**Projet de fin de session**

**ANNEXE «  STATION DE PESAGE  »**

**Session 5**

**Programme de TSO**

**Cours :**

**247-616-LI**

**247-627-LI**

**247-637-LI**

**247-647-LI**

**Présenté par:**

**Samuel Duclos**

**Présenté à :**

**Simon Ayoub**

**Stéphane Deschênes**

**Yves Roy**

**Date : 14 décembre 2020**

[1.1 Station de pesage 3](#_Toc58860244)

[Tableau 1: Composantes matérielles du uARM 5](#_Toc58859288)

[Figure 1: composantes matérielles de la station de pesage 3](#_Toc58859624)

[Figure 2 : architecture logicielle de la station de pesage 4](#_Toc58859625)

[Figure 3 : diagramme d’états pour la gestion de la station de pesage 7](#_Toc58859626)

## **Station de pesage**

Figure 1: composantes matérielles de la station de pesage

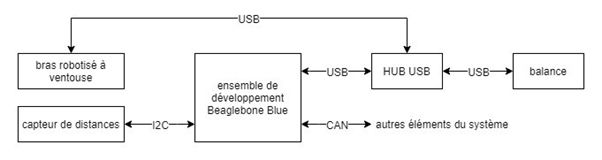


Figure 1: architecture matérielle de la station de pesage

Pour pouvoir s'intégrer au système, la station de pesage devra pouvoir au minimum :

* Reconnaître et récupérer les messages CAN qui lui seront d'intérêts ;
* Décoder les messages qui lui permettront de savoir si elle doit être en arrêt ou en opération ;
* Décoder les messages qui lui permettront de savoir si elle doit produire des valeurs de poids en grammes ou en onces ;
* Émettre des messages qui permettront de connaître s'il est en fonction, en erreur ou en arrêt ;
* Se mettre en arrêt ou en opération ;
* Opérer à l'aide de pilotes, d'interfaces, de processus et de services et opérer au besoin à l'aide de machines à états, d'une base de temps, d'un service de base de temps, de processus Linux, de fils d'exécution, de tuyau, de mutex et d'appels système ;
* Utiliser son bras robotisé et son capteur de distances pour localiser les blocs à peser ;
* Procéder au déchargement du véhicule et déposer les blocs à peser sur le plateau de sa balance ;
* Déterminer le poids des blocs en interagissant avec sa balance ;
* Déposer les blocs dans une unité d'entreposage (e.g. un plat) après leur pesée ;
* Émettre des messages qui permettront de connaître le poids des blocs.

La Figure 2 présente l'architecture logicielle de la station de pesage. La documentation pertinente de conception (requis logiciels, architecture logicielle, diagrammes d’états, *etc.*), d'assemblage et de programmation liée à toutes les stations est jointe au rapport de projet (aussi disponible sur Github) La documentation détaillée est incluse dans les programmes à l’aide du paramètre « --help », où chaque paramètre est expliqué.

Figure 2 : architecture logicielle de la station de pesage

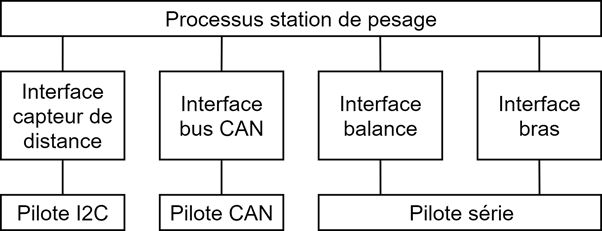


Figure 2: architecture logicielle de la station de pesage

Le tableau suivant décrit les composantes de l'architecture matérielle du uARM.

Tableau 1: Composantes matérielles du uARM

| No | Item | Description |
| --- | --- | --- |
| 1 | buzzer | Peut jouer « funky town » |
| 2 | Port ISP | RESET-PC6(RESET)->D13-PB5(SCK)->D12-PB4(MISO)->  VCC(+5V)-> D11-PB3(MOSI)->GND;  Le port PB/PC est le ATmega328p. |
| 3 | USB à série | FT232RL |
| 4 | ATmega328p | Microcontrôleur |
| 5 | Connecteur servo 3 (contrôle du bout) | GND->VCC(+5V)->D10- PB2(SS)->A3-PC3(ADC3);  D10 est une sortie PWM et A3 est une entrée analogique. |
| 6 | Connecteur servo 1 (contrôle servo gauche) | GND->VCC(+5V)->D13-PB5(SCK)->A0-PC0(ADC0);  D10 est une sortie PWM et A3 est une entrée analogique. |
| 7 | Interrupteur D4-PD4 (T0) | Condition d’échange uARM |
| 8 | MINIUSB-A | Télécharge logiciel et entrées/sorties |
| 9 | Port Bluetooth | D1-PD1(TXD)->D0-PD0(RXD)->VCC(+5v)->GND |
| 10 | Connecteur servo 0 (contrôle de la base du bras) | GND→VCC(+5)→D12→PB4(MISO)→A1PC1(ADC1). D12 est une sortie PWM et A1 est une entrée analogique |
| 11 | Connecteur servo (gripper) | GND→VCC(+5)→D9→PB1(MISO)→A6(ADC6). D9 est une sortie PWM et A6 est une entrée analogique |
| 12 | Alimentation | 5V, entrée >5A |
| 13 | Interrupteur D7-PD7 (AIN1) | Change la contion du uARM |
| 14 | Connecteur servo 2 (servo droit) | GND→VCC(+5)→D11→PB3(MOSI)→A2(PC2). D11 est une sortie PWM et A2 est une entrée analogique |
| 15 | Connecteur servo (réservé) | GND→VCC(+5)→D8→PB0(ICP)→A7(ADC7). D8 est une sortie PWM et A7 est une entrée analogique |
| 16 | Connecteur de la pompe | GND→VCC(+5)→D8→PB0(ICP)→A7(ADC7). D8 est une sortie PWM et A7 est une entrée analogique |
| 17 | « Limit switch » | GND->D2 |
| 18 | LDO | 5V transfer 3.3V |
| 19 | EEPROM | 64KB |
| 20 | Reset |  |

La figure suivante montre le diagramme d'état de la station de pesage. Lorsque la bonne couleur est détectée au centre de tri, celui-ci change un bit de la trame CAN, qui passe par le centre de commande puis est reçu à la station de pesage. Ce bit indique que la routine qui attend le véhicule peut démarrer. Lorsque le véhicule arrive devant la balance, le bras se positionne et démarre sa routine de balayage jusqu’à ce que le capteur de distance le retrouve. Le bras place le bloc sur la balance, attend le poids que retourne la balance par le port série, puis entrepose le bloc sur le camion qui peut continuer sa propre routine. L’exécution de cette machine d’états continue en boucle.

Figure 3 : diagramme d’états pour la gestion de la station de pesage

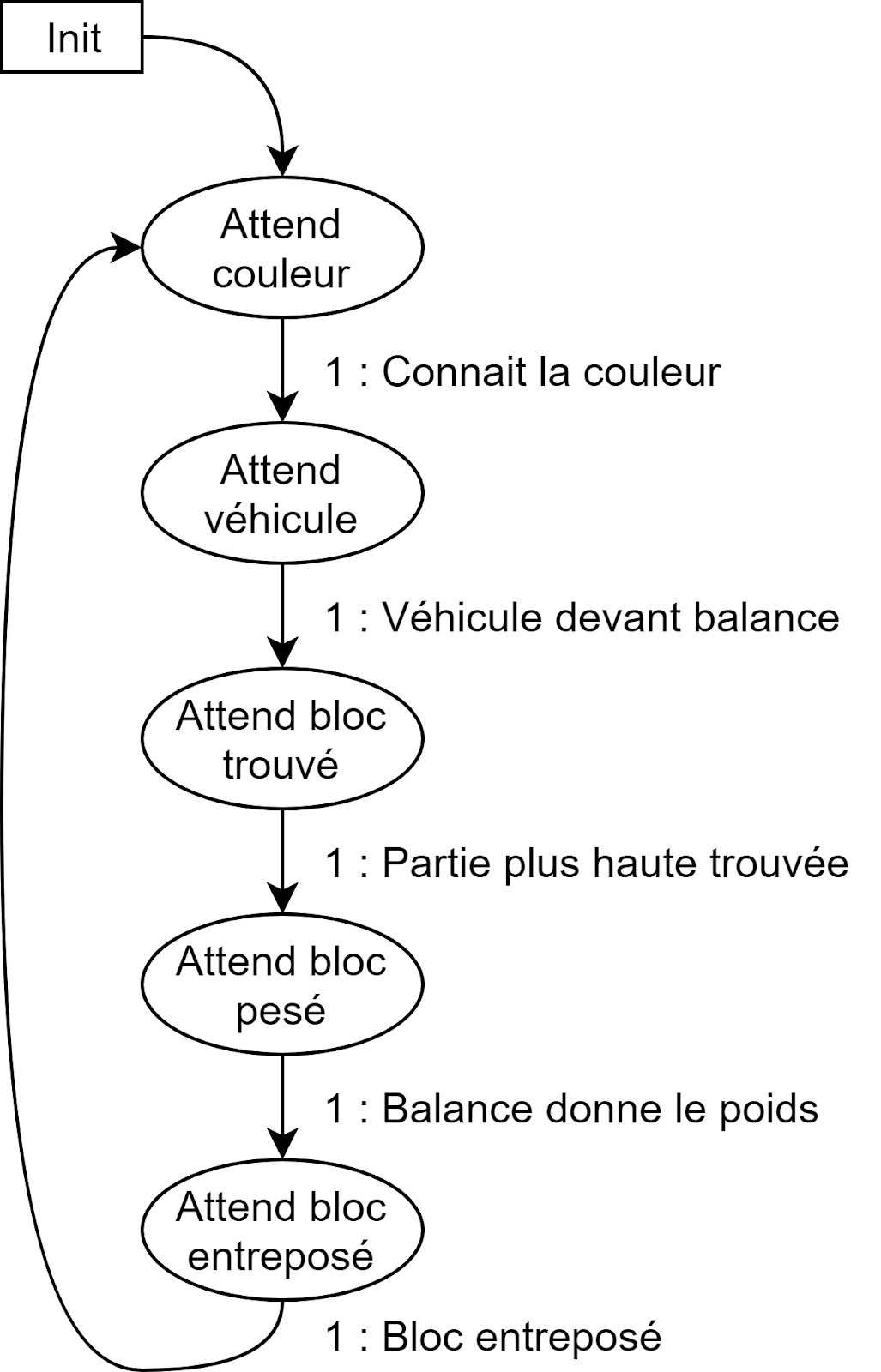


Figure 3 Diagramme d'états pour la gestion de la station de pesage