



**LE RÉSEAU DE CRÉATION
ET D'ACCOMPAGNEMENT PÉDAGOGIQUES**

**Ce document a été numérisé par le Canopé de l'académie de Bordeaux
pour la Base nationale des sujets d'Examens de l'enseignement professionnel.**

Ce fichier numérique ne peut être reproduit, représenté, adapté ou traduit sans autorisation.

**BREVET DE TECHNICIEN SUPÉRIEUR
INFORMATIQUE ET RÉSEAUX
POUR L'INDUSTRIE ET LES SERVICES TECHNIQUES**

ÉTUDE D'UN SYSTÈME INFORMATISÉ

Session 2014

Durée : 6 heures
Coefficient 5

Ce sujet comprend :

<i>Partie</i>	<i>Pagination</i>	<i>Couleur des feuilles</i>
Présentation du système et questionnement	pages 1 à 22	Bleu
Annexes	pages 1 à 19	Jaune
Document réponses À RENDRE OBLIGATOIREMENT, AGRAFÉ À UNE COPIE MODÈLE EN	pages 1 à 21	Blanc

Matériel autorisé :

L'usage de la calculatrice est autorisé (circulaire n°99-1 86 du 16-11-1999).

Tout autre document ou matériel est interdit.

Dès que le sujet vous est remis, assurez-vous qu'il est complet.

BTS INFORMATIQUE ET RÉSEAUX POUR L'INDUSTRIE ET LES SERVICES TECHNIQUES	Session 2014
ÉTUDE D'UN SYSTÈME INFORMATISÉ	Code : 14IRSES1

BREVET DE TECHNICIEN SUPÉRIEUR INFORMATIQUE ET RÉSEAUX POUR L'INDUSTRIE ET LES SERVICES TECHNIQUES

ÉTUDE D'UN SYSTÈME INFORMATISÉ

Session 2014

Durée : 6 heures
Coefficient 5

SUJET

(22 PAGES)

Toutes les réponses aux questions sont à fournir sur le livret intitulé « document réponse », à l'exclusion de tout autre support. Ce document sera agrafé à une copie modèle EN.

Les réponses doivent être **exclusivement** situées dans les emplacements prévus à cet effet. Si nécessaire, le candidat peut rectifier ses réponses sur la page non imprimée **en regard**. Une réponse ne doit être justifiée que si la question le demande.

Pour des raisons de confidentialité certaines informations industrielles ont été modifiées.

Temps conseillés et barèmes indicatifs :

A. PRÉSENTATION DU SYSTÈME	30 mn	
B. ANALYSE DU SYSTÈME	45 mn	16 points
C. PROGRAMMATION DU POSTE DE CODAGE	90 mn	24 points
D. RÉSEAUX INDUSTRIELS	90 mn	23 points
E. BASE DE DONNÉES	45 mn	17 points
F. RÉSEAU INFORMATIQUE	60 mn	20 points

Matériel autorisé :

L'usage de la calculatrice est autorisé (circulaire n°99-186 du 16-11-1999).

Tout autre document ou matériel est interdit.

Dès que le sujet vous est remis, assurez-vous qu'il est complet.

PLATEFORME DE TRI AUTOMATISÉ DES OBJETS POSTAUX

A Présentation du système

Les entreprises de presse envoient à des particuliers ou à d'autres entreprises des courriers en nombre tels que des revues (abonnements), publicité, etc.

La société de Traitement de Presse STP traite ce type de courriers appelés en interne « objets postaux ».

L'étude porte sur la plateforme de tri entièrement automatisée de cette société.

A.1 Présentation de la société STP

Créée en 1996, la société STP est une entreprise du groupe de La Poste spécialisée dans le traitement industriel et l'acheminement des objets postaux vers les centres distributeurs de La Poste.

Chaque année, cette société assure le tri de plus de 1,24 milliard d'objets postaux.

A.2 Expression du besoin

La société STP s'est dotée d'une plateforme de tri entièrement automatisée à haute cadence. Les objets postaux pré-triés selon la zone géographique (une partie de la France) sont déposés dans des conteneurs par les clients. Ils doivent être triés de façon entièrement automatisée selon leur destination postale et seront ensuite acheminés vers les centres distributeurs de La Poste.

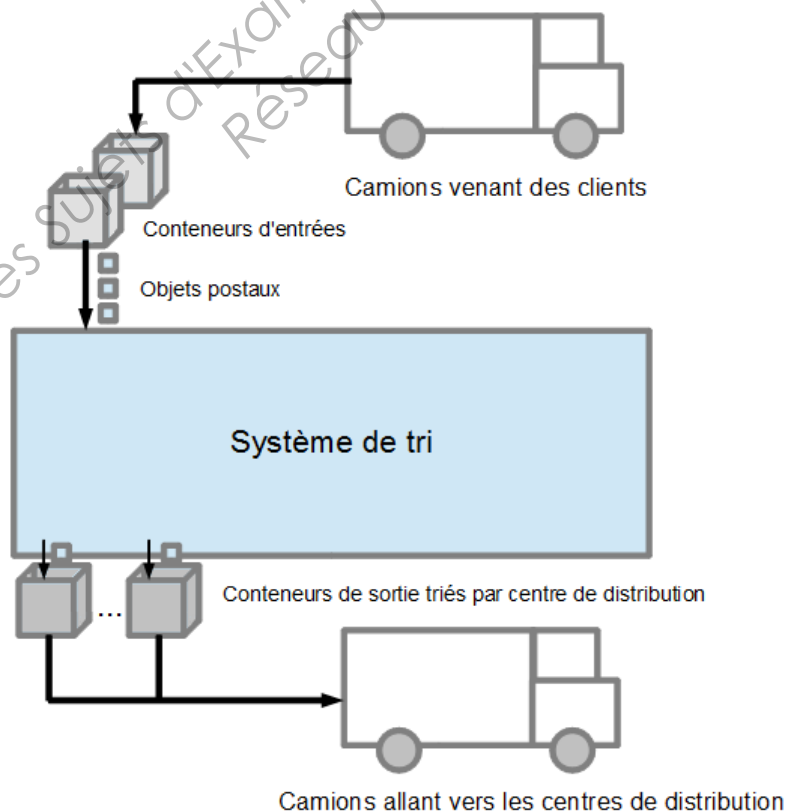


Figure 1 : Représentation du besoin

Si un centre distributeur reçoit peu d'objets, il sera regroupé avec un autre centre proche. Dans cette condition, les objets seront triés selon un **code de regroupement (de codes postaux)**.

A.3 Description du système

- Les conteneurs chargés d'objets postaux à trier, sont déposés par les camions dans l'aire de stockage.
- Ces conteneurs sont ensuite amenés vers les postes de codage où les opérateurs prennent un par un, les objets postaux puis ils saisissent leur code postal sur un clavier ou grâce à un lecteur de code-barres et les déposent sur le tapis du système.
- Les objets postaux introduits dans le système de tri sont transportés jusqu'aux conteneurs de sortie correspondant aux centres postaux de distribution.
- Les conteneurs sont ensuite acheminés par camion jusqu'aux centres de distribution.

Le système de tri peut être schématisé ainsi :

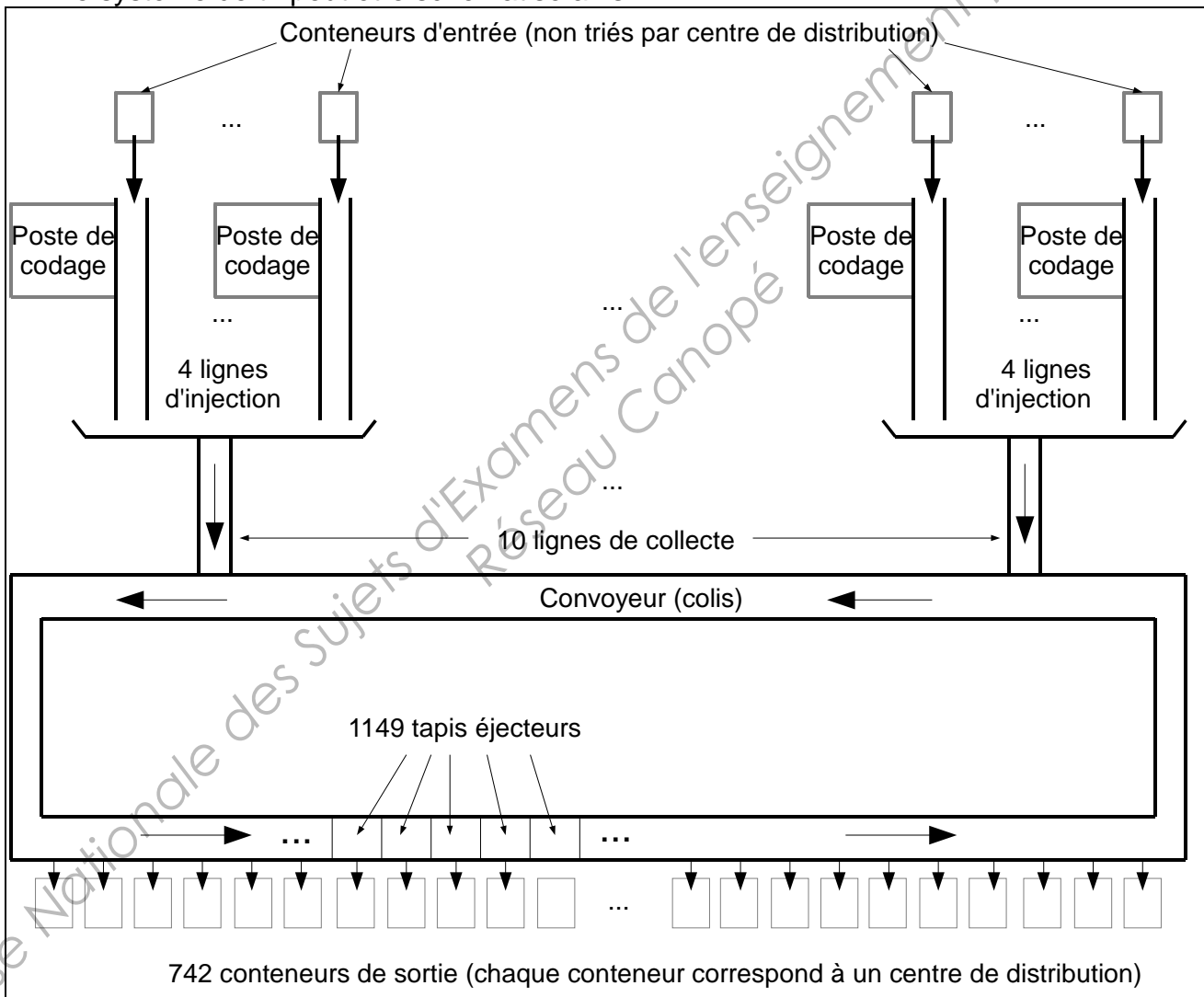


Figure 2 : Schéma de principe du système

Après saisie du code postal et dépose sur le tapis de la ligne d'injection, les objets postaux sont transportés sur la ligne de collecte des objets. Cette ligne attend qu'une place soit libre sur le convoyeur afin d'introduire l'objet. Quand l'objet sur le convoyeur arrive au niveau du conteneur de sortie correspondant à son code postal, l'objet est évacué dans le conteneur.

Toutes les informations sur le tri des objets postaux peuvent être consultées sur un ordinateur de supervision nommé « WCS » (Warehouse Control System).

Le contrôle de fonctionnement du système de tri s'effectue à l'aide d'un ordinateur de supervision nommé « BeOS » (Beumer Operating System).

A.4 Constitution du système

L'architecture matérielle du système de tri est donnée dans le document annexe 1 : « Plan du système de tri ».

Le système de « tri des objets postaux » est composé de :

- Dix lignes de collecte d'objets postaux numérotées de LC01 à LC10 ;
Chaque ligne de collecte est équipée de quatre lignes d'injection, numérotées de LI1 à LI4. Elles permettent aux opérateurs de tri d'introduire les objets postaux dont le code postal est déjà saisi et de les véhiculer sur les tapis éjecteurs.

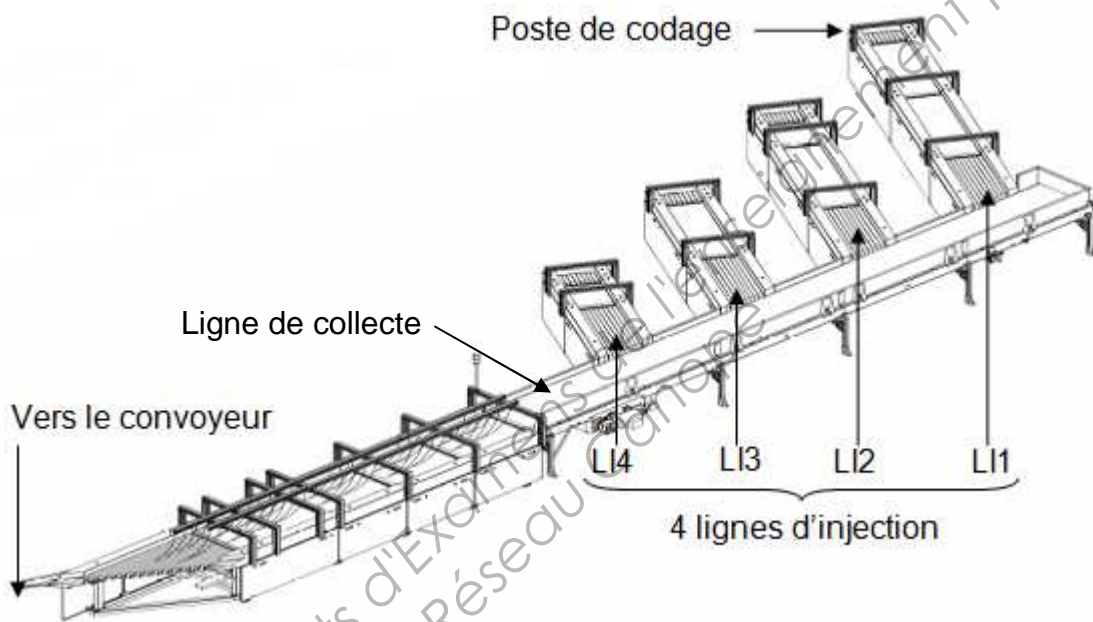


Figure 3 : Ligne de collecte des objets

Pour saisir le code postal des objets, chaque ligne d'injection est équipée d'un poste de codage.

Un poste de codage est constitué

- d'un PC ;
- d'un clavier ;
- d'un lecteur de code-barres sous forme de douchette relié à l'unité centrale par une liaison USB.

Ce poste permet à l'opérateur de tri de saisir le code postal des objets postaux et de visualiser les différentes informations auxquelles il a accès : la date et l'heure, l'état du système (trieur et injecteur) ainsi que les performances du poste (nombre d'objets saisis, nombre d'objets injectés, débit instantané, débit cumulé, etc.). Il permet aussi de voir les données saisies (code postal, numéro départemental) (figure 4).

Le poste de codage est relié au réseau de production par une liaison Ethernet utilisant le protocole TCP/IP.

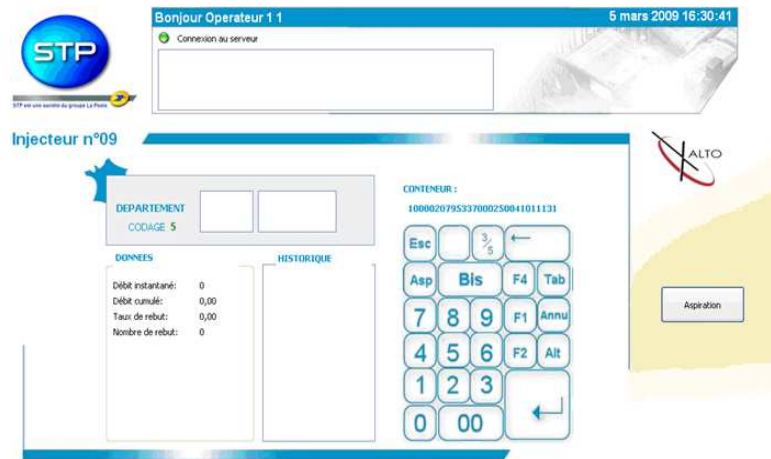


Figure 4 : Fenêtre de saisie

- Un convoyeur constitué d'un assemblage de 1149 tapis éjecteurs qui éjectent les objets vers les conteneurs de sortie (figure 5) ;

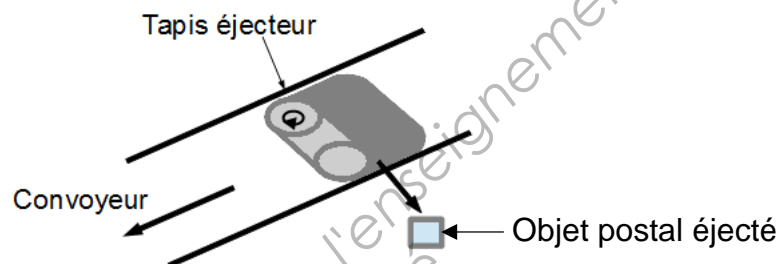


Figure 5 : Tapis éjecteur

- 742 postes de sortie de « conteneurisation » (figure 6).
 Quand un objet postal arrive à la hauteur du bon conteneur, le « tapis éjecteur » envoie l'objet postal dans celui-ci.
 Chaque conteneur de sortie reçoit les objets postaux selon leur code. Ces conteneurs seront ensuite acheminés vers les centres de distribution correspondants.
 Des colonnes de signalisation lumineuse à deux couleurs (orange et verte) ont pour fonction d'indiquer l'état de chaque poste de sortie de conteneurisation (conteneur non présent, conteneur plein, conteneur en attente d'appairage, etc).

Colonne de signalisation lumineuse à deux couleurs

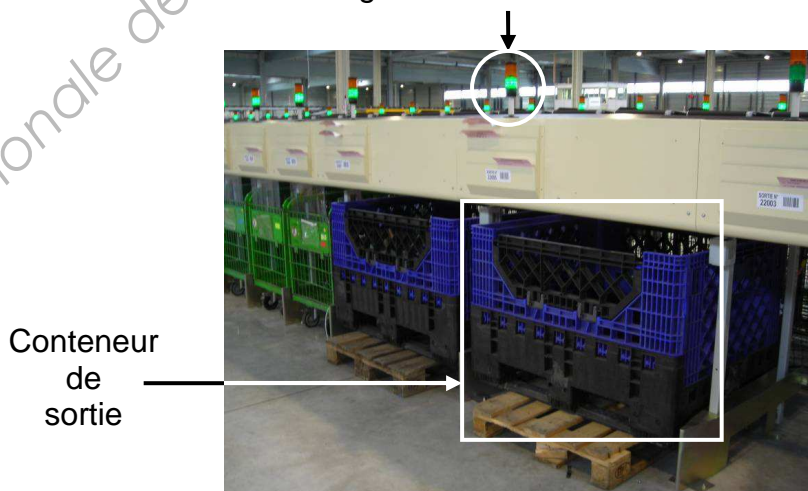


Figure 6 : Postes de sortie de conteneurisation



Figure 7 : Pistolet d'appairage

Pour vérifier que les étiquettes des conteneurs de sortie correspondent bien à la destination réelle donnée par sa position, une opération « d'appairage » est réalisée par les opérateurs de tri lors de la mise en place des conteneurs vides sur les sorties de conteneurisation. Trente pistolets d'appairage (figure 7) sont affectés à cet usage.

Les pistolets d'appairage sont des équipements mobiles qui sont reliés au réseau informatique par une liaison WiFi ;

➤ Un ensemble de matériels et logiciels nécessaires au pilotage de l'installation.

Cet ensemble est composé :

- d'un PC de supervision « BeSS » (Beumer Sorting System) ;
Le responsable d'activités utilise ce poste pour commander et configurer, via le serveur « BeSS » (Beumer Sorting System), le système de tri ;
- d'un PC de « supervision WCS (Warehouse Control System) » qui permet au responsable de qualité de superviser la production et de gérer les tris. Toutes ces données sont stockées dans le serveur de bases de données « WCS » ;
- d'un système de commande de tapis piloté par un ensemble de composants d'automatisme industriel.

Ces différents matériels sont déployés selon le diagramme ci-dessous :

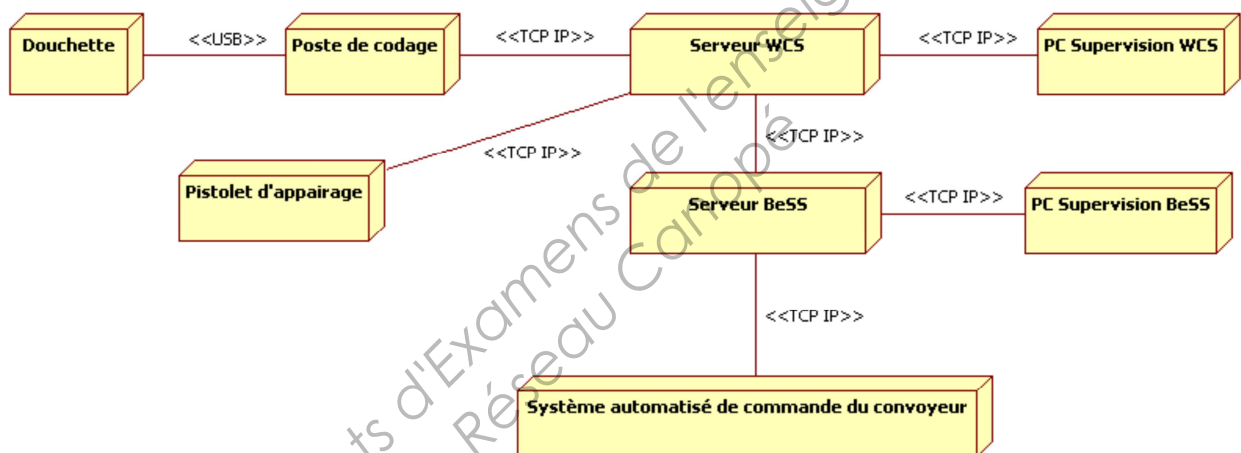


Figure 8 : Diagramme de déploiement partiel

A.5 Éléments d'analyse du système

Voir diagramme de cas d'utilisation figure 9.

Les acteurs intervenant dans le système de tri sont :

- **le responsable d'activités** : il contrôle le fonctionnement du trieur. Il affecte, suivant les charges de travail, les opérateurs de tri aux postes de codage et aux postes de sortie de conteneurisation ;
- **l'opérateur de tri** : il peut être affecté soit à un poste de codage pour charger les objets postaux à trier ou soit à un poste de sortie de conteneurisation ;
- **le responsable qualité** : il réalise et lance le plan de tri. Il supervise également la production ;
- **l'administrateur** : il gère le réseau informatique, les matériels, les logiciels et les utilisateurs du site de tri ;
- **le technicien de maintenance** : il assure le bon fonctionnement du trieur.

Les rôles des acteurs sont :

- **L'opérateur de tri** affecté à un poste de **codage** (chargement des objets) :
 - Fait lire son badge par un lecteur de code-barres afin de s'identifier au début de ses activités ;
 - Place le conteneur à traiter près de son poste de travail ;
 - Lit l'étiquette du conteneur avec le lecteur code-barres ;
 - Prend un par un les objets postaux du conteneur et en saisit le code postal ;
 - Pose les objets encodés sur la ligne d'injection.

- **L'opérateur de tri** affecté au poste de **sortie des conteneurs après le tri** :
 - Fait lire son badge par un lecteur de code-barres afin de s'identifier au début de ses activités ;
 - Place les conteneurs vides et vérifie que la colonne de signalisation lumineuse est au vert clignotant (conteneur présent) ;
 - Scanne le code-barres de la sortie de conteneurisation à l'aide du pistolet d'appairage ;
 - Place et scanne l'étiquette correspondant au plan de tri sur le conteneur ;
 - Vérifie le passage au vert fixe de la colonne de signalisation lumineuse, cela indique que le conteneur est bien pris en compte par le serveur « WCS » ;
 - Retire et amène dans la zone de stockage le conteneur plein (signalement par feu orange sur la colonne de signalisation lumineuse).

- **Le responsable d'activités** utilise le PC de supervision « BeSS » pour :
 - Démarrer et arrêter le trieur ;
 - Configurer et assigner les lignes d'injection ;
 - Affecter les activités aux opérateurs de tri.

- **Le responsable qualité** utilise le PC de supervision « WCS » pour contrôler la qualité de tri :
 - Il consulte les messages et les statistiques de défauts ;
 - Il acquitte les défauts ;
 - Il réalise et lance les plans de tri ;
Le plan de tri consiste à affecter un code postal sur une des sorties de conteneurisation.

- **Le technicien de maintenance** visualise, grâce au PC de supervision « BeSS », les défauts et y remédie. Il effectue également la maintenance préventive.

- **L'administrateur** gère les utilisateurs et édite leurs droits d'accès au système informatique.

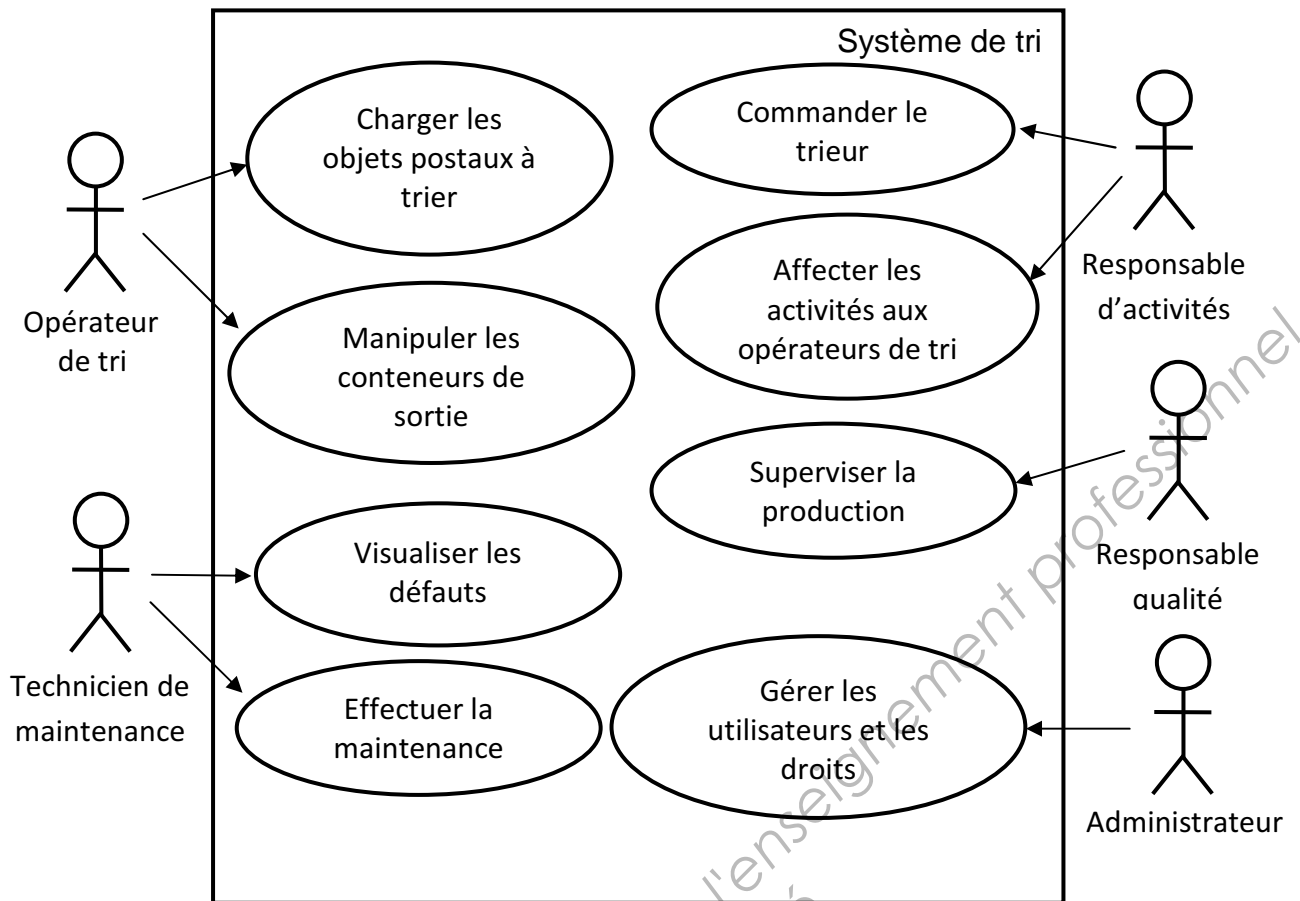


Figure 9 : Diagramme de cas d'utilisation

B Analyse du système

B.1 Fonctionnement du système

On se propose de déterminer le débit du système et la cadence d'un opérateur.

Après l'opération de codage, les opérateurs de tri posent l'objet sur un tapis injecteur qui le transfère sur le convoyeur constitué de tapis éjecteurs. Ces tapis éjectent les objets postaux vers les conteneurs de sortie.

À un instant donné, il n'y a qu'un objet sur chaque tapis éjecteur du convoyeur.

Les caractéristiques du système de tri sont :

Désignation	Données
Longueur du convoyeur	574,50 m
Vitesse du convoyeur	2.0 m/s
Nombre de tapis éjecteurs	1149 unités
Nombre de tapis d'injection	40 unités

Question B.1.1

Calculer le temps mis par le convoyeur pour effectuer un tour complet.

Question B.1.2

Calculer le nombre de tours effectués par le convoyeur en une heure.

Un tapis éjecteur peut donc éjecter un objet postal lorsque le convoyeur a effectué un tour.

Question B.1.3

Déterminer le nombre théorique d'objets postaux que le système est capable de trier en une heure.

On veut vérifier que la cadence des opérateurs de tri imposée par le système est réalisable. Pour cela, on demande de calculer le temps maximal que prend un opérateur pour coder et insérer les objets postaux quand le système est en pleine charge. Vous devez calculer le temps pour 100 objets afin de moyenner cette valeur.

Chaque poste de codage traite 720 objets postaux par heure quand le système de tri est en pleine charge.

Question B.1.4

Calculer le temps maximal en minutes et en secondes nécessaire à l'opérateur de tri pour entrer 100 objets postaux dans le convoyeur.

B.2 Étude UML

Pour conduire l'étude du système, on souhaite détailler les diagrammes de cas d'utilisation et de déploiement.

En particulier, on veut préciser le rôle du « responsable qualité » qui a pour fonction de **superviser la production**. Pour cette supervision, il doit obligatoirement **s'authentifier**. La supervision de la production peut éventuellement consister à **lancer le tri** et à **configurer les paramètres de production**.

En cas de problème, il **acquitte les défauts de sécurité** (il doit aussi s'authentifier dans ce cas).

Question B.2.1

Compléter la partie du diagramme de cas d'utilisation du document réponses en respectant les explications ci-dessus.

Question B.2.2

Compléter le diagramme de déploiement du document réponses en ajoutant les **cardinalités**.

B.3 Lecteur de code-barres

Document à consulter : « **Annexe 2 : Code EAN 13** ».

L'identification de l'opérateur qui travaille à un poste de codage ainsi que la lecture de l'étiquette sur les conteneurs à trier sont faites grâce à une douchette (lecteur de code-barres). Le codage utilisé pour les codes inscrits sur les badges et les étiquettes est du type EAN 13.

Ce code est décrit dans l'annexe 2. Grâce à cette annexe, répondre aux questions suivantes en s'inspirant de l'exemple de l'annexe 2, paragraphe 3.

Question B.3.1

Après application du codage EAN 13, déterminer le code en binaire des trois caractères manquants dans le code-barres ci-dessous.



Les 3 caractères manquants

Figure 10 : Code EAN à compléter

Question B.3.2

À quoi sert la clé de contrôle ?

C Programmation du poste de codage

C.1 Historique des codes postaux affichés à l'écran

Document à consulter : « **Annexe 3** : Chaînes de caractères en C/C++ »

On s'intéresse à l'affichage des 6 derniers codes postaux entrés par l'opérateur de tri. Ces codes postaux sont constitués de 5 caractères compris entre '0' et '9' et ils sont stockés dans un objet de type `string` (voir annexe 3).

Le programme doit d'abord vérifier que le code entré est bien constitué de 5 caractères compris entre 0 et 9.

Question C.1.1

Écrire le code de la fonction `verifierCode` vérifiant que le code entré par l'opérateur de tri est bien constitué de 5 chiffres.

Les 6 derniers codes postaux entrés sont affichés à l'écran du poste de codage.

Pour réaliser cet affichage, les codes sont stockés dans un tableau.

La classe `CCodesAffiches` permet de gérer cette liste contenant les codes affichés.

```
const int NB_CODES = 6 ;
class CCodesAffiches
{
    int nbCodes ;
    string liste[NB_CODES] ;

public :
    CCodesAffiches () ;
    void ajouterCode (const string &code ) ;
    ...
};
```

Question C.1.2

Que signifient le caractère `&` et le mot clé `const` dans l'argument de la méthode `ajouterCode` ?

La méthode `ajouterCode` ajoute un code dans la liste. Si le tableau est plein, les données seront décalées d'une case du tableau afin de placer le code ajouté en dernière position.

Question C.1.3

Donner le corps de la méthode `ajouterCode` sur le document réponses.

C.2 Communication réseau

L'objectif de cette partie est d'étudier l'envoi du code postal entré par l'opérateur sur le poste de codage, au serveur « WCS ». Ce serveur va pouvoir commander le système pour transmettre l'objet postal jusqu'au conteneur de sortie.

Donc le poste de codage après avoir vérifié que le code postal est correct, doit envoyer cette information au serveur « WCS », cette communication se fait par réseau et en particulier grâce à un logiciel client sur le poste de codage.

Le diagramme de classes partiel du client est le suivant :

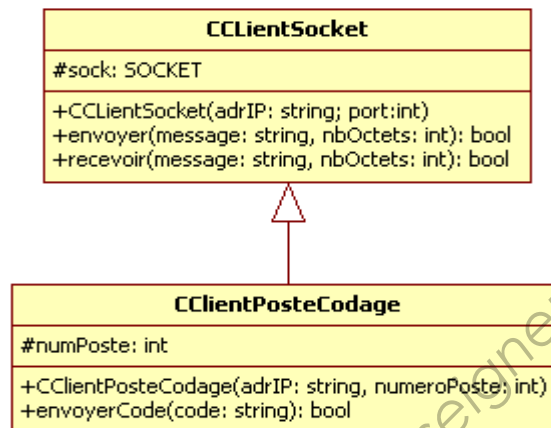


Figure 11 : Diagramme partiel du diagramme de classes

Question C.2.1

Quel est le type de relation entre la classe `CClientSocket` et la classe `CClientPosteCodage` ?

Question C.2.2

Donner les autres types de relation possible dans un diagramme de classes UML.

Un numéro de poste est attribué à chaque poste de codage. Ce numéro est transmis au serveur « WCS » avec le code postal saisi par l'opérateur de tri.

Question C.2.3

Écrire la définition de la classe `CClientPosteCodage` en C++.

Le port pour la classe `CClientPosteCodage` est 2345.

Question C.2.4

Écrire la définition du constructeur de la classe `CClientPosteCodage` en C++.

La méthode `envoyerCode` crée les données à transmettre au serveur « WCS ». Ces données comprennent les informations suivantes :

- 2 caractères ASCII correspondant au numéro du poste de codage ;
- 1 espace ;
- Les 5 caractères du code postal.

Exemple : Poste 4, code postal 01234 -> 04 01234

Question C.2.5

Écrire la définition de la méthode `envoyerCode` en C++.

C.3 Système multitâche

Le logiciel serveur doit pouvoir accepter des connexions de chacun des postes d'encodage simultanément. Pour cela, il doit créer un processus (processus lourd) ou un thread (processus léger) pour chacune des connexions.

Question C.3.1

Quelles sont les différences entre un processus lourd et un processus léger (thread) ?

La société a choisi d'utiliser des processus légers. Voici un extrait du code du programme serveur sans lancement d'un thread pour chaque client connecté.

```
void main (){
    ServeurSocket serveur (PORT);
    char donnees[9];
    int numPoste;
    string code;
    SOCKET sock_client;
    bool fin;
    donnees[8] = 0;
    while (1){
        //! Acceptation d'un client
        if (serveur.acceptClient(sock_client)) {
            //! boucle de traitement
            fin = false;
            do {
                //! - Lecture de la commande
                if (serveur.recevoir (sock_client, donnees, 8) != 8)
                    fin = true;
                else {
                    sscanf (donnees, "%d", &numPoste);
                    code = donnees+3;
                    ...
                }
            }
            while (!fin);
            //! -- Fermeture de la connexion avec le client
            serveur.fermerClient (sock_client);
        }
    }
}
```

Question C.3.2

Entourer, sur le document réponses, la partie du code qui doit être réalisée par le thread.

Il faut transmettre les données reçues par le processus serveur au processus en charge de gérer le trieur (processus s'exécutant sur le même ordinateur).

Les techniques possibles de communication entre processus sont : mutex, tube nommé, sémaphore, file de messages, signal, mémoire partagée.

Question C.3.3

Proposer les techniques de communication parmi celles données, qui permettent de transmettre des données entre deux processus d'un même ordinateur.

D Réseaux industriels

Le convoyeur du système de tri est commandé par un automate industriel du type Siemens S7-400.

Afin de contrôler le fonctionnement du convoyeur, 80 capteurs inductifs tout ou rien sont placés tout au long des 574,5 m de celui-ci. Pour simplifier le câblage, le bus AS-i est utilisé pour relier les 80 capteurs à l'API Siemens S7-400 via les esclaves AS-i et les passerelles Profibus/AS-i.

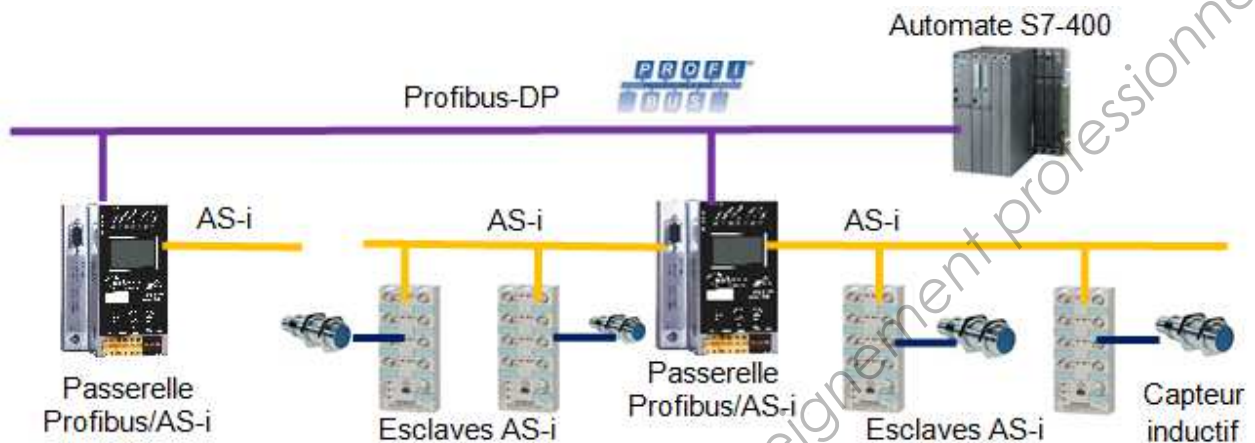


Figure 12 : Schéma de câblage Profibus-DP AS-i

D.1 Étude des réseaux industriels

Document à consulter : « **Annexe 4 : Bus AS-i** ».

Les réseaux Profibus-DP et AS-i sont des bus de terrain qui font partie des réseaux industriels.

Question D.1.1

Quels sont les numéros et les noms du modèle OSI utilisés par les bus de terrain ?

Question D.1.2

Donner la topologie physique des réseaux de terrain Profibus-DP et AS-i.

L'automate S7-400 ne dispose que des interfaces de réseau Ethernet et Profibus-DP. Pour lire l'état des capteurs inductifs reliés au bus AS-i, il faut utiliser une passerelle Profibus-DP/AS-i.

Question D.1.3

Quel est le rôle de cette passerelle ? (Cocher les bonnes réponses)

Question D.1.4

Donner la méthode d'accès au support de transmission du bus AS-i.

D.2 Étude du réseau AS-i

Documents à consulter : « **Annexe 4** : Bus AS-i »,
« **Annexe 5** : Passerelles Profibus-DP/AS-i » et « **Annexe 6** : Esclave AS-i ».

On se propose de déterminer les éléments du bus AS-i.

Question D.2.1

Donner les caractéristiques du bus AS-i en complétant le tableau du document réponses.

La longueur totale du convoyeur du système de tri est de 574,5m.

Question D.2.2

Déterminer le nombre de stations maîtres du bus AS-i à utiliser pour couvrir entièrement le convoyeur.

Question D.2.3

À l'aide de l'annexe 5, choisir une passerelle Profibus-DP/AS-i en donnant sa référence et le nombre de passerelles à utiliser. Vous devez privilégier le choix le plus économique répondant aux besoins du système.

La référence des stations esclaves utilisés est 3RK1200-0CQ20-0AA3.

Ses caractéristiques sont données dans l'annexe 6.

Question D.2.4

Donner le nombre maximal de capteurs et d'actionneurs qu'on peut relier sur chacune de ces stations.

Le convoyeur est constitué de 1149 tapis éjecteurs se déplaçant à la vitesse de 2 m/s.

Pour déterminer la position des tapis éjecteurs, un plot métallique est fixé sous chaque tapis éjecteur et 80 capteurs inductifs sont placés tout au long du convoyeur.

Question D.2.5

Justifier l'utilisation de capteurs inductifs dans ce système de convoyage.

On veut vérifier que le bus AS-i est un bus suffisamment performant pour répondre aux besoins du système de tri.

Question D.2.6

Calculer la durée totale d'une transaction entre une station maître AS-i et une station esclave.

Question D.2.7

Calculer le temps mis par la station maître AS-i pour interroger ses 31 stations esclaves.

Question D.2.8

Pourquoi le réseau AS-i est-il déterministe ?

D.3 Étude d'une transaction AS-i

Document à consulter : « **Annexe 4** : Bus AS-i ».

Une station esclave AS-i vient d'être remplacée par le technicien de maintenance. À la mise sous tension, la station maître AS-i de ce bus a émis la trame de requête suivante :

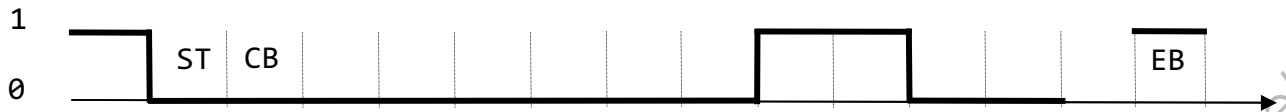


Figure 13 : Trame de requête à compléter

Question D.3.1

Donner l'état logique du bit de contrôle de parité de la trame de requête puis compléter le chronogramme du document réponses.

Question D.3.2

Après avoir relevé la valeur des bits A_0 à A_4 de la trame de requête ci-dessus, donner la signification de cette trame.

Pourquoi cette trame est-elle envoyée à la station qui vient d'être remplacée par le technicien ?

Question D.3.3

Relever la valeur des bits I_0 à I_4 de la trame de requête de la station maître AS-i.

À quoi correspond cette valeur ?

Question D.3.4

Compléter, sur le document réponses, le chronogramme de la trame de réponse de la station esclave concernée par la requête de la station maître AS-i.

Pour transmettre les données sur le support de transmission, le transmetteur AS-i transforme la suite de 1 et 0 en une suite d'impulsions de tension. Cette transformation est réalisée grâce à un traitement particulier comportant un codage de type Manchester et une modulation de type \sin^2 (voir l'annexe 4 section 6).

Question D.3.5

Compléter, dans le document réponses, les chronogrammes de la trame de requête de la station maître AS-i (le bit de contrôle de parité PB, le codage Manchester et les signaux sur le câble AS-i).

E Base de données

Toutes les informations sur le tri sont stockées dans des bases de données, certaines tables sont sauvegardées dans les ordinateurs du système d'information non représentés dans ce document. Elles contiennent toutes les informations sur les tris précédemment effectués. D'autres tables sont mises dans l'ordinateur WCS, elles donnent les informations sur le tri en cours et les compte rendus des tris effectués lors du plan de tri en cours.

E.1 Étude des tables du « WCS »

Document à consulter : « **Annexe 7** : Requêtes SQL ».

Le WCS reçoit les informations sur la répartition des codes postaux et des codes de regroupement selon leur destination (liste des communes). Ces informations sont contenues dans la table `PlanTransport`.

Les responsables d'exploitation créent un plan de tri qui s'applique à une date et heure données et précisent l'emplacement du conteneur de sortie (`NumeroSortie`) pour les codes postaux ou les codes de regroupement correspondants.

Au fur et à mesure des tris, des compte-rendus de tri sont créés dans la table `CompteRenduTri`. Cette table contient toutes les informations sur le tri dont l'opérateur qui l'a effectué. Les informations sur l'ensemble des opérateurs qui agissent sur le système, sont stockées dans la table `Operateurs`.

On obtient donc les tables du WCS suivantes :

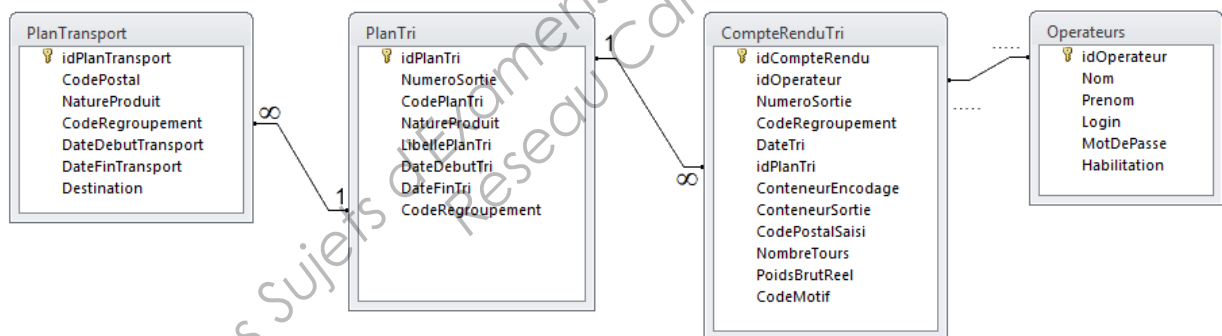


Figure 14 : Extrait du schéma relationnel de la base de données

Question E.1.1

Comment s'appellent les champs précédés d'une icône en forme de clé dans le schéma ci-dessus ?

Quelle est leur particularité ?

Question E.1.2

Compléter le schéma relationnel de la base de données (entre la table `CompteRenduTri` et la table `Operateurs`). Justifier la réponse.

Question E.1.3

Quelle est le type de clés du champ `idOperateur` dans la table `CompteRenduTri` ?

Question E.1.4

Écrire la requête qui donne le nom et le prénom de l'opérateur de tri ainsi que le code qu'il a saisi, pour le compte-rendu dont l'idCompteRendu est 22647.

Les cinq niveaux d'habilitation sont : - responsable qualité - opérateur de tri - technicien de maintenance - responsable d'activités - administrateur.

Le champ idOperateur de la table Operateurs a été créé avec l'attribut AUTO_INCREMENT.

Question E.1.5

Écrire la requête qui ajoute dans la table Operateurs un enregistrement correspondant à l'administrateur Bernard Majeur dont le login est bmajeur et le mot de passe mbjaure.

E.2 Affichage dans une page Web

Documents à consulter : « **Annexe 7** : Requêtes SQL »,
« **Annexe 8** : Tableau en HTML » et « **Annexe 9** : Langage PHP »

Pour visualiser les informations sur les tris, un site Web protégé par un login et mot de passe est proposé. Le login et le mot de passe sont ceux de la table Operateurs.

Seuls les administrateurs ont le droit d'agir sur cette table.

L'objectif de cette partie est d'afficher la liste des opérateurs sous la forme suivante :

Nom	Prénom	Habilitation
Martin	Albert	Administrateur
Dupont	Alfred	Responsable d'équipe
Dumas	Gérard	Codeur
Hugo	Jean	Responsable d'exploitation
Lafont	Jacques	Administrateur
Marie	Anatole	Maintenance

Figure 15 : Résultat affichage des opérateurs

La mise en forme du tableau est réalisée par un fichier CSS déjà existant.

Question E.2.1

Écrire la requête pour récupérer l'ensemble des opérateurs de tri avec les champs Nom, Prénom et Habilitation.

Le résultat de cette requête doit être affiché dans un tableau similaire à celui ci-dessus. On considère que la connexion à la base de données ne peut provoquer d'erreur.

Question E.2.2

Écrire le code PHP pour exécuter cette requête et afficher le tableau dans la page Web, en respectant le format précédent.

F Réseau informatique

F.1 Étude du réseau informatique

Le schéma simplifié du réseau informatique du site est donné ci-dessous.

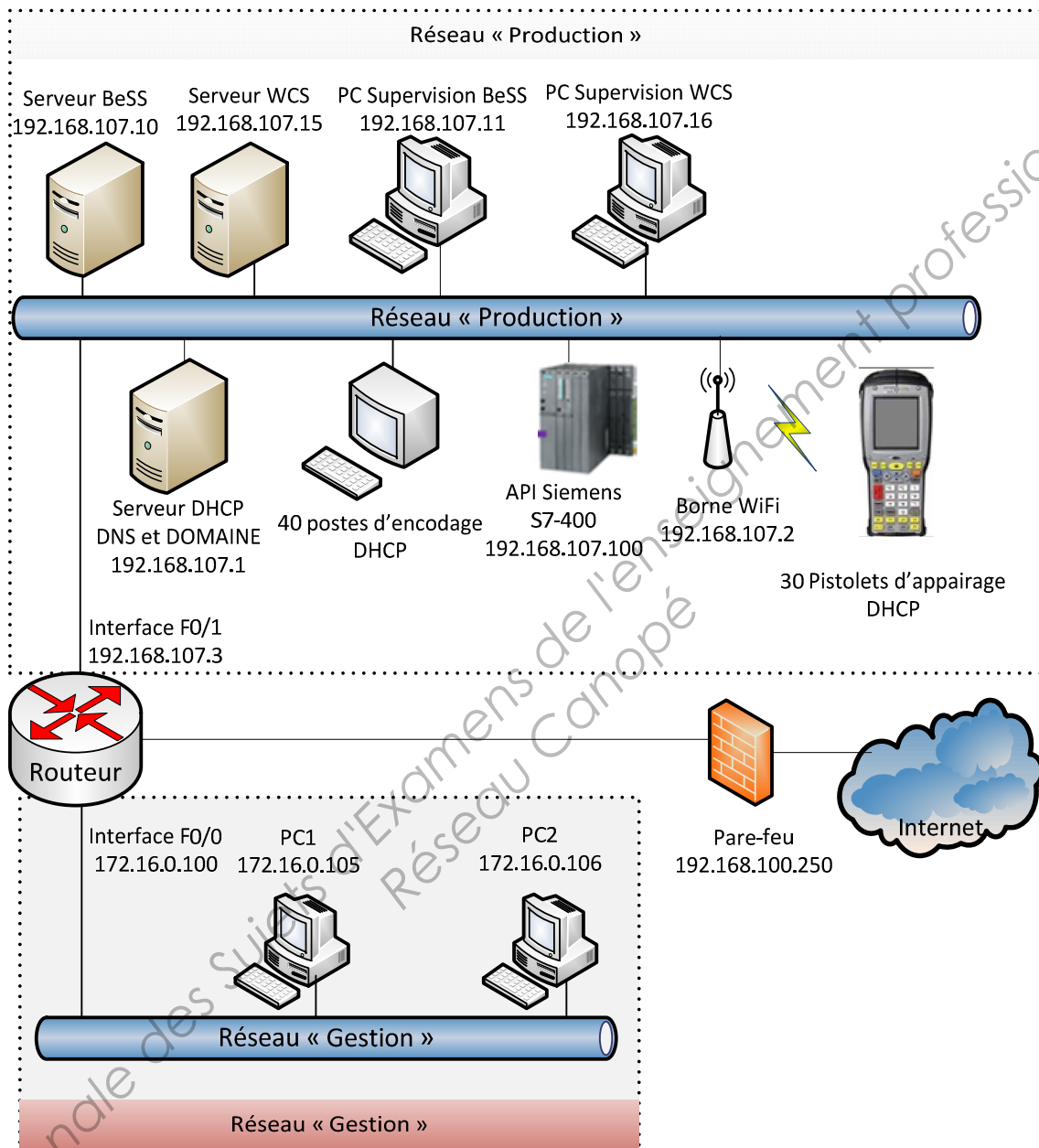


Figure 16 : Schéma architectural du réseau informatique

L'adresse IP du serveur « WCS » du réseau « Production » est 192.168.107.15 et celle du PC1 du réseau « Gestion » est 172.16.0.105

Question F.1.1

Donner les caractéristiques des adresses IP des réseaux « Production » et « Gestion » en complétant le tableau du document réponses.

Pour affecter dynamiquement une adresse IP, l'administrateur du réseau a configuré l'onglet « Général » de la fenêtre « Propriétés de Protocole Internet (TCP/IP) des deux postes de supervision équipés du système d'exploitation XP, » comme ci-dessous :

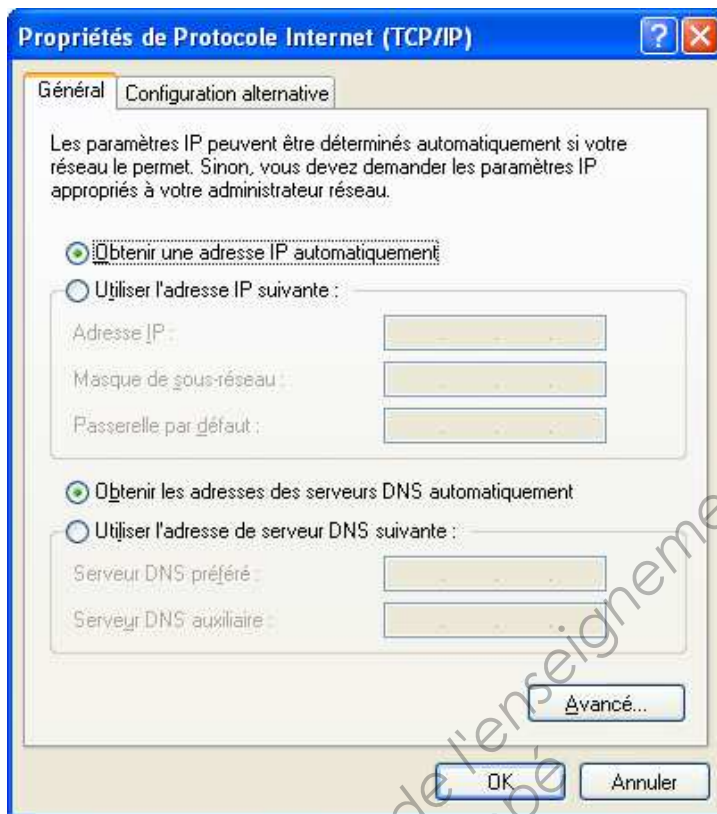


Figure 17 : Affichage de la fenêtre « Propriétés de protocole Internet (TCP/IP) » d'un poste de travail

Question F.1.2

Quel est le service qui permet d'obtenir une adresse IP automatiquement ?
Donner deux autres informations réseaux fournies par ce service.

Question F.1.3

Donner le rôle du serveur « DNS » dans un réseau informatique.

F.2 Analyse des trames d'échange entre un poste de codage et le serveur « WCS »

Document à consulter : « Annexe 10 : Code ASCII ».

Les trames Ethernet échangées entre un poste d'encodage et le serveur « WCS », lors de la première saisie du code postal d'un objet, sont données ci-dessous :

No.	Source	Destination	Proto	Info
1	192.168.107.102	192.168.107.15	TCP	1290 > 50000 [SYN] Seq=0 Win=16384 Len=0 MSS=1460
2	192.168.107.15	192.168.107.102	TCP	50000 > 1290 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0
3	192.168.107.102	192.168.107.15	TCP	1290 > 50000 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	192.168.107.102	192.168.107.15	TCP	1290 > 50000 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=14
5	192.168.107.15	192.168.107.102	TCP	50000 > 1290 [PSH, ACK] Seq=1 Ack=15 Win=65521 Len=1
6	192.168.107.102	192.168.107.15	TCP	1290 > 50000 [ACK] Seq=15 Ack=2 Win=17519 Len=0

À partir des trames capturées, répondre aux questions suivantes.

Question F.2.1

Donner le rôle des trois premières trames capturées (les trames 1, 2 et 3).

Les échanges de données entre le poste d'encodage et le serveur «WCS » commencent par la trame 4 dont le développement est :

```
Frame 4: 68 bytes on wire (544 bits), 68 bytes captured (544 bits)
Ethernet II, Src: Lite-OnT_a0:cb:e3 (00:16:44:a0:cb:e3), Dst: Dell_38:62:ad (00:24:e8:38:62:ad)
IPv4, Src: 192.168.107.102 (192.168.107.102), Dst: 192.168.107.15 (192.168.107.15)
TCP, Src Port: 1290 (1290), Dst Port: 50000 (50000), Seq: 1, Ack: 1, Len: 14
Data (14 bytes)
Data: 4c4330374c493343503036303030
```

Question F.2.2

Donner, en remplissant le tableau du document réponses, les caractéristiques de la trame 4.

Le champ de données (Data) de la trame 4 contient les informations codées en ASCII envoyées par le poste de codage au serveur « WCS ». Ces informations sont composées :

- du numéro de la ligne de collecte d'objet (LCXX) ;
- du numéro de la ligne d'injection (LIX) ;
- du numéro départemental (NDXX) ou du code postal (CPXXXXXX).

Question F.2.3

A l'aide de l'annexe 10, décoder les informations du champ de données de la trame 4.

Sur quelle ligne de collecte et sur quelle ligne d'injection se trouve ce poste de codage ?

Donner la valeur du code postal.

La trame 5 correspond à la réponse de la trame 4. Si le champ de données de la trame 4 est correct, le champ de données de la trame 5 contiendra le caractère 'O', sinon le caractère 'N'.

```
Frame 5: 55 bytes on wire (440 bits), 55 bytes captured (440 bits)
Ethernet II, Src: Dell_38:62:ad (00:24:e8:38:62:ad), Dst: Lite-OnT_a0:cb:e3 (00:16:44:a0:cb:e3)
IPv4, Src: 192.168.107.15 (192.168.107.15), Dst: 192.168.107.102 (192.168.107.102)
TCP, Src Port: 50000 (50000), Dst Port: 1290 (1290), Seq: 1, Ack: 15, Len: 1
Data (1 byte)
Data: 4f
```

Question F.2.4

Donner les deux rôles de cette trame par rapport à la trame 4.

F.3 Communication entre deux réseaux IP.

La société STP souhaite consulter les informations sur le tri stockées dans le serveur « WCS » à partir d'un PC du réseau « Gestion » dont l'adresse IP est : 172.16.1.105/16.

Pour cela, l'administrateur de réseau relie l'interface Ethernet F0/1 du routeur sur le réseau « Production » et celle F0/0 sur le réseau « Gestion ».

Question F.3.1

Donner la fonction de ce routeur. Sur quelle couche (le nom et le numéro) du modèle OSI se situe-t-il ?

Question F.3.2

Citer deux protocoles de routage dynamique.

La fenêtre « Propriétés de Protocole Internet (TCP/IP) » permet d'affecter l'adresse IP ainsi que la passerelle à son poste de travail.

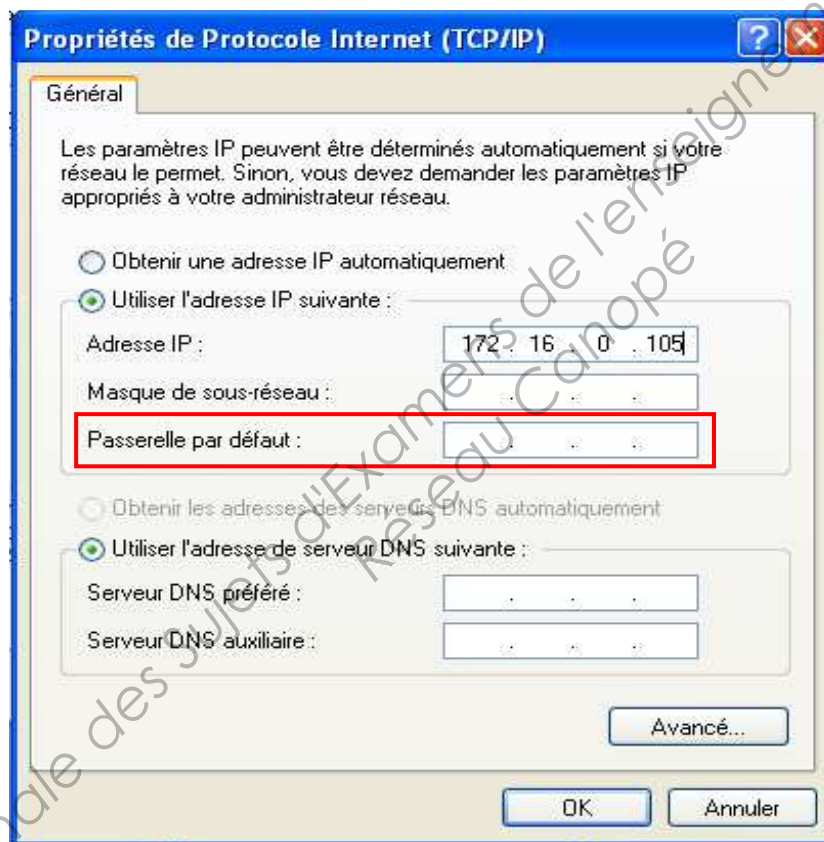


Figure 18 : Affichage de la fenêtre « Propriétés de Protocole Internet (TCP/IP) » du PC1 du réseau « Gestion »

Question F.3.3

Donner l'adresse IP à mettre dans le champ « Passerelle par défaut » pour que le PC1 du réseau « Gestion » puisse accéder au serveur « WCS » du réseau « Production ».

**BREVET DE TECHNICIEN SUPÉRIEUR
INFORMATIQUE ET RÉSEAUX
POUR L'INDUSTRIE ET LES SERVICES TECHNIQUES**

ÉTUDE D'UN SYSTÈME INFORMATISÉ

#

Session 2014

—————
DUREE : 6 HEURES

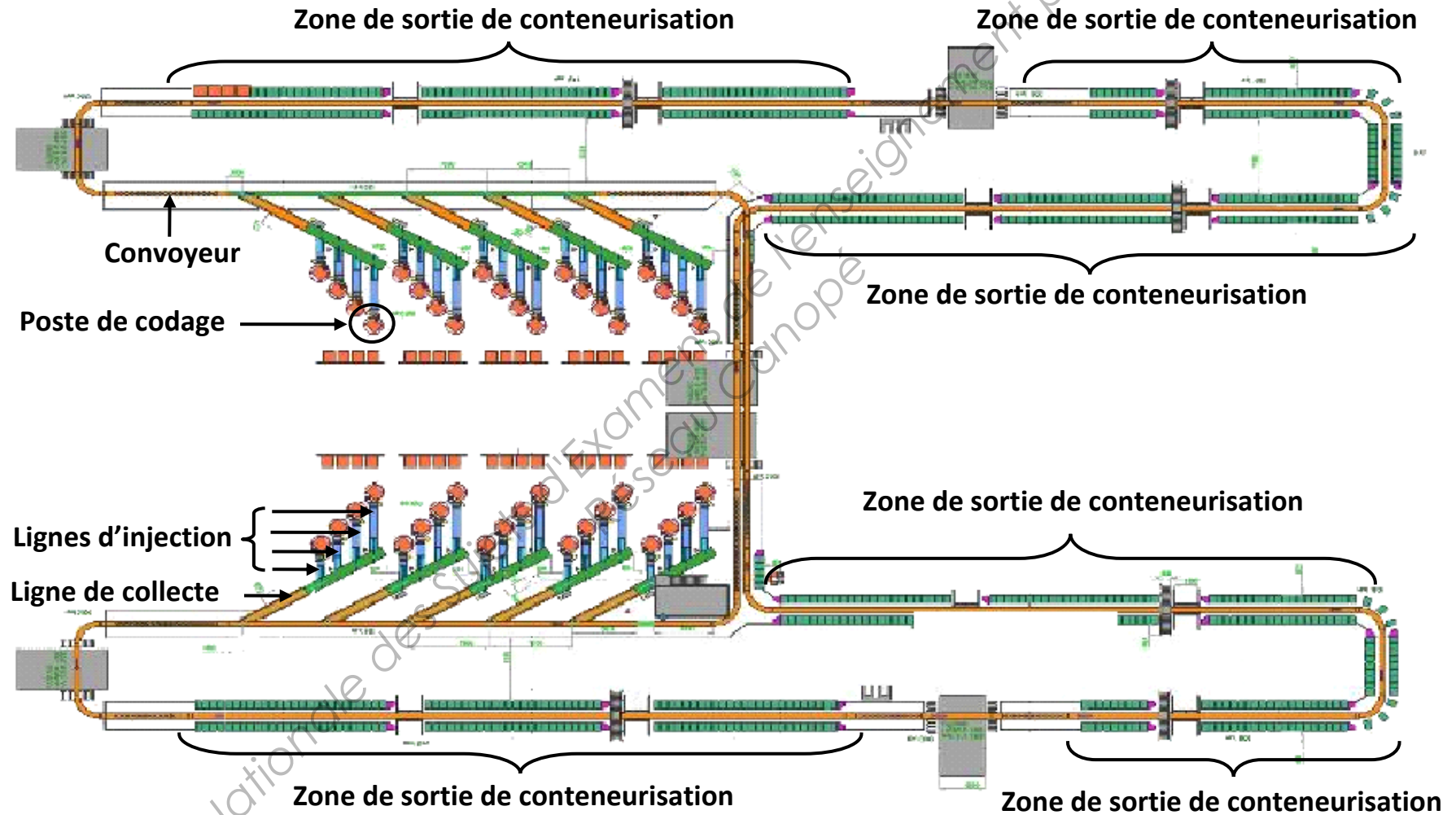
Coefficient 5
—————

ANNEXES

(19 PAGES)

Annexe 1	: Plan du système de tri.....	Page 2
Annexe 2	: Code EAN 13.....	Page 3
Annexe 3	: Chaînes de caractères en C/C++.....	Page 5
Annexe 4	: Bus AS-i.....	Page 6
Annexe 5	: Passerelles Profibus-DP/AS-i.....	Page 10
Annexe 6	: Esclave AS-i.....	Page 11
Annexe 7	: Requêtes SQL.....	Page 12
Annexe 8	: Tableau HTML.....	Page 14
Annexe 9	: Langage PHP.....	Page 15
Annexe 10	: Code ASCII.....	Page 19

Annexe 1 : Plan du « système de tri »



Annexe 2 : Code EAN 13

1 Présentation

Les codes EAN 13 (European Article Numbering à 13 chiffres) sont les codes à barres les plus utilisés dans le monde entier pour l'ensemble de produits de grande consommation. Ils comportent 13 chiffres dont la signification varie suivant le type du produit.

2 Composition d'un code-barres

Le code EAN 13 est constitué d'une suite de 13 caractères. Chaque caractère étant un chiffre compris entre 0 et 9.

Pour être traité par ordinateur (accès aux bases de données...), le code EAN doit être numérisé. Il est alors décomposé en une suite de bits (0 ou 1).

Les bits sont représentés par des barres qui peuvent être lues par des lecteurs optiques. Les 1 sont représentés par des barres noires, les 0 par des barres blanches. Toutes les barres ont la même épaisseur.

Le code barre est constitué (de gauche à droite) par :

- Le premier caractère (chiffre) du préfixe qui n'est pas codé sur les barres.
- Le Début, codé 101 : il permet d'identifier les limites du code-barres, et de donner une référence pour les largeurs des barres.
- Le second caractère du Préfixe + les cinq premiers caractères du Numéro d'Article.
- Le Séparateur Central, codé 01010.
- Les cinq caractères suivants du Numéro d'Article.
- La clé de contrôle.
- Le caractère de Fin, codé 101 (même fonction que Début et Séparateur central)

Chacun des 12 caractères est codé par 7 bits, ce qui fait un total de 95 barres noires ou blanches par article ($3 + 6 \times 7 + 5 + 6 \times 7 + 3 = 95$).



3 Codage d'un caractère

Chaque caractère peut prendre 10 valeurs différentes (entre 0 et 9).

- Les caractères situés à gauche du Séparateur Central du symbole EAN-13 utilisent deux jeux de codification nommés Set A et Set B. Ceux situés à droite utilisent le jeu de codification nommé Set C (table de codage des caractères ci-dessous).
- Le premier caractère du Préfixe (qui n'est pas codé) détermine l'alternance des Set A et B à utiliser pour le codage des 6 caractères situés à gauche du séparateur central (table de parité ci-dessous).

Table des caractères

Caractère	Set A	Set B	Set C
	Partie gauche		Partie droite
0	0001101	0100111	1110010
1	0011001	0110011	1100110
2	0010011	0011011	1101100
3	0111101	0100001	1000010
4	0100011	0011101	1011100
5	0110001	0111001	1001110
6	0101111	0000101	1010000
7	0111011	0010001	1000100
8	0110111	0001001	1001000
9	0001011	0010111	1110100

Table de parité

1° caractère	Succession des jeux
0	AAAAAA
1	AABABB
2	AABBAB
3	AABBBA
4	ABAABB
5	ABBAAB
6	ABBBAA
7	ABABAB
8	ABABBA
9	ABBABA

Exemple : le code du paragraphe précédent

- Premier caractère = 7 donc le caractère suivant est codé avec le Set A, le suivant le Set B, puis le Set A et ainsi de suite ;
- Second caractère = 6 donc en Set A donne 0101111 soit un espace, un trait fin noir, un espace et 4 traits fins noirs accolés ;
- Troisième caractère = 1 soit en Set B 0110011 donc un espace, un double trait noir, un double espace et un double trait noir ;
- Huitième caractère = 6 en Set C car à droite du séparateur, cela donne 1010000 soit un trait noir, un espace, un trait noir et un quadruple espace.

4 Calcul de la clé de contrôle EAN 13

Prenons l'exemple du calcul de la clé de contrôle du code EAN 13 dont les 12 premiers chiffres sont 471-9-5120-0288-x (où x est la clé de contrôle que l'on cherche). Il faut établir le tableau suivant :

Chiffres du code à barres	4	7	1	9	5	1	2	0	0	2	8	8
Poids	1	3	1	3	1	3	1	3	1	3	1	3
Produit	4	21	1	27	5	3	2	0	0	6	8	24

1. Pour les poids, on alterne les valeurs 1 et 3.
2. On multiplie le poids par le chiffre du code à barres
3. On calcule ensuite la somme des résultats,
4. On calcule le reste de la division par 10 de la somme précédemment calculée :
 - Si le reste de la division est égal à 0, alors la clé est 0,
 - Sinon, on ôte à 10, le reste ainsi trouvé : Clé = 10 - Reste.

La somme vaut dans cet exemple $4+21+1+27+5+3+2+0+0+6+8+24 = 101$, le reste de la division par 10 est 1 ; la clé vaut donc $10-1 = 9$.

Le code EAN 13 complet est 4719-5120-0288-9.

Annexe 3 : Chaînes de caractères en C/C++

3-1 Classe "string"

La classe string fait partie de la bibliothèque STL. Elle nécessite l'inclusion du fichier string :

```
#include <string>
```

Les strings sont des chaînes de caractères avec une gestion de la mémoire et des méthodes de gestion intégrées.

Voici quelques méthodes de cette classe :

- string(); // constructeur d'une chaîne vide
- string(const char *s); // constructeur d'une chaîne copie du char *
- const char *c_str() const; // conversion du string vers un char*.
- size_t length() const; // renvoie la longueur de la chaîne
- bool empty() const; // indique si la chaîne est vide (Vrai = vide).
- char &operator[](size_t n); // retourne le caractère à la position n
- char &at(size_t n); // idem

Exemples :

```
string s1 ;
string s2 = "Bonjour " ;
string s3 = "monsieur" ;
cout << s3.length() << endl ; // 8
s1= s2 + s3 ;
cout << s1.c_str() << endl ; // Bonjour monsieur
cout << s2[2] << endl ; // Affiche n
```

3-2 Fonction sprintf

La fonction sprintf copie des données dans une variable de la même façon que printf les affiche à l'écran.

Syntaxe :

```
#include <stdio.h>
int sprintf(char *str, const char *format, ...);
```

Exemple :

Ci-dessous un programme qui contient deux chaînes de caractères : nom et prenom et une variable entière qui contient un age. On va placer dans une autre chaîne de caractères une phrase disant : nom prenom a age ans :

```
#include <stdio.h>
int main()
{
    char chaine[256], prenom[]="Jacques", nom[]="Dupond";
    int age = 30;
    sprintf(chaine, "%s %s a %d ans", prenom, nom, age);
    printf("Le contenu de la variable : %s\n", chaine);
    return 0;
}
```


Annexe 4 : Bus AS-i

Le bus **AS-i** (*Actuator Sensor Interface*) a été développé en 1993 par le consortium AS-i (*Siemens, Schneider, Festo, Eurotherm,...*).

Son objectif est de faciliter la connexion des éléments d'entrées (capteurs) et de sorties (actionneurs) de type tout ou rien (TOR) des systèmes automatisés.



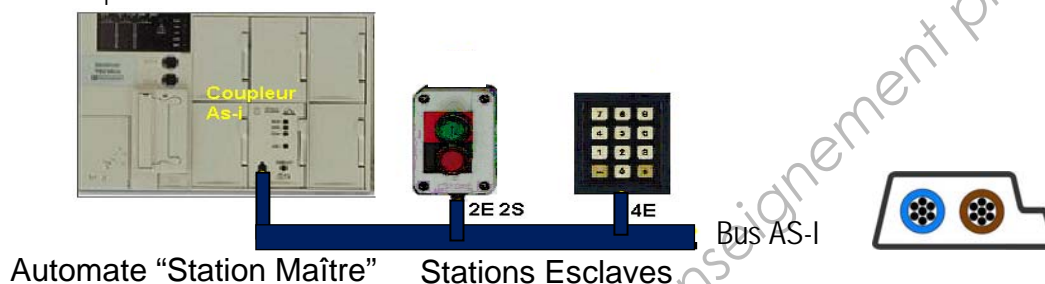
1 Couche physique :

La topologie physique du réseau AS-i est de type «bus». La longueur maximale du câble d'un bus AS-i est de 100m.

Le câble du bus AS-i est composé d'une paire de fils de couleur bleue et marron gainés d'un isolant électrique jaune. Il fournit la tension alimentation $U_{bus} = 30V$ à des stations esclaves tout en véhiculant des données numériques.

Les capteurs et les actionneurs sont reliés au bus via les stations esclaves. Celles-ci se connectent au bus par une prise vampire.

Chaque esclave peut contenir soit 4 entrées soit 4 sorties soit 2 entrées et 2 sorties.



2 Couche liaison de données :

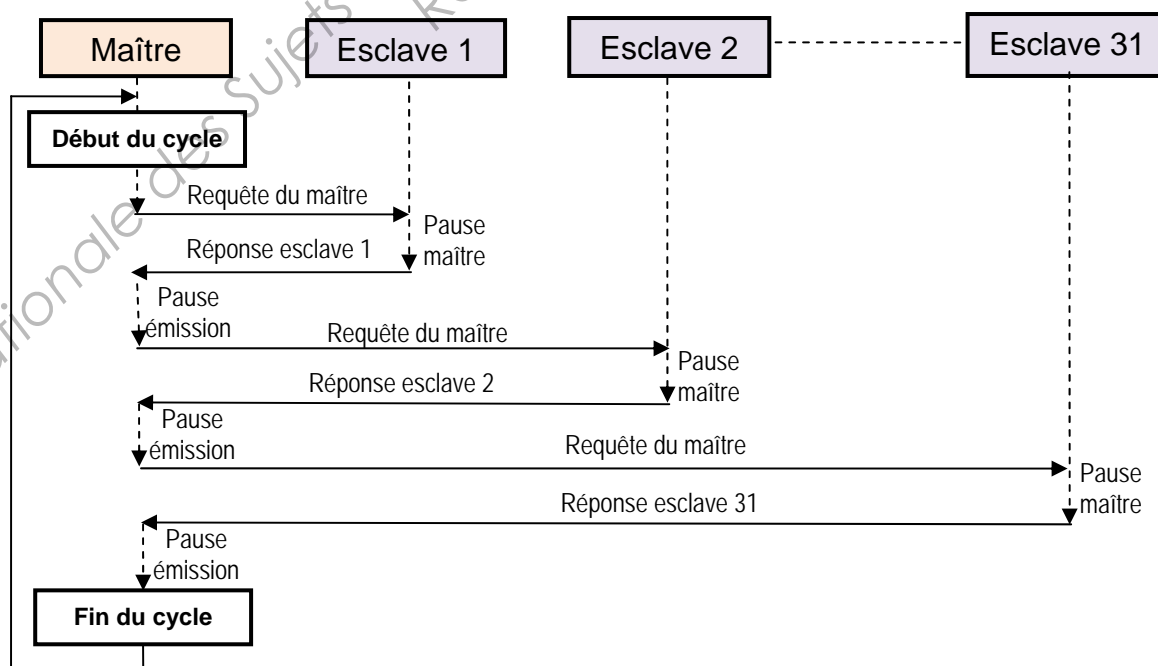
La méthode d'accès du bus AS-i est de type « maître-esclave ». Chaque bus est composé d'un seul maître et de 31 esclaves maximum. Le maître fournit à chaque esclave une adresse unique comprise entre 1 et 31.

Il existe deux types de maître :

- Maître coupleur "automate". Ce type de maître est directement intégré à un automate programmable industriel ;
- Maître "passerelle". Ce type de maître est utilisé pour interfacier le bus AS-i avec un autre bus de terrain.

3 Protocole de communication AS-i :

Le maître interroge cycliquement, l'un après l'autre, chacune des 31 stations esclaves potentielles sur le bus. En un cyclique, le maître met à jour les sorties et fait acquisition de l'état des entrées de l'ensemble des esclaves. Le temps de cycle est garanti.

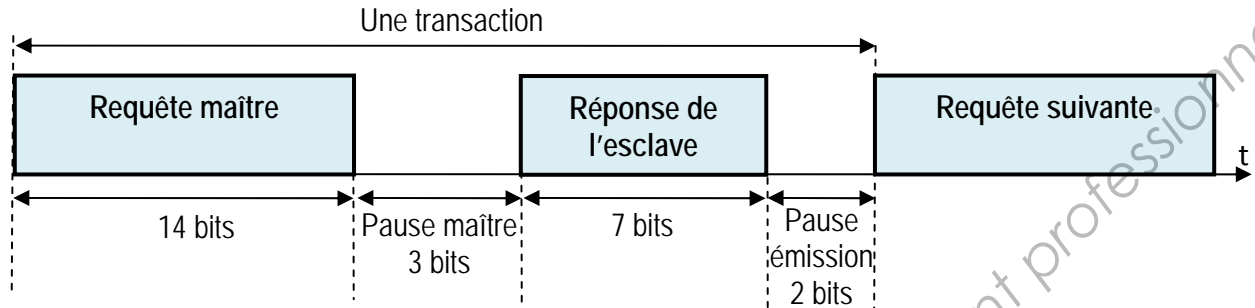


Le maître émet la requête (interroge l'esclave) et attend la réponse de l'esclave interrogé durant le temps dit « pause maître » de trois intervalles bits.

Après avoir reçu une réponse correcte, le maître respecte une pause dite « pause émission » ou « pause esclave » de durée de 2 temps de bits avant d'entamer une nouvelle transaction (interrogation d'une nouvelle esclave).

La durée d'un temps de bit est 6 μ s.

Une transaction est composée de la requête du maître, le temps de « pause maître », la réponse de l'esclave et le temps de « pause émission ».



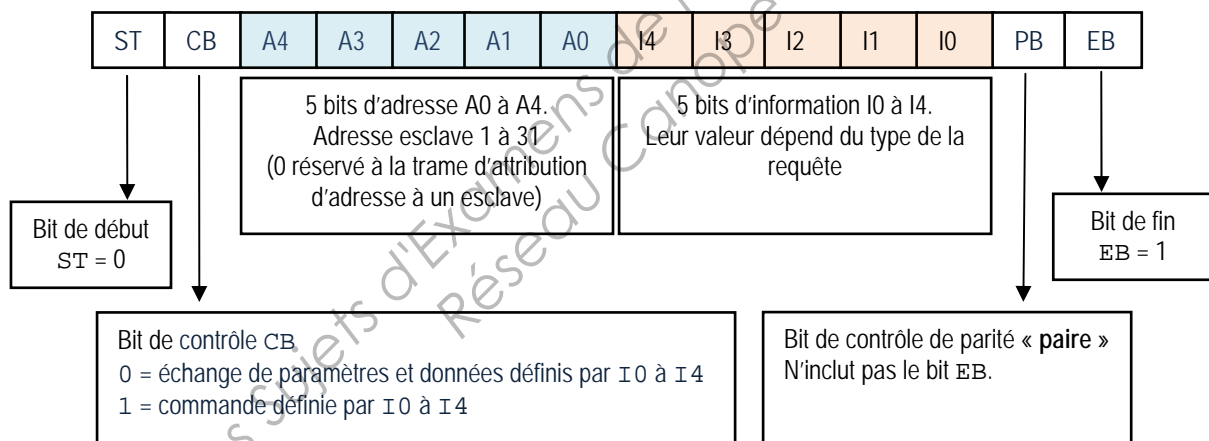
4 Structures des trames AS-i :

Une transaction AS-i est constituée par une trame de requête de la station maître et une trame de réponse de l'esclave.

Les trames émises par le maître et l'esclave commencent par l'émission du « bit de début ST » de niveau logique 0 et se terminent par le bit « fin de bit EB » de niveau logique 1.

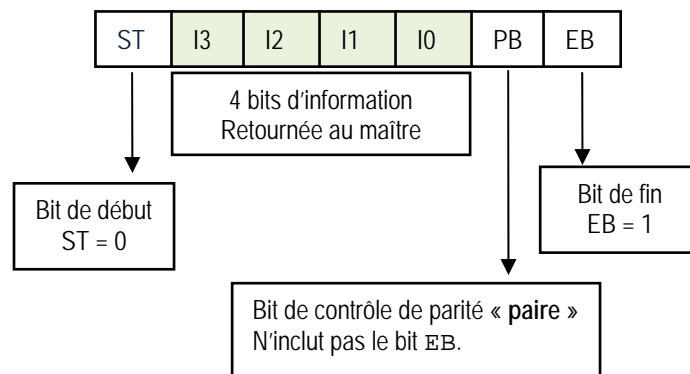
4-1 Structure des trames de requête :

Les trames de requête du maître contiennent toujours 14 bits et sont constituées de la façon suivante :



4-2 Structure des trames de réponse :

Les trames de réponse des stations esclaves contiennent toujours 7 bits et sont constituées de la façon suivante :



5 Trames pour réaliser les transactions :

Les différents types de trames de requête du maître et de réponse de l'esclave sont donnés dans le tableau ci-dessous.

Requêtes du maître	CB	5 bits d'adresse esclave				
Échanger des données	0	A4	A3	A2	A1	A0
Écrire des paramètres	0	A4	A3	A2	A1	A0
Attribuer l'adresse à un esclave	0	0	0	0	0	0
Mettre à zéro l'esclave	1	A4	A3	A2	A1	A0
Réinitialiser l'adresse d'un esclave	1	A4	A3	A2	A1	A0
Lire I/O configuré	1	A4	A3	A2	A1	A0
Lire code ID d'un esclave	1	A4	A3	A2	A1	A0
Lire le statut d'un esclave	1	A4	A3	A2	A1	A0
Lire et remettre à zéro le statut d'un esclave	1	A4	A3	A2	A1	A0

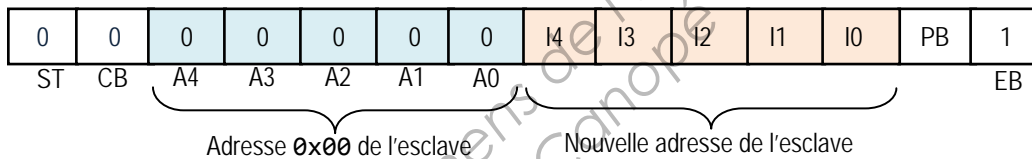
5-1 Trames pour réaliser une transaction de type « attribuer l'adresse à un esclave » :

À la première mise sous tension, l'adresse par défaut d'une station esclave est à 0x00.

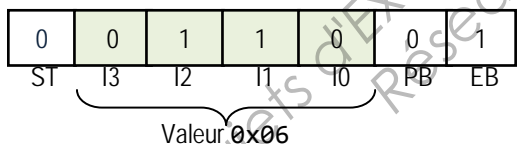
Le maître utilise l'adresse 0x00 pour attribuer une adresse à une station esclave nouvellement installée sur le bus.

Après avoir envoyé la trame d'acquiescement, la station esclave peut recevoir de nouvelles requêtes du maître basées sur sa nouvelle adresse.

Trame de requête du maître :



Trame de réponse de l'esclave (acquiescement) :



C'est la trame d'acquiescement qui a pour valeur 6 quelle que soit l'adresse de l'esclave.

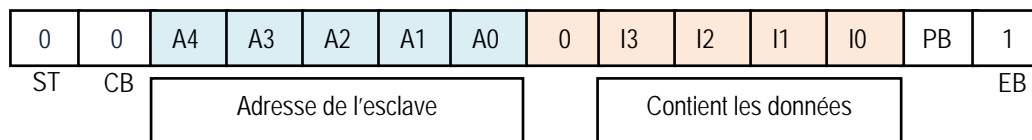
5-2 Trames pour réaliser une transaction de type « échanger des données » :

Le maître utilise deux types de transaction :

- Pour demander l'état des capteurs reliés à un nœud esclave où sont reliés les capteurs tout ou rien. Dans ce cas, les bits I0 à I3 de la trame réponse de l'esclave contiendra l'état des capteurs.
- Pour modifier l'état des actionneurs reliés à un esclave de sortie. Le maître indique l'état des actionneurs à l'aide des bits I0 à I3 de sa trame de requête.

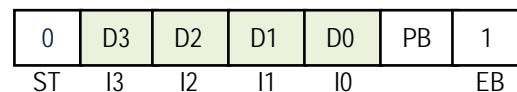
Trame de requête du maître :

La trame de requête pour transférer les données a la structure suivante :



Trame de réponse de l'esclave :

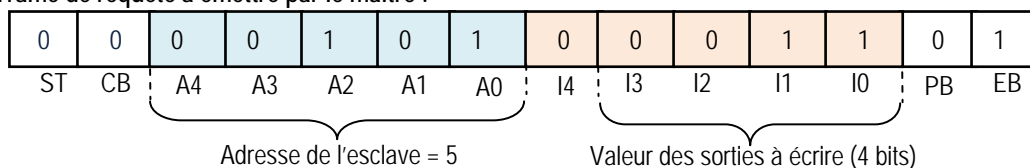
La trame de réponse de l'esclave doit avoir la structure suivante :



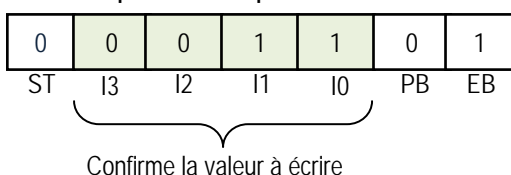
Exemple :

Exemple de requête du maître pour forcer les quatre sorties d'un nœud esclave contenant 4 sorties « Tout Ou Rien » d'adresse 5 :

Trame de requête à émettre par le maître :



Trame de réponse émise par l'esclave 5 :



6 Caractéristiques des signaux électriques véhiculées dans le support physique :

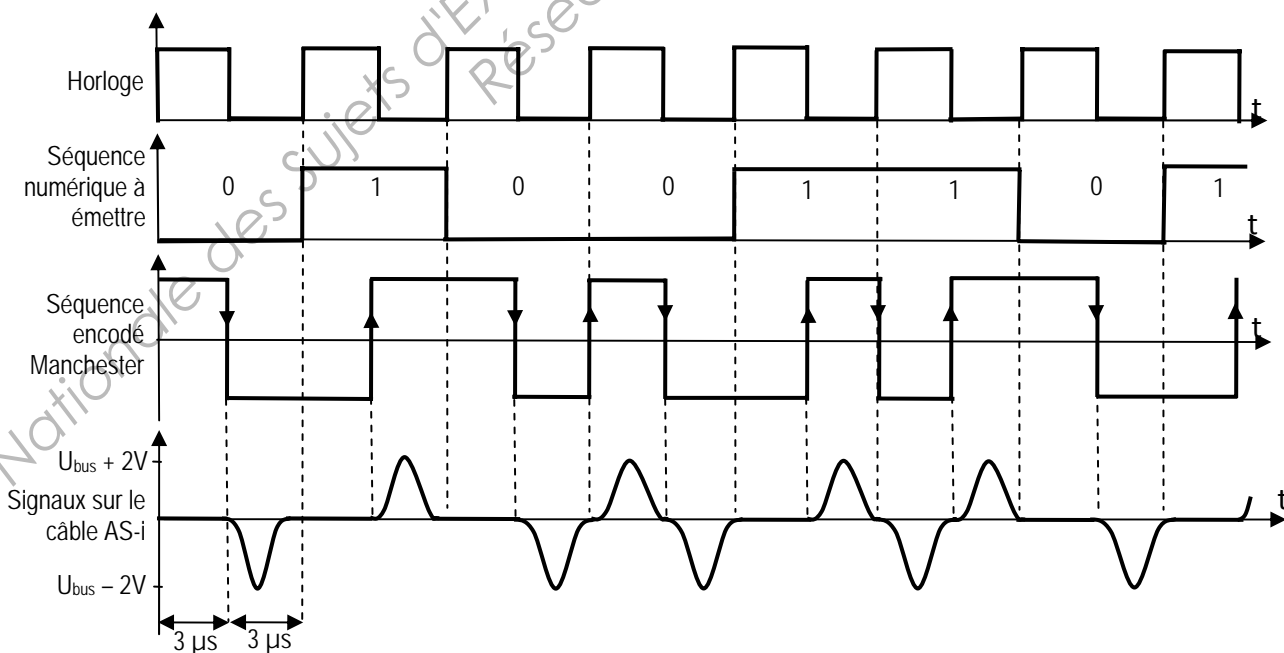
Le bus AS-i établit la communication entre la station maître et les stations esclaves. C'est une communication de données numérique en série et en multipoint.

Les signaux numériques à véhiculer sur le support physique sont codés en code «Manchester» puis modulés par des impulsions alternatives (APM).

- un «0» logique est codé en code Manchester par un intervalle de 3 µs au niveau haut suivi d'un intervalle de 3 µs au niveau bas ;
- un «1» logique est codé en code Manchester par un intervalle de 3 µs au niveau bas suivi d'un intervalle de 3 µs au niveau haut.



Puis les signaux du code Manchester sont transformés en impulsions de tension.

- un front descendant sur le signal du code Manchester se traduit par une impulsion de tension « négative » sur le câble de transmission ;
- un front montant sur le signal du code Manchester se traduit par une impulsion de tension « positive » sur le câble de transmission.



Annexe 5 : Passerelles Profibus-DP/AS-i

Extrait du catalogue Siemens :

Passerelles entre AS-Interface et PROFIBUS ou PROFINET selon spéc. 3.0 AS-i			
DP/AS-i LINK Advanced	Désignation	Caractéristiques techniques	N° de référence
	1 maître AS-i	Passerelle compacte entre réseau PROFIBUS DP et AS-Interface, écran entièrement graphique et touches de commande, interface Web pour un diagnostic, une mise en service et des mises à jour du firmware interactives	6GK1 415-2BA10
	2 maîtres AS-i		6GK1 415-2BA20
IE/AS-i LINK PN IO			
	1 maître AS-i	Passerelle compacte entre réseau PROFINET et AS-Interface, écran entièrement graphique et touches de commande, interface Web pour un diagnostic, une mise en service et des mises à jour du firmware interactives	6GK1 411-2AB10
	2 maîtres AS-i		6GK1 411-2AB20

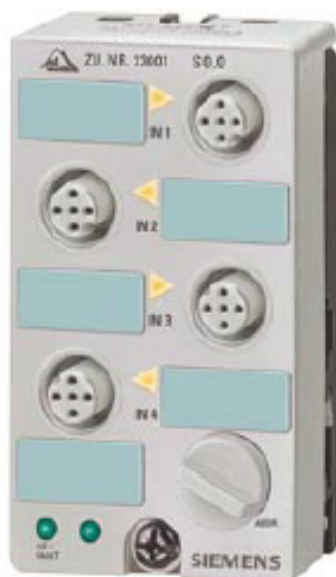
Extrait d'une offre de prix :

Poste	Quantité	code article, référence , désignation	Prix HT en EUR	Unité	Délai
1	1	6GK14152BA10 DP/AS-I Link Advanced mono maître AS-I V3	596.11/	U	10 jours
2	1	6GK14152BA20 DP/AS-I Link Advanced double maître AS-I V3	775.66/	U	10 jours
3	1	6GK14112AB10 IE/ASI Link PNIO, Passerelle PNIO-ASI mono maître	730.51/	U	2 SEM
4	1	6GK14112AB20 IE/ASI Link PNIO, Passerelle PNIO-ASI double maître	955.21/	U	2 SEM

Annexe 6 : Esclave AS-i

Product data sheet

3RK1200-0CQ20-0AA3



AS-INTERFACE COMPACT MOD. K45, IP67,
DIGITAL, 4 INPUTS 4 X 1 INPUT,
200MA MAX., PNP 4 X M12 SOCKET

General technical data:

Design of the product	digital I/O modules for operation in the field, IP67 - K45	
Type	4 inputs	
Design of the slave type	standard slave	
I/O configuration	0	
ID/ID2 code	0/F	
Number / I/O sockets	4	
Design of the electrical connection / of the inputs and outputs	M12 screw-type terminals	
AS interface / total current input / max	mA	270
operating voltage • according to AS-Interface specification	V	26.5 ... 31.6
Ground terminal	Using PIN5 on the M12 sockets. Outgoing via flat tab sleeve (2.8 x 0.8 mm form A)	
Addressing	front addressing socket	
AS interface / Connection / Polarity reversal protection	built-in	
Delivery note	the modules are delivered without mounting plate	
Note 1	All K45 compact modules are delivered with stainless steel screws/sockets	

Annexe 7 : Requêtes « SQL »

S.Q.L. (Structured Query Language) est un langage normalisé de requêtes structurées et un standard d'accès aux bases de données relationnelles.

1 La requête SELECT

Elle permet :

- De sélectionner tous ou certains champs (ou colonnes) d'une ou plusieurs tables en fonction de critères ;
- D'extraire certaines occurrences ou tuples ou enregistrements et de les trier en fonction de critères ;
- D'utiliser des fonctions arithmétiques et de groupements pour des calculs.

Syntaxe générale de la requête SELECT :

```
SELECT [DISTINCT] Liste des champs séparés par une virgule
FROM Liste des tables concernées, séparées par une virgule
WHERE Liste des critères de choix ;
```

L'option DISTINCT permet de ne pas prendre en compte les doublons.

Exemples :

```
SELECT n_dep , nom_dep FROM DEPOSITAIRE ;
Extrait la liste des numéros et des noms des dépositaires de la table DEPOSITAIRE.
SELECT * FROM DEPOSITAIRE ;
Extrait tous les enregistrements et tous les champs de la table DEPOSITAIRE.
SELECT DISTINCT nom_dep FROM DEPOSITAIRE ;
Extrait la liste des noms des dépositaires sans doublons de la table DEPOSITAIRE.
```

La clause WHERE :

La clause WHERE permet de sélectionner dans la table les tuples correspondants aux critères précisés dans cette clause.

Les conditions sont une expression logique pouvant contenir :

- les champs ou colonnes des tables citées dans FROM ;
- les opérateurs de comparaison : >, <, =, >=, <= ;
- les opérateurs NOT, OR, AND ;
- les opérateurs d'ensemble BETWEEN, IS NULL, IS NOT NULL, LIKE, IN.

Exemple :

```
SELECT date FROM commande WHERE (numero > 25 AND numero < 50) ;
Extrait la date des commandes dont le numéro est compris entre 26 et 49.
```

La jointure ou sélection sur plusieurs tables permet d'utiliser les relations entre les tables :

```
SELECT nom_table1.nom_col1 , nom_table2.nom_col1 [, ....]
FROM nom_table1, nom_table2
WHERE nom_table1.nom_col = nom_table2.nom_col ;
```

Pour coupler deux tables ou plus, il faut d'abord préciser les tables concernées dans la clause FROM, ainsi que le ou les critère(s) qui permettront d'associer les lignes des différentes tables dans une clause WHERE pour former un résultat grâce à une clause SELECT.

Exemple :

```
SELECT * FROM LIVRAISON, EDITION
WHERE LIVRAISON.n_edit=EDITION.n_edit AND Lib_edit="La Provence";
Extrait toutes les livraisons de l'édition "La Provence".
```

2 La requête INSERT

INSERT permet d'ajouter un ou plusieurs enregistrements dans une table.

Syntaxe générale de la requête INSERT :

```
INSERT INTO nom_table [ (nom_col1 [, nom_col2,...])]
VALUES (constante1 [, constantes2,...] );
```

Exemple :

```
INSERT INTO DEPOSITAIRE (n_dep , nom_dep , adr_dep)
VALUES (68,'Quentin','Marseille') ;
Insère un nouvel enregistrement dans la table dépositaire avec les valeurs :
N_dep=68 ; nom_dep='Quentin' ; adr_dep='Marseille'
```

3 La requête UPDATE

Elle permet de mettre à jour les données d'un enregistrement.

Syntaxe générale de la requête UPDATE :

```
UPDATE nom_table SET nom_col1= constante1|NULL [, nom_col2= constante2|NULL, ...]
WHERE conditions... ;
```

Exemple :

```
UPDATE DEPOSITAIRE SET adr_dep='Toulon' WHERE n_dep=68 ;
Met à jour l'adresse du dépositaire de numéro 68 avec la valeur 'Toulon'.
```

4 La requête DELETE

Elle permet de supprimer un enregistrement d'une table.

Syntaxe générale de la requête DELETE :

```
DELETE FROM nom_table WHERE conditions... ;
```

Exemples :

```
DELETE FROM LIVRAISON WHERE jour=7 ;
Supprime toutes les livraisons effectuées le jour 7.
DELETE FROM DEPOSITAIRE ;
Supprime tous les dépositaires de la base
```


Annexe 8 : Tableau HTML

Un tableau HTML est encadré par les balises `<table>` et `</table>`. Il comprend :

- Des balises `<tr>` `</tr>` (Table Row) qui permet d'ajouter une ligne dans le tableau.
- Des balises `<th>` `</th>` (Table Header) dans les balises `tr` pour ajouter des cellules d'entête.
- Des balises `<td>` `</td>` (Table Data) dans les balises `tr` pour ajouter des cellules de valeur.

Exemple :

```
<table>
  <tr>
    <th>Évènement</th>
    <th>Date</th>
  </tr>
  <tr>
    <td>Mariage</td>
    <td>22/6/2014</td>
  </tr>
  <tr>
    <td>Anniversaire</td>
    <td>3/7/2014</td>
  </tr>
</table>
```

Résultat :

Évènement	Date
Mariage	22/6/2014
Anniversaire	3/7/2014

Annexe 9 : Langage « PHP »

1 Les bases du langage

Le code source PHP se trouve sur le serveur qui l'interprète et l'exécute. Il ne transmet au navigateur que le code HTML obtenu.

1.1 Structure d'un script PHP

Le script PHP doit avoir comme extension .php et le code doit être compris entre `<?php` et `?>`. Chaque instruction se termine par un point virgule « ; » comme en C et les commentaires sont précédés par « // » ou compris entre « /* » et « */ » comme en C++. Les blocs d'instructions sont délimités par des accolades { }.

Pour afficher une ligne dans la page HTML, il faut utiliser la structure `echo`. Par exemple, l'instruction « `echo "la variable foo est égal à $foo";` » permet d'afficher « la variable `foo` est égal à `bonjour` » si la variable `$foo` contient « `bonjour` ».

1.2 Les variables

Les variables peuvent être du type booléen, entier, chaîne de caractères ... mais elles ne sont pas déclarées et commencent toutes par le caractère \$. Le type est déterminé lors de l'exécution par l'interpréteur PHP. Par exemple, « `$foo="essai" ;` » crée une variable de type chaîne de caractères appelée `$foo`.

Les types PHP possibles sont :

- booléen `TRUE` `FALSE` (i.e. `0`, "", `"0"`) ;
- entier ;
- flottant ;
- chaîne de caractères (entre " ou ') ;
- tableau.

1.3 Les structures conditionnelles

La structure si-sinon s'écrit comme suit :

```
if (condition)
{
    // code si condition vraie;
}
else
{
    // code si condition fausse;
}
```

`else if` permet d'enchaîner une série d'instructions et évite d'avoir à imbriquer des instructions `if`.

La structure switch s'écrit comme suit :

```
switch ($variable)
{
    case valeur_1 :
        // Premier bloc d'instructions...
        break;
    case valeur_2 :
        // Second bloc d'instructions...
        break;
    ...
    case valeur_N :
        // Nième bloc d'instructions...
        break;
    default :
        // bloc d'instructions par défaut...
        break;
}
```

1.4 Les structures itératives

Le langage PHP propose quatre types de boucles :

La boucle for permet d'exécuter une série d'instructions un nombre déterminé de fois :

```
for (initialisation; condition; incrémentation) {
    // bloc d'instructions...
}
```

La boucle while permet d'exécuter une série d'instructions tant que la condition est vérifiée :

```
while (condition) {
    // bloc d'instructions...
}
```

La boucle do while s'écrit comme suit :

```
do {
    // bloc d'instructions...
}
while (condition);
```

La boucle foreach est utilisée exclusivement pour les tableaux ou toutes autres structures tabulaires. Elle s'écrit comme suit :

```
foreach ($tableau as $valeur) {
    // bloc d'instructions...
}
```

1.5 Les tableaux

Le langage PHP dispose de deux types de tableaux : les tableaux indicés et associatifs.

Les tableaux indicés peuvent se créer avec la fonction `array`.

Exemple :

```
$prenoms = array ('Pierre', 'Michel', 'Nicole', 'Véronique',  
'Albert');  
echo $prenom[1] ; // affiche dans la page HTML « Michel »
```

Les tableaux associatifs fonctionnent sur le même principe, sauf qu'au lieu de numéroter les cases, on va les étiqueter en leur donnant à chacune un nom différent.

Exemple :

```
$coordonnees = array (  
'prenom' => 'Pierre',  
'nom' => 'Dupont',  
'adresse' => '25 rue nouvelle',  
'ville' => 'Paris');  
$coordonnees['nom'] = 'Martin'; // le nom est maintenant Martin  
echo $coordonnees['ville']; // affiche dans la page HTML « Paris »
```

2 Accès à la base de données MySQL

L'accès s'effectue en 5 étapes :

1. Se connecter ;
2. Sélectionner une base ;
3. Exécuter des requêtes ;
4. Exploiter les résultats et traiter les erreurs ;
5. Fermer la connexion.

2.1 Connexion à un serveur MySQL

```
resource mysql_connect ( string server , string username ,  
string password , bool new_link , int client_flags ) ;
```

Cette fonction établit une connexion avec le serveur (MySQL), pour un compte utilisateur et un mot de passe secret. Elle renvoie l'identifiant de connexion qui sera utilisé ensuite pour dialoguer avec la base de données en cas de succès et FALSE sinon.

Exemple :

```
$con=mysql_connect("192.168.109.10","user1","gigngtrrl");
```

2.2 Sélection d'une Bdd

```
bool mysql_select_db ( string database_name , resource  
link_identifiant ) ;
```

Cette fonction permet de sélectionner une base de données. Les requêtes s'appliqueront donc à cette base.

Exemple :

```
mysql_select_db("mabase",$con);
```

2.3 Exécution d'une requête

```
resource mysql_query ( string query , resource link_identifieur );
```

Cette fonction adresse une requête SQL au serveur MySQL. Le code doit contenir la requête SQL.

Pour les requêtes du type SELECT , SHOW , DESCRIBE ou EXPLAIN , mysql_query retournera une ressource en cas de succès, ou FALSE en cas d'erreur. Cette ressource sera à utiliser ultérieurement dans les opérations de consultation.

Pour les autres types de requêtes, UPDATE , DELETE , DROP , etc., mysql_query retourne TRUE en cas de succès ou FALSE en cas d'erreur.

2.4 Exploitation des résultats

```
array mysql_fetch_row ( resource result );
```

Cette fonction récupère une des lignes du résultat de la fonction mysql_query et positionne le curseur sur la ligne suivante.

La ligne est représentée sous forme d'un tableau (une liste de valeurs).

```
array mysql_fetch_array ( resource result , int result_type );
```

Cette fonction transforme la ligne courante en un tableau associatif (clé/valeur) à indices numériques ou nommés. L'accès aux données se fait grâce à \$Tableau[Indice] où les indices correspondent aux champs utilisés dans la requête.

Pour récupérer toutes les lignes du résultat, il faut les lire une par une grâce à une des deux fonctions ci-dessus. Le premier appel retournera la première du résultat de la requête, le second appel, la seconde ligne et ainsi de suite.

Exemple :

```
$resultat = mysql_query("SELECT NumAvion,Type,Nom FROM avion");
while ($ligne = mysql_fetch_row($resultat)) {
    echo "NumAvion : $ligne[0]<br>";
    echo "Type : $ligne[1]<br>";
    echo "Nom : $ligne[2]<br>";
}
```

2.5 Fermeture la connexion

```
bool mysql_close([ressource $link_identifieur]);
```

Cette fonction ferme la connexion au serveur MySQL associée à l'identifiant spécifié. Si \$link_identifieur n'est pas spécifié, cette commande s'applique à la dernière connexion ouverte.

Exemple :

```
mysql_close($con);
```

Annexe 10 : Code ASCII

	Hexadécimale.	Caractère	Décimale	Hexadécimale	Caractère
0	0x00	NUL Null	64	0x40	@
1	0x01	SOH Start of Header)	65	0x41	A
2	0x02	STX Start Of Text	66	0x42	B
3	0x03	ETH End Of Text	67	0x43	C
4	0x04	EOT End Of Transmit	68	0x44	D
5	0x05	ENQ Enquiry	69	0x45	E
6	0x06	ACK Acknowledgment	70	0x46	F
7	0x07	BEL Bell	71	0x47	G
8	0x08	BS Backspace)	72	0x48	H
9	0x09	HT Horizontal Tab	73	0x49	I
10	0x0A	LF Line Feed	74	0x4A	J
11	0x0B	VT Vertical Tab	75	0x4B	K
12	0x0C	FF Form Feed	76	0x4C	L
13	0x0D	CR Carriage Return)	77	0x4D	M
14	0x0E	SO Shift Out	78	0x4E	N
15	0x0F	SI Shift In	79	0x4F	O
16	0x10	DLE Data Link Escape	80	0x50	P
17	0x11	DC1 Device Control 1	81	0x51	Q
18	0x12	DC2 Device Control 2	82	0x52	R
19	0x13	DC3 Device Control 3	83	0x53	S
20	0x14	DC4 Device control 4	84	0x54	T
21	0x15	NAK	85	0x55	U
22	0x16	SYN Synchronous Idle	86	0x56	V
23	0x17	ETB End Transm. Block	87	0x57	W
24	0x18	Cancel	88	0x58	X
25	0x19	EM End Of Medium	89	0x59	Y
26	0x1A	SUB Substitute	90	0x5A	Z
27	0x1B	ESC Escape	91	0x5B	[
28	0x1C	FS File Separator	92	0x5C	\
29	0x1D	GS Group Separator	93	0x5D]
30	0x1E	RS Request to Send	94	0x5E	^
31	0x1F	US Unit Separator	95	0x5F	_
32	0x20	SP Space	96	0x60	
33	0x21	!	97	0x61	a
34	0x22	" double quote	98	0x62	b
35	0x23	# number sign	99	0x63	c
36	0x24	\$ dollar sign	100	0x64	d
37	0x25	% percent	101	0x65	e
38	0x26	& ampersand	102	0x66	f
39	0x27	' single quote	103	0x67	g
40	0x28	(104	0x68	h
41	0x29)	105	0x69	i
42	0x2A	* asterisk	106	0x6A	j
43	0x2B	+ plus	107	0x6B	k
44	0x2C	, comma	108	0x6C	l
45	0x2D	- minus	109	0x6D	m
46	0x2E	. dot	110	0x6E	n
47	0x2F	/	111	0x6F	o
48	0x30	0	112	0x70	p
49	0x31	1	113	0x71	q
50	0x32	2	114	0x72	r
51	0x33	3	115	0x73	s
52	0x34	4	116	0x74	t
53	0x35	5	117	0x75	u
54	0x36	6	118	0x76	v
55	0x37	7	119	0x77	w
56	0x38	8	120	0x78	x
57	0x39	9	121	0x79	y
58	0x3A	:	122	0x7A	z
59	0x3B	;	123	0x7B	{
60	0x3C	<	124	0x7C	
61	0x3D	=	125	0x7D	}
62	0x3E	>	126	0x7E	~
63	0x3F	?	127	0x7F	DEL

**BREVET DE TECHNICIEN SUPÉRIEUR
INFORMATIQUE ET RÉSEAUX
POUR L'INDUSTRIE ET LES SERVICES TECHNIQUES**

ÉTUDE D'UN SYSTÈME INFORMATISÉ

SESSION 2014

—————

Durée : 6 heures
Coefficient 5

—————

DOCUMENT RÉPONSES

(21 PAGES)

À RENDRE IMPÉRATIVEMENT À LA FIN DE L'ÉPREUVE,
AGRAFÉ À UNE COPIE MODÈLE EN.

BTS INFORMATIQUE ET RÉSEAUX POUR L'INDUSTRIE ET LES SERVICES TECHNIQUES	Session 2014
ÉTUDE D'UN SYSTÈME INFORMATISÉ – DOCUMENT REPNSES	Code : 14IRSES1
	Page:1/21

B Analyse du système

B.1. Fonctionnement du système

Question B.1.1

Calculer le temps mis par le convoyeur pour effectuer un tour complet.

.....
.....
.....

Question B.1.2

Calculer le nombre de tours effectués par le convoyeur en une heure.

.....
.....
.....

Question B.1.3

Déterminer le nombre théorique d'objets postaux que le système est capable de trier en une heure.

.....
.....
.....

Question B.1.4

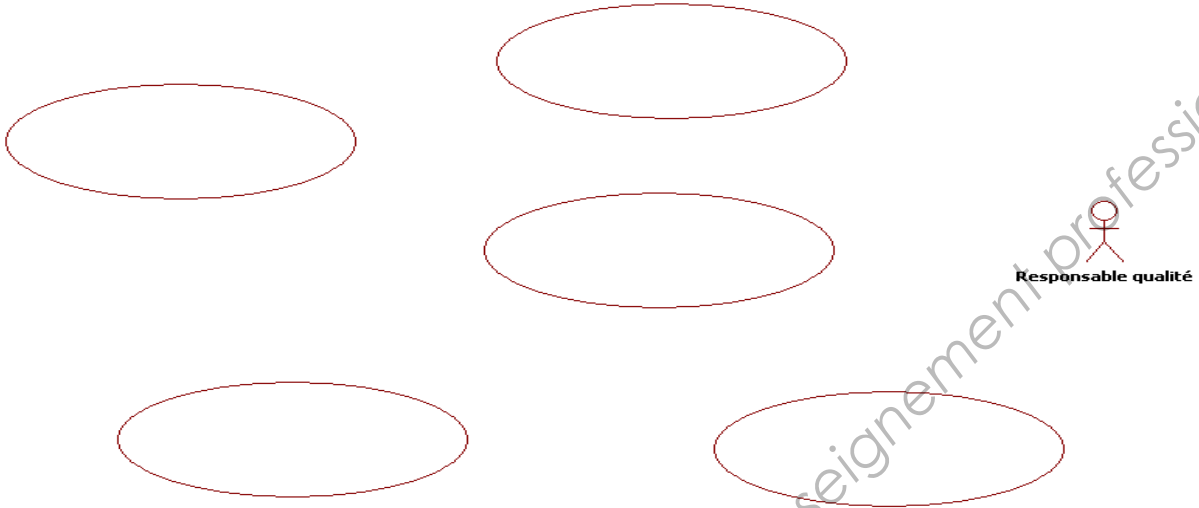
Calculer le temps maximal en minutes et en secondes nécessaire à l'opérateur de tri pour entrer 100 objets postaux dans le convoyeur.

.....
.....
.....
.....
.....
.....

B.2. Étude UML

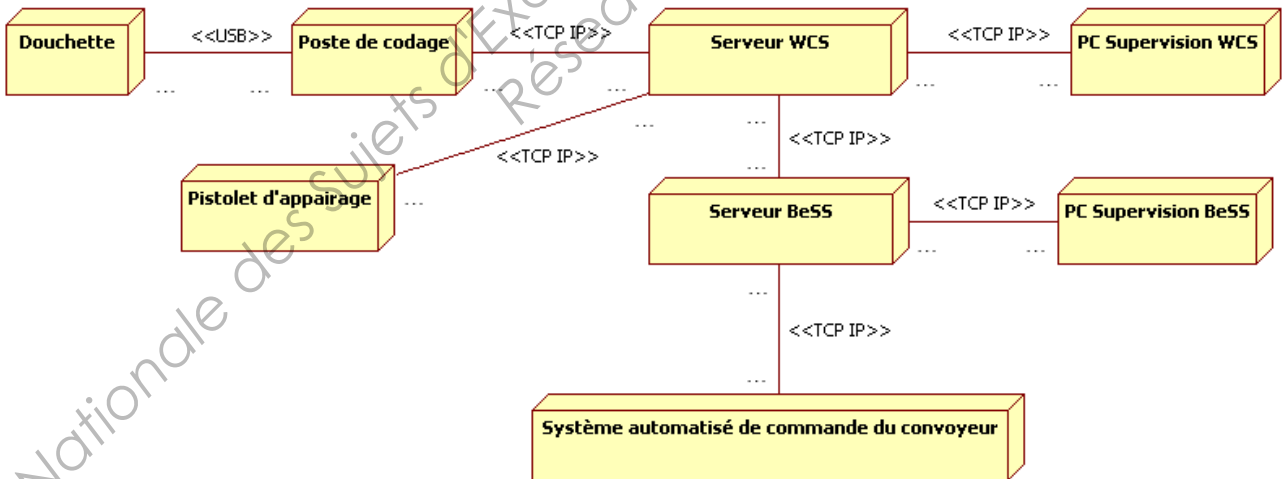
Question B.2.1

Compléter la partie du diagramme de cas d'utilisation ci-dessous en respectant les explications données dans le sujet.



Question B.2.2

Compléter le diagramme de déploiement ci-dessous en ajoutant les **cardinalités**.



B.3. Lecteur de code-barres

Question B.3.1

Après application du codage EAN 13, déterminer le code en binaire des trois caractères manquants dans le code-barres.

Succession des jeux :

Code du premier caractère (7) :

Code du second caractère (1) :

Code du septième caractère (4) :

Question B.3.2

À quoi sert la clé de contrôle ?

.....
.....
.....

Base Nationale des Sujets d'Examens de l'enseignement professionnel
Réseau Canopé

C Programmation du poste de codage

C.1. Historique des codes postaux affichés à l'écran

Question C.1.1

Écrire le code de la fonction `verifierCode` vérifiant que le code entré par l'opérateur de tri est bien constitué de 5 chiffres.

```
bool verifierCode (string code)
```

```
{
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

Question C.1.2

Que signifient le caractère `&` et le mot clé `const` dans l'argument de la méthode `ajouterCode` ?

```
.....
```

```
.....
```

```
.....
```


Question C.2.3

Écrire la définition de la classe `CClientPosteCodage` en C++.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Question C.2.4

Écrire la définition du constructeur de la classe `CClientPosteCodage` en C++.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Question C.2.5

Écrire la définition de la méthode `envoyerCode` en C++.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Question C.3.2

Entourer la partie du code ci-dessous qui doit être réalisée par le thread.

```
void main ()
{
    ServeurSocket serveur (PORT);
    char donnees[9];
    int numPoste;
    string code;
    SOCKET sock_client;
    bool fin;
    donnees[8] = 0;
    while (1)
    {
        //! Acceptation d'un client
        if (serveur.acceptClient(sock_client))
        {
            //! boucle de traitement
            fin = false;
            do
            {
                //! - Lecture de la commande
                if (serveur.recevoir (sock_client, donnees, 8) != 8)
                    fin = true;
                else
                {
                    sscanf (donnees, "%d ", &numPoste);
                    code = donnees+3;
                    ...
                }
            }
            while (!fin);
            //! -- Fermeture de la connexion avec le client
            serveur.fermerClient (sock_client);
        }
    }
}
```

Question C.3.3

Proposer les techniques de communication parmi celles données, qui permettent de transmettre des données entre deux processus d'un même ordinateur.

Mettre une croix dans la case correspondant au bon choix.

	Permet	Ne permet pas
Mutex		
Tube nommé		
Sémaphore		
File de messages		
Signal		
Mémoire partagée		

Base Nationale des Sujets d'Examens de l'enseignement professionnel
Réseau Canopé

D Réseaux industriels

D.1. Étude des réseaux industriels

Question D.1.1

Quels sont les numéros et les noms du modèle OSI utilisés par les bus de terrain ?

Les numéros du modèles OSI utilisés par les bus de terrain

.....

Les noms du modèles OSI utilisés par les bus de terrain

.....

Question D.1.2

Donner la topologie physique des réseaux de terrain Profibus-DP et AS-i.

.....

.....

Question D.1.3

Quel est le rôle de cette passerelle ? (Cocher les bonnes réponses)

- Transmet toutes les trames sur ces deux réseaux
- Permet de relier 2 segments d'un réseau utilisant des médiums différents
- Permet de relier 2 réseaux de protocoles différents
- Permet de diminuer le nombre de collisions sur ce réseau
- Filtre les trames en fonction des adresses MAC

Question D.1.4

Donner la méthode d'accès au support de transmission du bus AS-i.

.....

.....

.....

D.2. Étude du réseau AS-i

Question D.2.1

Donner les caractéristiques du bus AS-i en complétant le tableau ci-dessous.

La longueur maximale du bus AS-i	
Le nombre d'entrées/sorties sur une station d'esclave	
Nombre maximum d'esclaves sur un bus	
Nombre de bits de la trame de requête	
Nombre de bits de la trame de réponse	
Nombre de bits de « pause émission »	
Nombre de bits de « pause maître »	
Le nombre total de bits pour effectuer une transaction	

Question D.2.2

Déterminer le nombre de stations maîtres du bus AS-i à utiliser pour couvrir entièrement le convoyeur.

.....

.....

.....

Question D.2.3

À l'aide de l'annexe 5, choisir une passerelle Profibus-DP/AS-i en donnant sa référence et le nombre de passerelles à utiliser. Vous devez privilégier le choix le plus économique répondant aux besoins du système.

Référence de la passerelle	Nombre

Question D.2.4

Donner le nombre maximal de capteurs et d'actionneurs qu'on peut relier sur chacune de ces stations.

Nombre de capteurs :

Nombre d'actionneurs :

Question D.2.5

Justifier l'utilisation de capteurs inductifs dans ce système de convoyage.

.....
.....
.....
.....
.....

Question D.2.6

Calculer la durée totale d'une transaction entre une station maître AS-i et une station esclave.

.....
.....
.....
.....

Question D.2.7

Calculer le temps mis par la station maître AS-i pour interroger ses 31 stations esclaves.

.....
.....
.....

Question D.2.8

Pourquoi le réseau AS-i est-il déterministe ?

.....
.....
.....

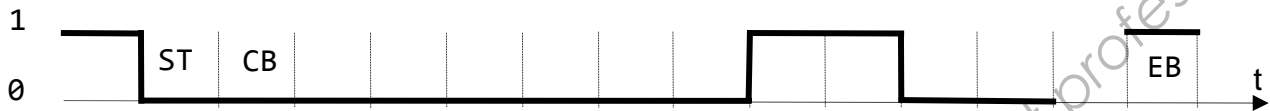
D.3. Étude d'une transaction AS-i

Question D.3.1

Donner l'état logique du bit de contrôle de parité de la trame de requête puis compléter le chronogramme ci-dessous.

.....
.....

Compléter ce chronogramme.



Question D.3.2

Après avoir relevé la valeur des bits A_0 à A_4 de la trame de requête ci-dessus, donner la signification de cette trame.

Pourquoi cette trame est-elle envoyée à la station qui vient d'être remplacée par le technicien ?

.....
.....
.....
.....

Question D.3.3

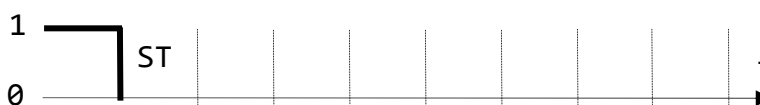
Relever la valeur des bits I_0 à I_4 de la trame de requête de la station maître AS-i.

À quoi correspond cette valeur ?

.....
.....
.....

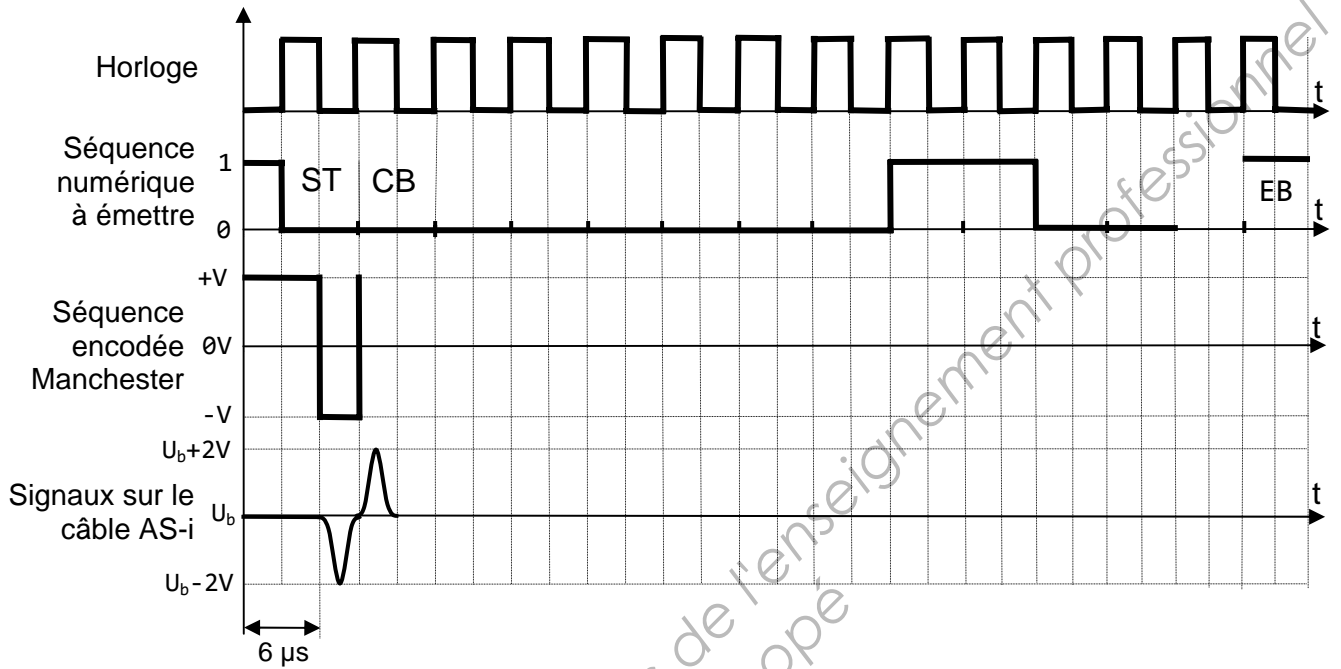
Question D.3.4

Compléter le chronogramme de la trame de réponse de la station esclave concernée par la requête de la station maître AS-i.



Question D.3.5

Compléter les chronogrammes de la trame de requête de la station maître ci-dessous (le bit de contrôle de parité, le codage Manchester et les signaux sur le câble AS-i).



E Base de données

E.1. Étude des tables du serveur « WCS »

Question E.1.1

Comment s'appellent les champs précédés d'une icône en forme de clé dans le schéma ci-dessous ?

Quelle est leur particularité ?

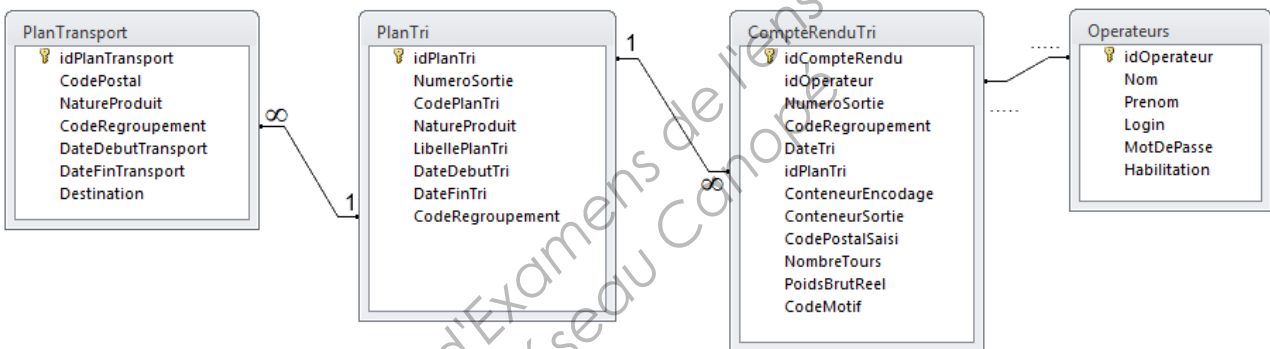
.....

.....

.....

Question E.1.2

Compléter le schéma relationnel de la base de données (entre la table `CompteRenduTri` et la table `Operateurs`). Justifier la réponse.



.....

.....

.....

Question E.1.3

Quelle est le type de clés du champ `idOperateur` dans la table `CompteRenduTri` ?

.....

.....

.....

Question E.1.4

Écrire la requête qui donne le nom et le prénom de l'opérateur de tri ainsi que le code qu'il a saisi, pour le compte-rendu dont l'idCompteRendu est 22647.

.....

.....

.....

.....

.....

Question E.1.5

Écrire la requête qui ajoute dans la table `Operateurs` un enregistrement correspondant à l'administrateur Bernard Majeur dont le login est `bmajeur` et le mot de passe `mbjaure`.

.....

.....

.....

.....

E.2. Affichage dans une page Web

Question E.2.1

Écrire la requête pour récupérer l'ensemble des opérateurs de tri avec les champs `Nom`, `Prenom` et `Habilitation`.

.....

.....

.....

Question E.2.2

Écrire le code PHP pour exécuter cette requête et afficher le tableau dans la page Web, en respectant le format précédent.

Début du code PHP

.....

Connexion à la base de données STP sur le PC local avec le login `rootstp` et mot de passe `pwdstp`

.....

.....

.....

.....

.....

.....

Exécution de la requête avec récupération du résultat dans `$listeOperateurs`

.....

.....

.....

Affichage de la première ligne du tableau

.....

.....

Affichage de l'ensemble du résultat dans les lignes du tableau

.....

.....

.....

.....

.....

Fin du tableau

.....

Fin du code PHP

.....

F Étude du réseau informatique

F.1. Étude du réseau informatique

Question F.1.1

Donner les caractéristiques des adresses IP des réseaux « Production » et « Gestion » en complétant le tableau ci-dessous.

	Réseau « Production »	Réseau « Gestion »
La classe de cette adresse		
Son masque de sous réseau		
Son adresse de réseau		
L'adresse IP du premier hôte		
L'adresse IP du dernier hôte		
Le nombre d'adresses hôte		
L'adresse de diffusion		

Question F.1.2

Quel est le service qui permet d'obtenir une adresse IP automatiquement ?
Donner deux autres informations réseaux fournies par ce service.

.....

.....

.....

.....

.....

.....

.....

Question F.1.3

Donner le rôle du serveur « DNS » dans un réseau informatique.

.....

.....

.....

.....

F.2. Analyse des trames d'échange entre un poste de codage et le serveur « WCS ».

Question F.2.1

Donner le rôle des trois premières trames capturées (les trames 1, 2 et 3).

.....

.....

.....

Question F.2.2

Donner, en remplissant le tableau ci-dessous, les caractéristiques de la trame 4.

Adresse MAC de l'émetteur	
Adresse MAC du destinataire	
Adresse IP de l'émetteur	
Port source	
Port destination	
Information du champ de données	

Question F.2.3

A l'aide de l'annexe 10, décoder les informations du champ de données de la trame 4.
 Sur quelle ligne de collecte et sur quelle ligne d'injection se trouve ce poste de codage ?
 Donner la valeur du code postal.

Code en ASCII :

Ligne de collecte :

Ligne d'injection :

Valeur du code postal :

Question F.2.4

Donner les deux rôles de cette trame par rapport à la trame 4.

.....

.....

.....

.....

F.3. Communication entre deux réseaux IP

Question F.3.1

Donner la fonction de ce routeur. Sur quelle couche (le nom et le numéro) du modèle OSI se situe-t-il ?

.....

.....

.....

.....

Question F.3.2

Citer deux protocoles de routage dynamique.

.....

.....

.....

Question F.3.3

Donner l'adresse IP à mettre dans le champ « Passerelle par défaut » pour que le PC1 du réseau « Gestion » puisse accéder au serveur « WCS » du réseau « Production ».

.....

.....

Base Nationale des Sujets d'Examens de l'enseignement professionnel
Réseau Canopé