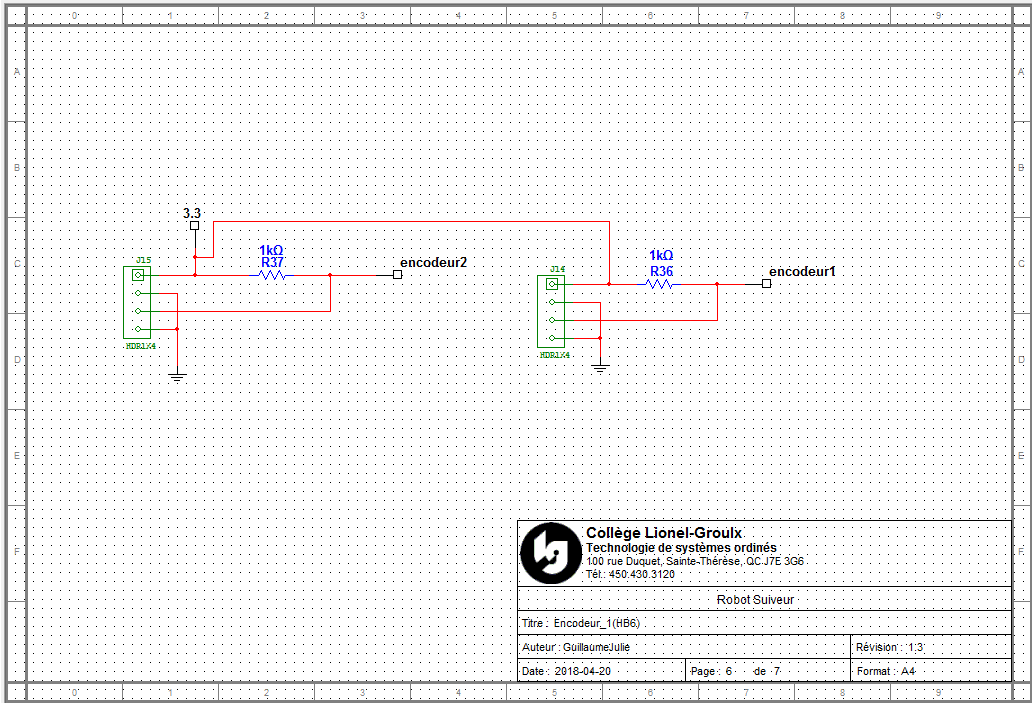


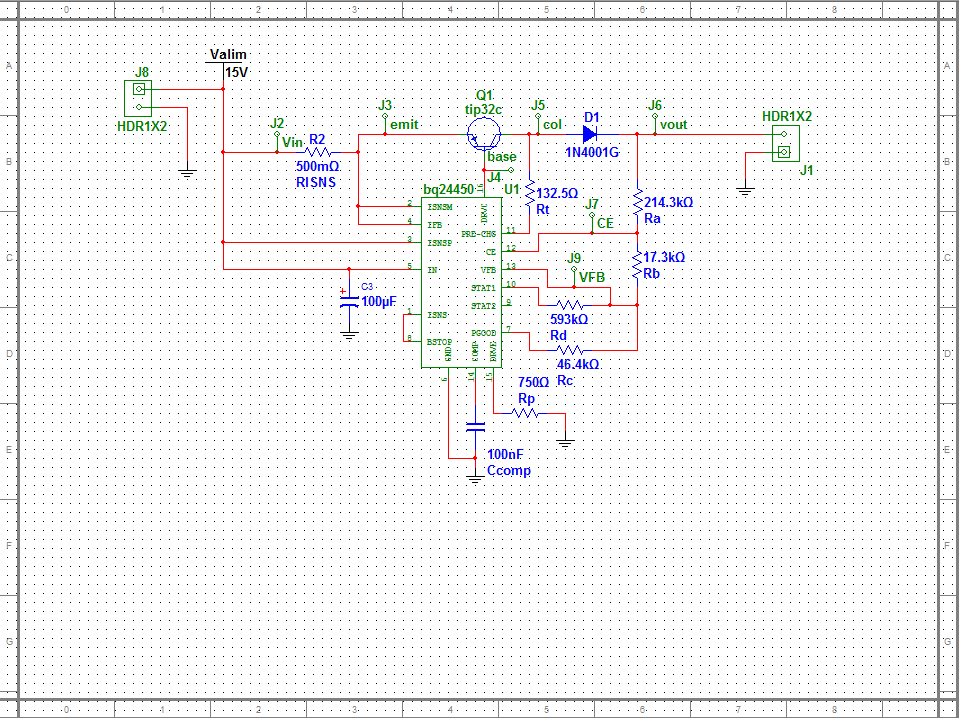








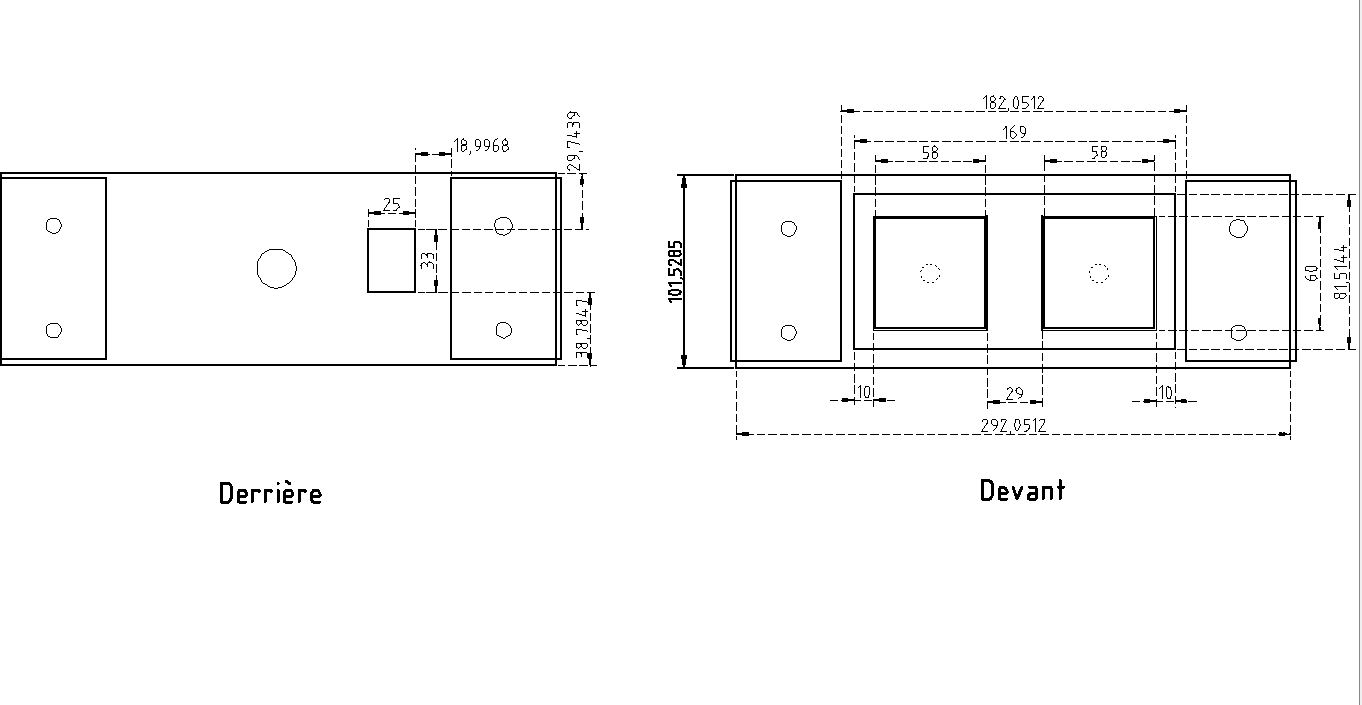


Station de Charge :

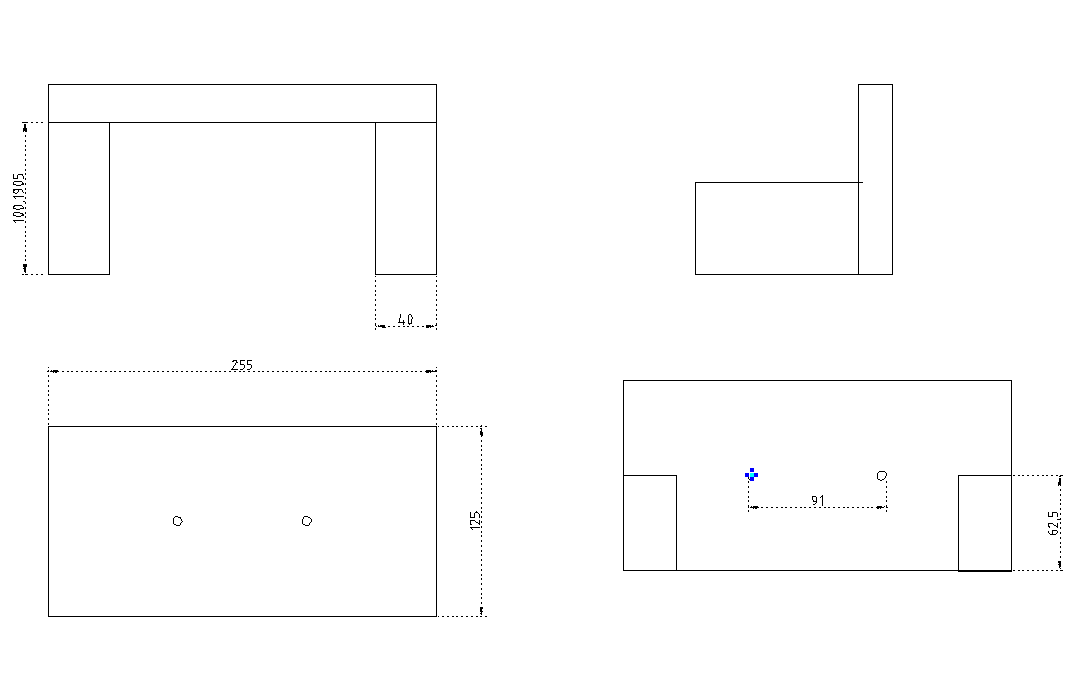
## Schéma mécanique

LibreCad :

Robot :



Station de Charge :



# Référence :

Relier à la programmation :

Librairie :

<https://projects.drogon.net/raspberry-pi/wiringpi/functions/>

exemple d’un projet d’un robot qui suit un objet :

<https://www.youtube.com/watch?v=3BJFxnap0AI>

Installer les logiciels et utiliser putty :

<https://www.youtube.com/watch?v=2LeXT9tBpeI>

Relier à la drive :

Moteur :

<http://www.robotshop.com/ca/fr/moteur-24v-engrenage-491-encodeur-devantech.html?gclid=EAIaIQobChMIlKDmsvqn1gIVELnACh1DbwpIEAYYBCABEgIm0PD_BwE>

Explication complète de système infrarouge :

<http://elonics.in/breadboard-projects/infrared-ir-proximity-obstacle-sensor-using-lm-358>

composante infrarouge :

<https://www.robotshop.com/media/files/pdf2/gp2y0a21yk_e.pdf>

ADC :

<http://f-leb.developpez.com/tutoriels/raspberry-pi/conv-analogique-numerique/>

<http://raspberrypi.pagesperso-orange.fr/dossiers/19-00.htm>

Programmation Guillaume:

[Open](https://l.facebook.com/l.php?u=http%3A%2F%2Fbookencore.com%2Flogiciel-dessin-industriel-gratuit%2F&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg) cv:  
[https://github.com/HesselM/rpicross\_notes/blob/master/06-xc-opencv.md](https://l.facebook.com/l.php?u=https%3A%2F%2Fgithub.com%2FHesselM%2Frpicross_notes%2Fblob%2Fmaster%2F06-xc-opencv.md&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
  
  
[https://elinux.org/RPi-Cam-Web-Interface](https://l.facebook.com/l.php?u=https%3A%2F%2Felinux.org%2FRPi-Cam-Web-Interface&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
[https://www.pyimagesearch.com/2015/02/23/install-opencv-and-python-on-your-raspberry-pi-2-and-b/](https://l.facebook.com/l.php?u=https%3A%2F%2Fwww.pyimagesearch.com%2F2015%2F02%2F23%2Finstall-opencv-and-python-on-your-raspberry-pi-2-and-b%2F&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)

[https://hackaday.io/project/7008-fly-wars-a-hackers-solution-to-world-hunger/log/23068-installing-opencv-on-a-raspberry-pi-the-easy-way](https://l.facebook.com/l.php?u=https%3A%2F%2Fhackaday.io%2Fproject%2F7008-fly-wars-a-hackers-solution-to-world-hunger%2Flog%2F23068-installing-opencv-on-a-raspberry-pi-the-easy-way&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)

[https://www.eclipse.org/forums/index.php/t/418691/](https://l.facebook.com/l.php?u=https%3A%2F%2Fwww.eclipse.org%2Fforums%2Findex.php%2Ft%2F418691%2F&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)

[https://github.com/robidouille/robidouille/tree/master/raspicam\_cv](https://l.facebook.com/l.php?u=https%3A%2F%2Fgithub.com%2Frobidouille%2Frobidouille%2Ftree%2Fmaster%2Fraspicam_cv&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
  
[https://www.pyimagesearch.com/2015/10/26/how-to-install-opencv-3-on-raspbian-jessie/](https://l.facebook.com/l.php?u=https%3A%2F%2Fwww.pyimagesearch.com%2F2015%2F10%2F26%2Fhow-to-install-opencv-3-on-raspbian-jessie%2F&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
  
[https://gist.github.com/willprice/c216fcbeba8d14ad1138#file-install-opencv-sh](https://l.facebook.com/l.php?u=https%3A%2F%2Fgist.github.com%2Fwillprice%2Fc216fcbeba8d14ad1138%23file-install-opencv-sh&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)

[https://github.com/andygrove/raspicam-facedetect-cpp/blob/master/example.cpp](https://l.facebook.com/l.php?u=https%3A%2F%2Fgithub.com%2Fandygrove%2Fraspicam-facedetect-cpp%2Fblob%2Fmaster%2Fexample.cpp&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
  
[https://www.pyimagesearch.com/2015/05/04/target-acquired-finding-targets-in-drone-and-quadcopter-video-streams-using-python-and-opencv/](https://l.facebook.com/l.php?u=https%3A%2F%2Fwww.pyimagesearch.com%2F2015%2F05%2F04%2Ftarget-acquired-finding-targets-in-drone-and-quadcopter-video-streams-using-python-and-opencv%2F&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
[https://github.com/joaofaro/KCFcpp](https://l.facebook.com/l.php?u=https%3A%2F%2Fgithub.com%2Fjoaofaro%2FKCFcpp&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
[https://www.learnopencv.com/object-tracking-using-opencv-cpp-python/](https://l.facebook.com/l.php?u=https%3A%2F%2Fwww.learnopencv.com%2Fobject-tracking-using-opencv-cpp-python%2F&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
[https://github.com/OmarAflak/HC-SR04-Raspberry-Pi-C-](https://l.facebook.com/l.php?u=https%3A%2F%2Fgithub.com%2FOmarAflak%2FHC-SR04-Raspberry-Pi-C-&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
[https://www.opencv-srf.com/2010/09/object-detection-using-color-seperation.html](https://l.facebook.com/l.php?u=https%3A%2F%2Fwww.opencv-srf.com%2F2010%2F09%2Fobject-detection-using-color-seperation.html&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
<https://www.windowscentral.com/how-set-and-manage-ftp-server-windows-10>  
<http://aishack.in/tutorials/tracking-colored-objects-opencv/>

Mqtt :  
[https://xperimentia.com/2015/08/20/installing-mosquitto-mqtt-broker-on-raspberry-pi-with-websockets/](https://l.facebook.com/l.php?u=https%3A%2F%2Fxperimentia.com%2F2015%2F08%2F20%2Finstalling-mosquitto-mqtt-broker-on-raspberry-pi-with-websockets%2F&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
[http://www.instructables.com/id/Installing-MQTT-BrokerMosquitto-on-Raspberry-Pi/](https://l.facebook.com/l.php?u=http%3A%2F%2Fwww.instructables.com%2Fid%2FInstalling-MQTT-BrokerMosquitto-on-Raspberry-Pi%2F&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
[https://lonesysadmin.net/2013/02/22/error-while-loading-shared-libraries-cannot-open-shared-object-file/](https://l.facebook.com/l.php?u=https%3A%2F%2Flonesysadmin.net%2F2013%2F02%2F22%2Ferror-while-loading-shared-libraries-cannot-open-shared-object-file%2F&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)

GDB :  
[http://darkdust.net/files/GDB%20Cheat%20Sheet.pdf](https://l.facebook.com/l.php?u=http%3A%2F%2Fdarkdust.net%2Ffiles%2FGDB%2520Cheat%2520Sheet.pdf&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
[https://www.youtube.com/watch?v=sCtY--xRUyI](https://l.facebook.com/l.php?u=https%3A%2F%2Fwww.youtube.com%2Fwatch%3Fv%3DsCtY--xRUyI&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)

pwm :  
[http://www.hertaville.com/rpipwm.html](https://l.facebook.com/l.php?u=http%3A%2F%2Fwww.hertaville.com%2Frpipwm.html&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
[https://www.raspberrypi.org/forums/viewtopic.php?t=172673&p=1104755](https://l.facebook.com/l.php?u=https%3A%2F%2Fwww.raspberrypi.org%2Fforums%2Fviewtopic.php%3Ft%3D172673%26p%3D1104755&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)  
<https://projects.drogon.net/raspberry-pi/wiringpi/software-pwm-library/>  
[https://www.raspberrypi.org/forums/viewtopic.php?f=33&t=67381](https://l.facebook.com/l.php?u=https%3A%2F%2Fwww.raspberrypi.org%2Fforums%2Fviewtopic.php%3Ff%3D33%26t%3D67381&h=ATP_IXQFaQHwZ5MeEoHZeGcZqoC2_H3RpSy4b1-Qvk8u43r4NFxomDzLh5SYumI9J5kcvxBlx9c-AobIOhM96zeM4QqSi4JSgfBU2JpmnwaQWnOXWgWovg)