

PROJET DE MASTER 2

Autoscope

Second compte rendu

Auteurs :

Thomas ABGRALL
Clément AILLOUD
Thibaud LE DOLEDEC
Thomas LEPOIX

MASTER Systèmes Embarqués

E.S.T.E.I.

École Supérieure des Technologies Électronique, Informatique, et Infographie
Département Systèmes Embarqués

9 janvier 2019

Table des matières

Table des matières	1
I Partie de groupe	2
1 Évolution du cahier des charges	3
1.1 Nécessité du traçage d'astre	3
1.2 Nécessité d'une centrale inertielle et d'un GPS	4
1.3 Changement de SoC	4
II Thomas ABGRALL	6
2 Motorisation	7
2.1 Placement des moteurs	7
2.2 Choix des moteurs	9
2.3 Contrôle des moteurs	9
2.4	10

Première partie

Partie de groupe

Chapitre 1

Évolution du cahier des charges

Plusieurs éléments du cahier des charges ont été amenés à changer depuis la première revue de projet.

1.1 Nécessité du traçage d'astre

Le traçage d'astre, vu au départ comme une fonctionnalité intéressante, s'est imposé comme une fonctionnalité nécessaire. En effet il peut être particulièrement difficile pour une personne non initiée à l'astronomie de positionner le télescope vers un astre précis ou de reconnaître un astre que l'on observe. Il nous est donc apparu primordial que le télescope permette d'affranchir l'utilisateur de la nécessité d'avoir des bases en astronomie pour observer le ciel.

En étudiant les solutions disponibles nous avons trouvé des logiciels de simulation du ciel, comme par exemple le logiciel libre Stellarium.

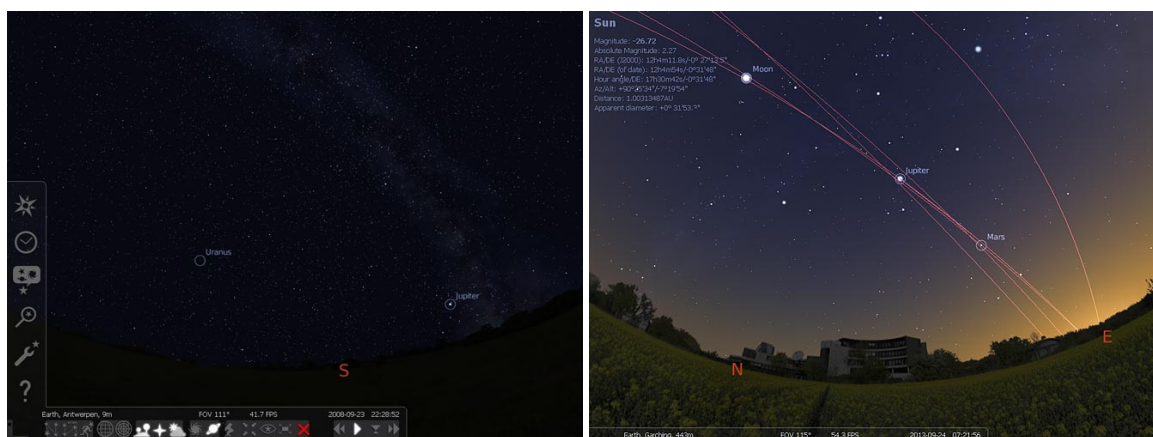


FIGURE 1.1 – Captures d'écran de Stellarium

Celui-ci permet notamment :

- De connaître les coordonnées d'un astre par rapport à l'endroit sur terre où se situe l'observateur.
- De s'orienter dans le ciel selon des coordonnées.
- De s'interfacer avec d'autres systèmes logiciels et/ou matériels

Nous avons donc décidé dans un premier temps de développer une interface pour piloter le télescope depuis un ordinateur distant doté de Stellarium. Puis éventuellement d'embarquer Stellarium dans l'ordinateur du télescope. Ainsi Stellarium fera partie intégrante de son interface utilisateur.

Celle-ci pourrait être alors un menu discret permettant de passer de l'exploration virtuelle du ciel à la vue correspondante à travers le télescope à d'autres éléments comme un dispositif d'amélioration de la qualité des images prises.

1.2 Nécessité d'une centrale inertielle et d'un GPS

L'utilisation d'un logiciel de traçage d'astre tel Stellarium nécessite la compatibilité du télescope avec les coordonnées d'azimut et d'élévation couramment utilisées en astronomie. Il est également nécessaire pour cela de savoir de quel endroit sur terre le télescope observe le ciel, d'où l'utilisation d'un GPS.

Pour connaître l'azimut et l'élévation, il faut avoir des repères dans les deux dimensions. Un magnétomètre permet de déterminer la direction du nord et un accéléromètre permet de connaître la direction du sol, c'est à dire la verticale.

Une centrale inertielle est un composant intégrant un magnétomètre, un accéléromètre et un gyroscope. Elle permet de connaître directement les coordonnées absolues de son orientation dans l'espace.

1.3 Changement de SoC

L'utilisation de Stellarium requiert au minimum 512MiB de RAM et 1GiB pour une utilisation optimale. Or la carte PICO-PI ne dispose que de 512MiB de RAM, elle est donc incapable de faire fonctionner Stellarium correctement.

Nous avons choisi une Raspberry Pi 3 B avec 2GiB de RAM en remplacement. En dépit de ses faibles capacités d'industrialisations, la Raspberry Pi a l'avantage d'être populaire dans le milieu de l'électronique amateur, c'est-à-dire le public le plus susceptible d'être intéressé par ce genre de projet.

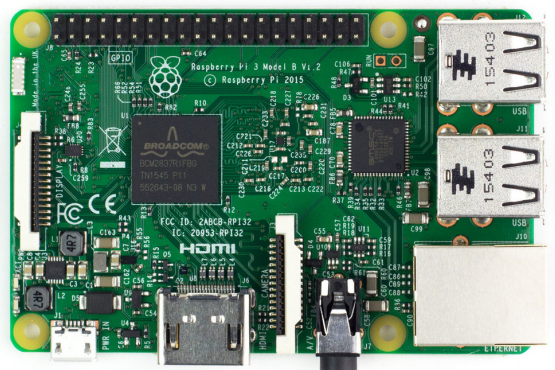


FIGURE 1.2 – Captures d'écran de Stellarium

La caméra et l'écran utilisés ne seront donc plus ceux fournis avec la PICO-PI mais ceux de la Raspberry Pi, à savoir :

- Le module Raspicam v2.1 intégrant une caméra IMX219 de 8Mpx.

L'écran demeure toutefois une option, celui-ci sera intégré si le télescope embarque Stellarium.

Deuxième partie

Thomas ABGRALL

Chapitre 2

Motorisation

Le projet repose sur un projet de télescope manuel, qui devait être placé à la main pour viser les étoiles. Nous allons voir ici les choix et mise en oeuvre réalisés pour automatiser notre télescope.

2.1 Placement des moteurs

Le télescope est composé de deux axes à automatiser : la rotation et l'inclinaison.

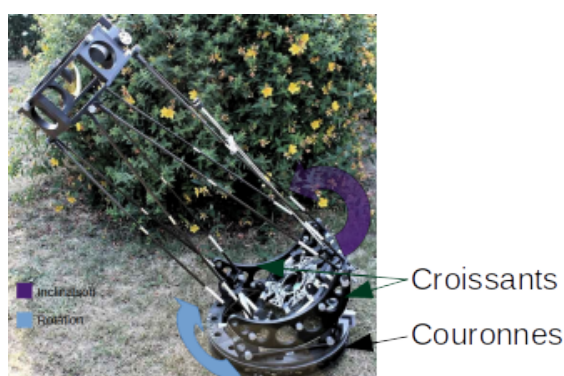


FIGURE 2.1 – Illustration des axes telescope

Pour mouvoir le télescope en rotation, nous avons choisi de placer un moteur sur la couronne fixe de la base du télescope. Qui ce dernier ferai tourner la couronne du dessus qui sera équipée d'une courroie sur ça face intérieure.

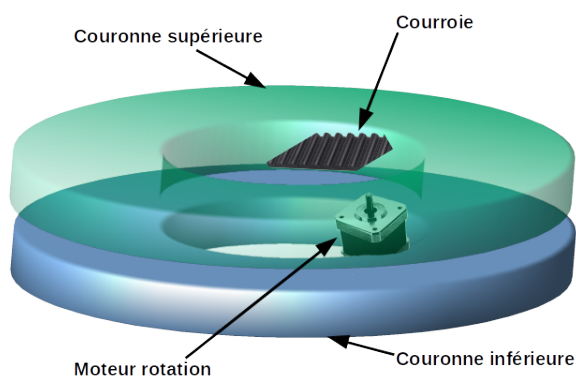


FIGURE 2.2 – Illustration placement moteur rotation

Avant de choisir cette solution nous avons réfléchi sur plusieurs autres. Nous avons envisagé de placer une courroie sur la face intérieur du croissant. La forme non-circulaire de l'intérieur du croissant empêche de placer un axe FIXE pour entrainer une courroie car la forme du croissant va se faire rapprocher la courroie de l'arbre du moteur durant ça progression. Une autre solution visée à placer le moteur au centre de la courroie, cependant vous avons conclus que la surface sur laquelle va s'appliquer la force du moteur est plus petite que dans la solution que nous avons retenue.

Pour agir sur l'inclinaison, nous relierons les deux barre qui maintiennent les croissants avec une courroie. Elle sera tendu grâce à l'arbre du moteur et un pignon. Le tout sera placé entre des deux croissants, sous le miroir principale.

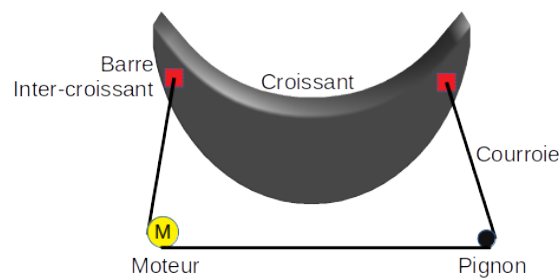


FIGURE 2.3 – Illustration placement moteur inclinaison

Il nous reste ensuite une dernière chose à automatiser, le zoom. Le zoom sera placé entre le miroir secondaire et la caméra. Il sera entouré par une courroie qui fera le lien entre le zoom et une autre courroie entraînée par le moteur zoom.

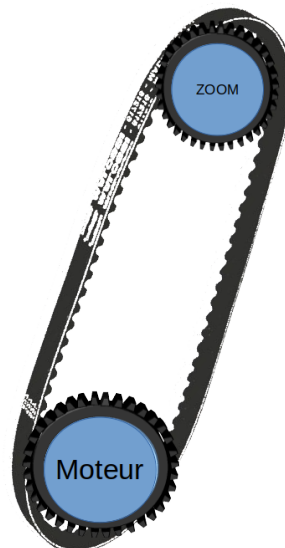


FIGURE 2.4 – Illustration moteur zoom

2.2 Choix des moteurs

D'après la quantité de matière au bout du télescope, les matériels qui seront installés au bout (miroir, zoom, caméra, moteur pour le zoom, fixation) nous avons estimé que doit pouvoir soulever au minimum 4 kilogrammes. Nous avons donc choisi le moteur de référence : 17HM15-0904S. De plus nous disposons d'un tel moteur, avec lequel nous avons pu réaliser des tests pour nous assurer qu'il réponde bien a notre cahier des charges. Le moteur du zoom a besoin de moins d'énergie pour agir sur le zoom, nous avons donc choisi un moteur moins puissant celui retenu est le S20TH30-0604A.



FIGURE 2.5 – Photo du moteur

2.3 Contrôle des moteurs

Pour contrôler le moteur nous avons choisi le composant "A4988 Stepper Motor Driver Carrier". Il permet de contrôler la rotation du moteur, choisir entre le mode pas, demi-pas, quart de pas, huitième de pas et seizième de pas. Il y a également une pin activation et endormissement mais que nous n'utiliserons pas.

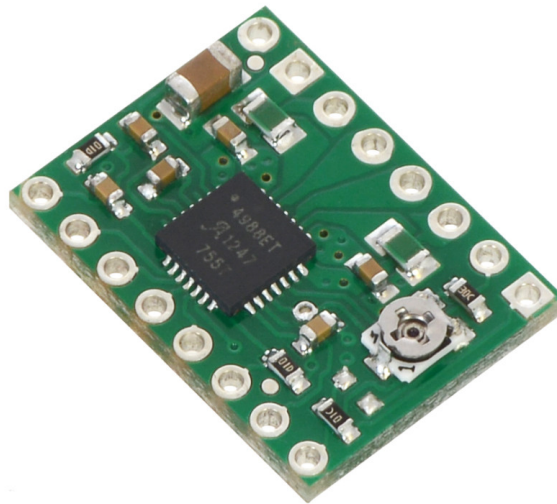


FIGURE 2.6 – Photo du controleur

2.4

Je développe un C un driver linux pour permettre le contrôle des moteurs, mais également de surveiller des interrupteurs.

Le choix de réaliser un drive linux au lieu d'un logiciel est justifié parcequ'il permettra à un programme en couche supérieure d'utiliser ces fonctions pour utiliser les moteurs.

Pour contrôler les moteur faut générer une impulsion, à chaque front montant le moteur réalise un pas si il se trouve dans le mot pas à pas, sinon une demi-pas etc. Il faut également choisir le sens de rotation du moteur

Fonctionnement :

Lors de l'initialisation du drive les interruptions de chacun des interrupteurs se mette en service, dès lors si les interrupteurs placer en fin de course de la rotation s'active alors le moteur de rotation est immédiatement arrêté. Il en est de même pour l'inclinaison. Cette partie est autonome, et ne doit pas être utilisé par des logiciels tiers. A l'inverse du contrôle des moteurs. Car chaque moteur dispose d'un fonction qui sera appelé par un logiciel tiers pour lancer un moteur sur un certain nombre de pas et dans une certaine direction.

Avancement :

Le contrôle des moteurs rotation et inclinaison sont réalisés. Les interruptions des interrupteurs de rotations sont également réalisées.

Travail à venir :

Réalisation du code permettant de contrôler le moteur zoom.

Contrôle des modes des moteurs (mode de pas) en fonction de la distance à parcourir. Quand le moteur s'approche de la fin on réduit la taille des pas pour augmenter la précision, afin d'arriver exactement là où il faut.