TP SYSTEME CAMERA

23/11/2021

BERTHIER Thomas

Boucher Louis

CAUET Clément

# Page d’évaluation du TP

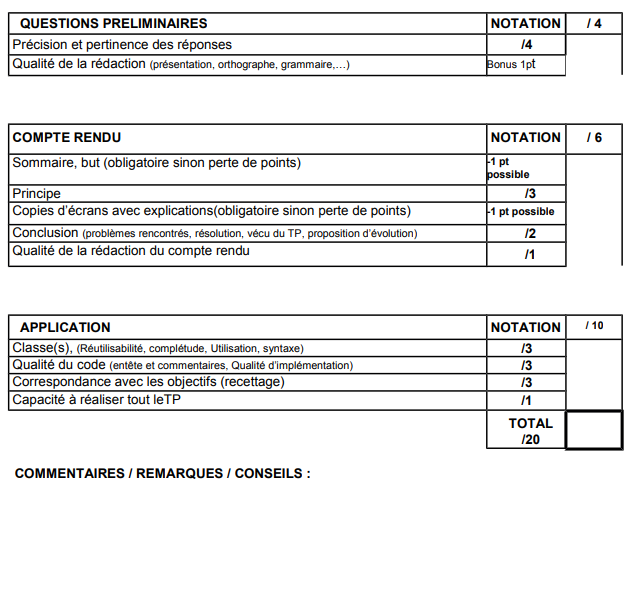


Table des matières

[Page d’évaluation du TP 1](#_Toc88573865)

[But 3](#_Toc88573866)

[Principe 3](#_Toc88573867)

[Réponses aux questions préliminaires 4](#_Toc88573868)

[Algorithmes 6](#_Toc88573869)

[Code 7](#_Toc88573870)

[Recette 8](#_Toc88573871)

[Conclusion 9](#_Toc88573872)

# But

Le but du projet est de créer un programme capable de contrôler une caméra par liaison série RS232.

# Principe

Pour réaliser ce projet, nous avons tout d’abord répondu aux questions préliminaires pour nous approprier le projet.

Nous avons ensuite dû ouvrir la communication entre le port série et la caméra ainsi que configurer cette ouverture pour pouvoir envoyer des instructions.

Pour finir, nous avons créé un programme capable d’effectuer le pilotage de la caméra avec les critères demandés.

# Réponses aux questions préliminaires

1) Donnez les caractéristiques de la caméra et de ses possibilités de déplacements

La caméra est une EVID30 reliée par RS232 à un ordinateur puis reliée en VHS sur un écran pour afficher l’image capturée. La caméra peut se déplacer à gauche, à droite, en haut et en bas.

1)Comment sont transmises les données sur la liaison série (expliquez le fonctionnement de la transmission)

Les données sont transmises par RS232, la commande est envoyée à la camera par liaison série et la caméra récupère la commande et l’exécute.

3) Différence entre liaison synchrone et asynchrone ?

Dans la transmission synchrone, les données sont transférées sous la forme de trames, alors que dans la transmission asynchrone, les données sont envoyées sous forme d'octet ou de caractère.

4) Donnez le format de transmission (vitesse …) de la caméra ?

Le format de transmission de la caméra est de 9600 bauds / seconde.

5) Que veut dire la chaîne de caractères (code ascii) suivante pour la caméra : : 0x81,0x01,0x04,0x00,0x03,0xff

0x81 est le Header

0x01 est le Com-mode

0x04 est la Category

0x00 et 0x03 sont le Message

0Xff est le Terminator

6) Donnez les chaînes Ascii qui permettent de :

- Allumer (Allumer la caméra) - Eteindre (Eteindre la caméra)

Allumer : 8x 01 04 00 02 FF

Eteindre : 8x 01 04 00 03 FF

# Algorithmes

On effectue tout d’abord l’ouverture du port série et sa configuration puis on créer un timer.

On ajoute ensuite les différents slots qui permettent d’envoyer les instructions à la caméra.

On créer l’IHM qui permettra l’appel des slots grâce à des boutons.

Pour finir on ajoute un système de déplacements automatiques qui s’effectue toutes les 5 secondes permettant d’effectuer 3 fois de suite un ensemble de motifs répétitifs qui balaye l’ensemble de la pièce.

# Code

Lien Github contenant le code :

<https://github.com/ThomasBerthier/TPdmx>

# Recette

|  |  |  |
| --- | --- | --- |
| Fonctionnalités | ETAT (OK ou NOK) | Commentaires |
| Allumer caméra via Hercules | OK |  |
| Allumer caméra via IHM CPP | OK |  |
| Zoomer au maximum ou au minimum en fonction d’un curseur logiciel. | OK |  |
| Déplacer la camera dans les Axes X et Y à vitesse fixe. | OK |  |
| Mode automatique qui permet de balayer, toute la salle à partir du point de référence, et 3 fois de suite. | OK |  |

# Conclusion

Ce projet aura été une réussite car il fonctionne parfaitement, de plus il nous aura permis d’apprendre le fonctionnement de lampes pilotable.