

Le sujet comporte trois parties indépendantes :

- Architecture logicielle
- Architecture matérielle
- Communication et Réseau

Le candidat peut traiter ces parties dans un ordre quelconque à la condition de bien identifier sur sa copie la partie traitée.

## 1. Architecture logicielle

Les caisses sont reconnues par les lecteurs Code Barre (poste B1). A ce moment, un ordre de préparation de commande au magasin automatique est lancé.

Chaque caisse poursuit son chemin après avoir reçu les médicaments jusqu'au poste B2 où elle est de nouveau identifiée (*voir dossier technique : plan général de l'atelier*).

- **Q1.1** : Proposer une structure de données dynamique permettant de représenter le cheminement des caisses entre le poste B1 et le poste B2 ?

Si un intervenant supprime une caisse sur le tapis d'acheminement entre le point B1 et le point de remplissage des médicaments en sortie du magasin automatique :

- **Q1.2** : Quelles sont les conséquences de cette manipulation au poste B2 ?

Une modélisation partielle du système est représentée par un diagramme de classe donné en annexe 1 du sujet .

- **Q1.3** : Ecrire en langage C++ la déclaration de la classe CBonLivraison.  
*On rappelle que cette déclaration sera définie dans un fichier appelé "CBonLivraison.h".*

L'automate de contrôle du tapis reçoit ces ordres sous formes d'une chaîne de caractères de la forme suivante : **STXCOOOONNETX**

- STX : 1 caractère STX ( Start of text) de code ASCII 0x 02
- C : 1 caractère. Coté du tunnel: '1' coté gauche, '0' coté droit.
- OOOO: 4 caractères. Offset Tapis de l'éjecteur par rapport au rail.
- NN : 2 caractères. Nombres de boites à éjecter.
- ETX : 1 caractère ETX ( End of text) de code ASCII 0x 03

La classe CBonLivraison possède une méthode *ConvertirCE()* définie comme suit :  
IOString& CBonLivraison ::ConvertirCE( int C , int OT, int Nbre)

C : entier. Coté de l'éjecteur : 1 coté gauche, 0 coté droit.

OT : entier. Offset Tapis

Nbre : entier. Nombre de boîtes à éjecter.

L'annexe 4 du dossier technique donne un extrait de la documentation de la classe IOString.

➤ **Q1.4** : En utilisant les fonctionnalités offertes par la classe IOString, donnez le codage en C++ de la méthode *ConvertirCE()* qui retourne la chaîne de caractères désirée.

On étudie maintenant la partie magasin automatique et en particulier la logique qui permet d'éjecter le bon médicament sur le tapis (voir dossier technique : constitution du magasin automatique)

On précise :

Le Code d'Ejection comporte deux champs:

- Un champs *Id\_coté* (codé sur 1 bit) déterminant le coté du tunnel où se situe le rail
  - *Id\_coté* = 1 => rail coté gauche du tunnel
  - *Id\_coté* = 0 => rail coté droit du tunnel
- Un champs *Nb\_topcodeur* (codé sur 12 bits) déterminant l'Offset Tapis du rail par rapport à l'origine tapis compté en nombre d'impulsions codeur.

Les rails sont distants de **5cm**

➤ **Q1.5** : Proposez un type de donnée permettant de manipuler aisément le Code Ejection. Donnez alors la déclaration en C++ du type de donné que vous avez retenu.

On donne en annexe 2 du sujet ( *extrait modèle relationnel*) une représentation de la base de données du poste superviseur. Cet extrait représente la constitution de la base de données relatif aux médicaments du magasin automatique et est composée des tables MEDICAMENT, ZONEATELIER et LOCALISER.

➤ **Q1.6** : Ecrire la requête SQL permettant de connaître l'ensemble des codes géographiques concernant le médicament de code CIP **3326357**.

## 2 Architecture matérielle

Dans cette partie nous étudierons l'architecture d'une balance.

Nous rappelons qu'une balance est constituée d'une carte à base d'un MC68332 dont une documentation partielle est fournie dans le dossier technique en annexe 1.

Le schéma de la carte 68332 donnée en annexe 3 du sujet se limite, pour simplifier l'étude, aux seuls éléments indispensables permettant de répondre aux questions suivantes :

### 2.1 Etude des zones ROM et RAM de la carte

( voir schéma carte 68332 *annexe 3 du sujet* )

➤ **Q2.1.1** : Identifier parmi les composants U1, U2, U3 ceux contenant la mémoire morte et la mémoire vive du système. Justifiez ?.

➤ **Q2.1.2** : Quelle est la capacité ( en nombre d'octets) de mémoire morte et de mémoire vive disponibles ?

➤ **Q2.1.3** : Par quels signaux du microcontrôleur sont sélectionnés les boîtiers U1, U2 et U3 ?

➤ **Q2.1.4** : A partir des éléments de l'annexe 1 du dossier technique MC68332, donner l'espace d'adresse ( première et dernière adresse) utilisé par la mémoire morte après le RESET de la carte.

➤ **Q2.1.5** : Les registres CSBAR0, CSOR0, CSBAR1 et CSOR1 déterminent la génération des signaux /CS0 et /CS1. Vous expliquerez le rôle de ces registres.

➤ **Q2.1.6** : On souhaite implanter la mémoire vive à partir de l'adresse \$400 000. Donner sur le document réponse les valeurs à inscrire dans les registres CSBAR0 et CSBAR1 ainsi que les valeurs des champs BYTE, et R/W des registres CSOR0 et CSOR1.

## 2.2 Etude de la liaison entre le poste de pesée et une balance

Les balances et le poste de pesée sont reliés par une liaison RS485.

Les questions suivantes utilisent le document constructeur SN75176 en **annexe 6 du sujet** ainsi que le schéma de la partie convertisseur et liaison en **annexe 4 du sujet**.

➤ **Q2.2.1** : Quelle est la plage de tension de l'entrée différentielle du SN75176 pour laquelle la donnée en sortie sur R est indéterminée ?

➤ **Q2.2.2** : Quel est le rôle du cavalier CAV1 ?

➤ **Q2.2.3** : Sur quelle(s) balance(s) située(s) sur la liaison RS485 devez vous mettre CAV1 en place ? (voir *annexe 7 du sujet*)

## 2.3 Etude du convertisseur

Le document constructeur AD7851 en **annexe 5 du sujet**, présente le convertisseur appartenant à la chaîne de mesure du poids des caisses contenant les boîtes de médicaments du bon de livraison.

➤ **Q2.3.1** : Quel est le nombre de bits du convertisseur ?

➤ **Q2.3.2** : Les valeurs mesurées sont elles transmises du convertisseur vers le 68332 en parallèle ou en série ?

L'**annexe 5 du sujet** donne des informations sur le convertisseur. Le lancement d'une conversion n'est pas effectué par la ligne /CONVST, mais est déclenché par logiciel à l'aide du module QSM du 68332 relié au convertisseur.

Le QSM peut transmettre des données 16 bits au convertisseur (MOSI vers DIN). Dans le même temps, Il effectue la lecture des données transmises par le convertisseur (DOUT vers MISO).

➤ **Q2.3.3** : Donner la valeur à transmettre au convertisseur pour lancer une mesure et lire la valeur de la mesure précédente en complétant le **document réponse**.

## 3 Communication et Réseau

### 3.1 Organisation physique du réseau

Les postes suivants communiquent par un réseau ethernet, pour chaque machine nous précisons les adresses IP :

Machine	Adresse IP
Serveur gestion	151.148.180.100
Serveur production	151. 148.180.1
Superviseur	151. 148.180.7
Poste de lancement	151. 148.180.3
Poste pesées	151. 148.180.4
Impression commandes	151. 148.180.250
Impression factures	151. 148.180.252

- **Q3.1.1** : Donner le nom de la topologie physique et logique du réseau correspondant à la connexion des machines citées ci-dessus (*voir dossier technique architecture informatique*).

L'adresse IP de l'imprimante « l'impression facture » est : **151.148.180.252** .

- **Q3.1.2** : Donner la classe de ce réseau.

Connaissant les adresses IP suivantes :

Adresse réseau	Adresse sous réseau	Adresse masque sous réseau
151.148.0 .0	151.148.180.0	255.255.224.0

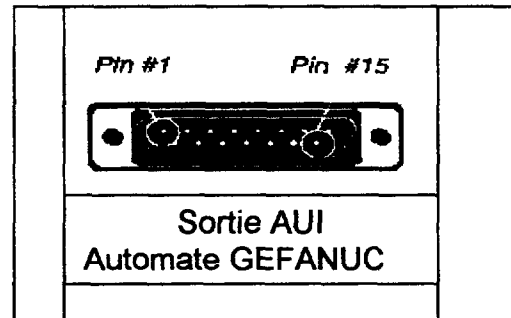
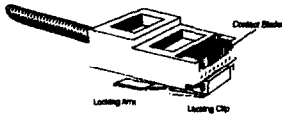
- **Q3.1.3** : Donner le nombre de sous réseaux que l'on peut réaliser en tenant compte de ces adresses .

- **Q3.1.4** : Quel est le nombre maximum de machines que l'on peut interconnecter sur un de ces sous réseau ?

## 3.2 Liaison automate

L'automate GEFANUC est relié sur la deuxième carte « coupleur génius » du poste de lancement.

Vers PC de  
lancement



➤ **Q3.2.1** : En fonction du schéma ci-dessus donner le nom du ou des appareils à prévoir afin de permettre la liaison entre ces deux machines.

➤ **Q3.2.2** : Quel est le numéro de la couche du modèle OSI qui correspond à ce ou ces matériels.

La méthode d'accès du réseau GENIUS est de type jeton. Cette méthode permet de qualifier ce réseau de **déterministe**.

➤ **Q3.2.3** : Expliquer le terme déterministe pour un réseau.

Le mode de transmission de ce réseau est qualifié de **large bande**. Certains réseaux sont qualifiés de **bande de base**.

➤ **Q3.2.4** : Expliquer le terme bande de base pour un réseau.

## 3.3 Liaison balance

Les balances et le poste de pesée sont reliés par une liaison RS485 sans répéteur.

➤ **Q3.3.1** : Justifiez l'utilisation d'une liaison RS 485 pour relier les balances dans l'atelier ?

➤ **Q3.3.2** : Combien de balances pourra-t-on connecter dans l'atelier de production ?