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

## E4 - ÉTUDE D'UN SYSTÈME INFORMATISÉ U4

Session 2015

---

Durée : 6 heures
Coefficient 5

---

## SUJET

(16 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 CONTEXTE | 30 mn | 15 points |
| **C.** | CONCEPTION et CODAGE | 75 mn | 30 points |
| **D.** | ORGANISATION DES DONNÉES | 60 mn | 15 points |
| **E.** | COMMUNICATION | 75 mn | 20 points |
| **F.** | RESEAUX | 60 mn | 20 points |
| | Relecture | 30 mn | |

**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.

# Étude d'une chaine de laminage à froid

## A. PRÉSENTATION DU SYSTÈME

L'acier est issu d'un mélange de minerai de fer et de coke chauffé dans des hauts fourneaux. Après décapage, il est souvent fourni sous forme de bobines appelées coils.

Le processus de fabrication de ces coils est montré figure 1.

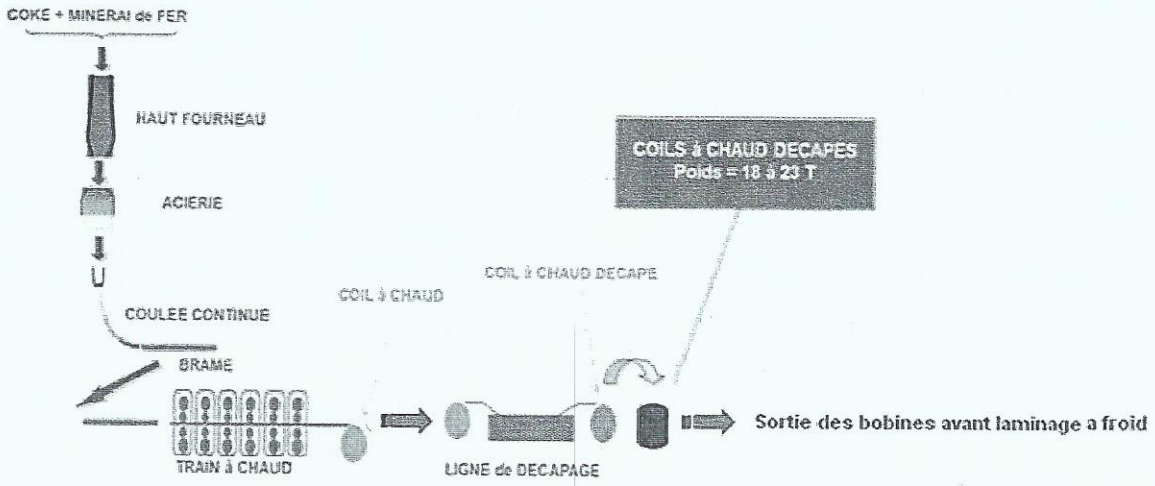



**Figure 1 : Production des bobines d'acier, avant laminage.**

Le **laminage** est un procédé de fabrication par déformation plastique. Il concerne différents matériaux comme le métal ou tout autre matériau sous forme pâteuse tels que le papier ou les pâtes alimentaires. Cette déformation est obtenue par compression continue au passage entre deux cylindres contrarotatifs appelés laminoir.

Un laminoir est une installation industrielle (figure 2) ayant pour but la réduction d'épaisseur d'un matériau (généralement du métal). Il permet également la production de barres profilées (produits longs).

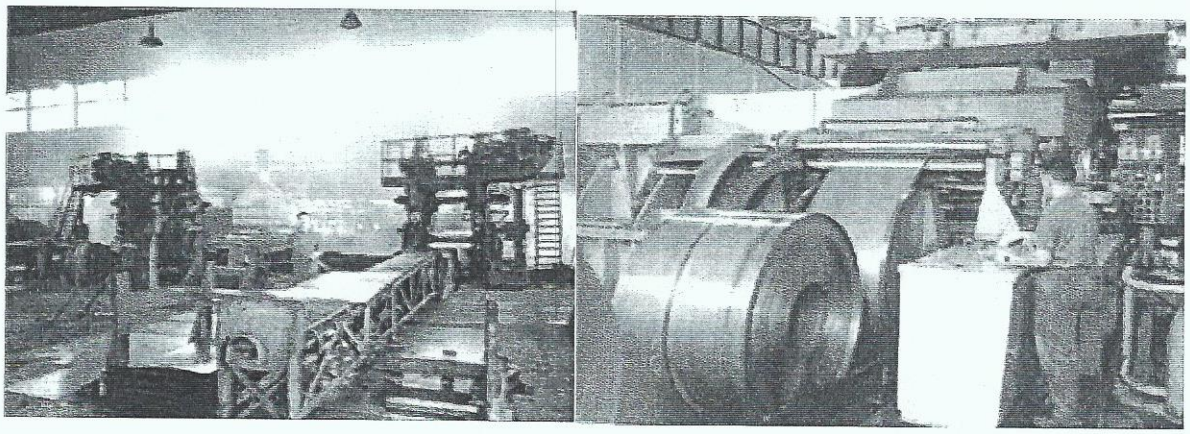


**Figure 2 : Laminage à froid.**

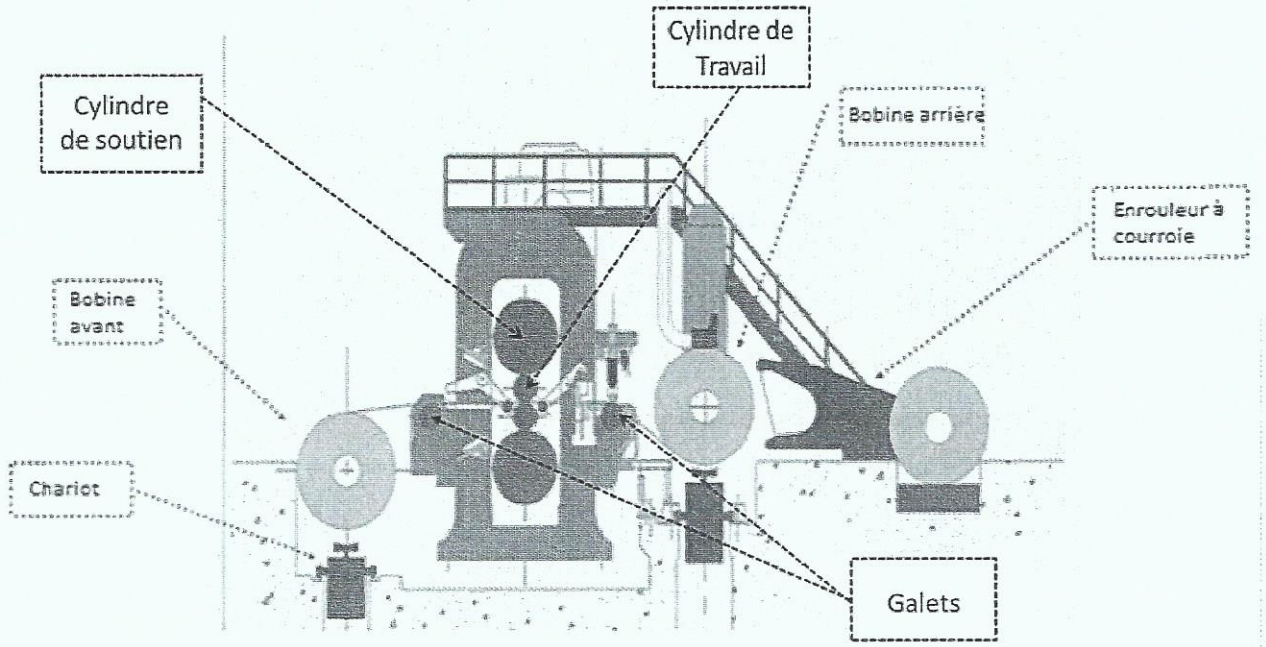



**Figure 3 : Cage de laminage (QUARTO)**

Une cage de laminoir (Figure 3) est composée :

- d'un jeu de cylindres de laminage ;
- d'une paire de cylindres dits « de travail » entre lesquels le matériau est allongé. Les cylindres de travail sont en fonte rectifiée ;
- d'une autre paire de cylindres dits « de soutien » (un cylindre de chaque côté de la paire de cylindres de travail) pour réduire la déformation des cylindres de travail. Les cylindres de soutien sont en acier ;
- de deux anneaux métalliques maintenant ensemble les cylindres (un anneau de chaque côté).

En sortie de laminoir, la bande peut être soumise à un arrosage d'eau qui lui confère des propriétés proches de celles de la trempe. En fin de ligne le métal est enroulé sur des bobineuses. En sidérurgie, le **coil** (mot anglais) est une bobine de tôle issue soit du laminage à chaud, soit de la transformation à froid. (Figure 5, une bobine prête à être transférée)



**Figure 5 : Coil (bobine) en sortie du laminoir, non cisaillée, non refendue**

C'est un produit fini pour la sidérurgie, mais un demi-produit pour la transformation.

Dans le cas des tôles, pour améliorer l'employabilité du produit, celles-ci sont livrées en bobines (ou coils) d'un poids moyen de 12 tonnes mais pouvant dépasser les 25 tonnes et généralement d'une largeur de 1 250 à 2 000 mm.

Les bobines sont ensuite acheminées vers des transformateurs et destinées soit :

- au découpage en feuilles (cisaillage voir figures 8 et 9),
- au refendage (découpe dans le sens de la longueur et reformation de bobines de plus faible largeur),
- à la fabrication de profilés reconstitués (poutrelles, cornières, tés, tubes etc.)
- à l'emboutissage.

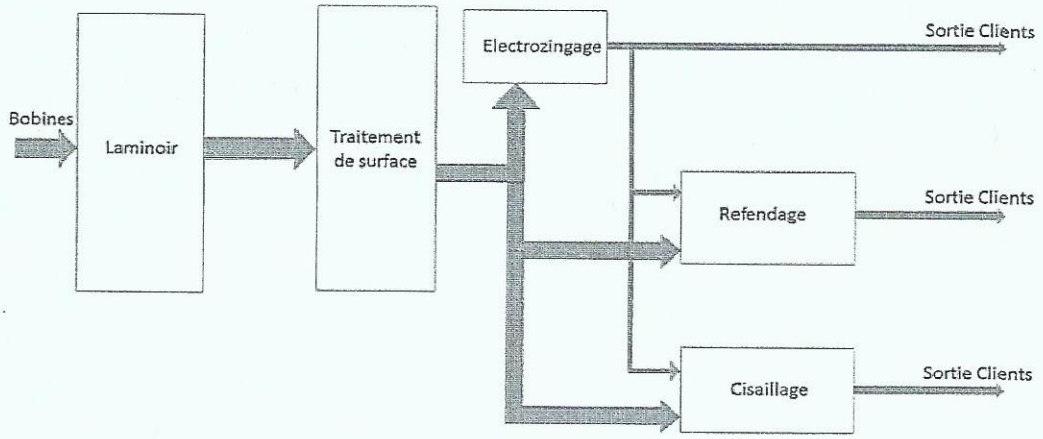



**Figure 4 : acheminement des bobines vers des postes de transformation.**

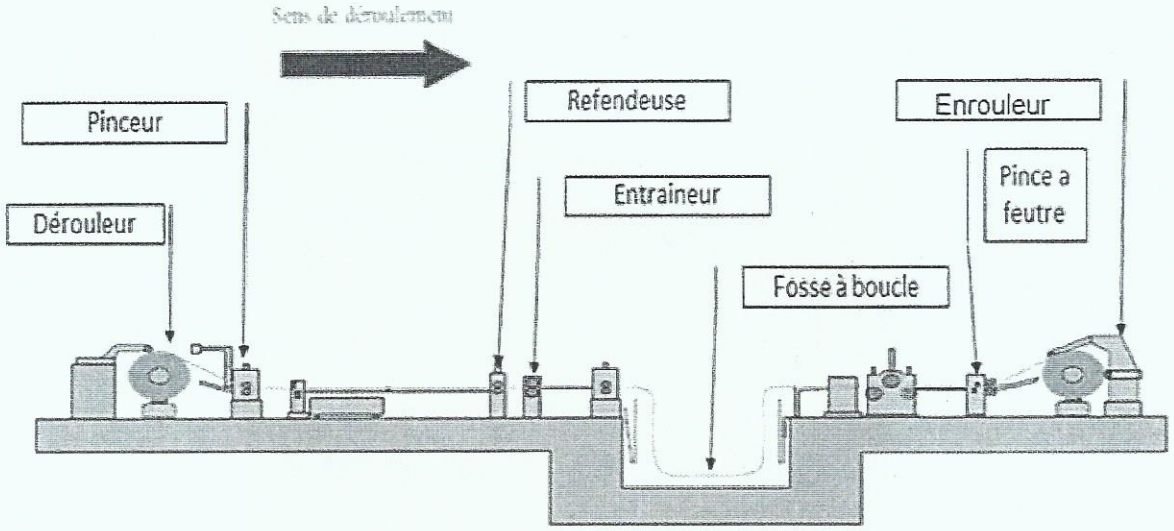



**Figure 6 Schéma de la ligne de refendage.**

Vitesse de ligne : 90 m/min.

Production : 20 T/h

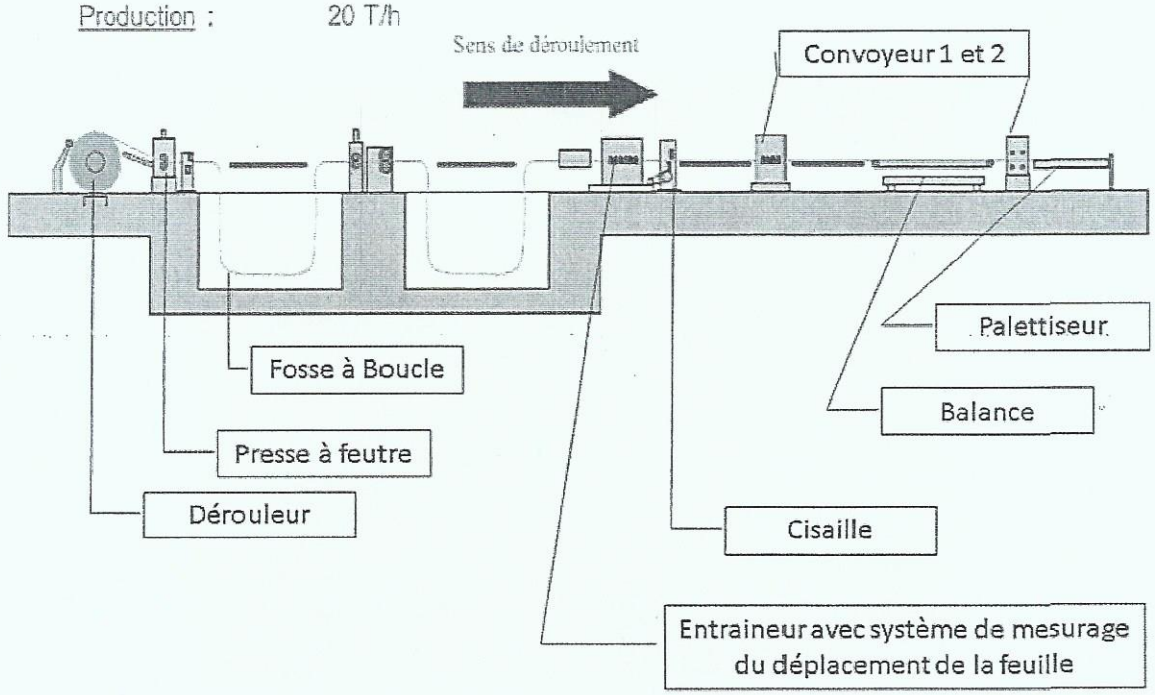



**Figure 7 Schéma de la ligne de cisaillage.**

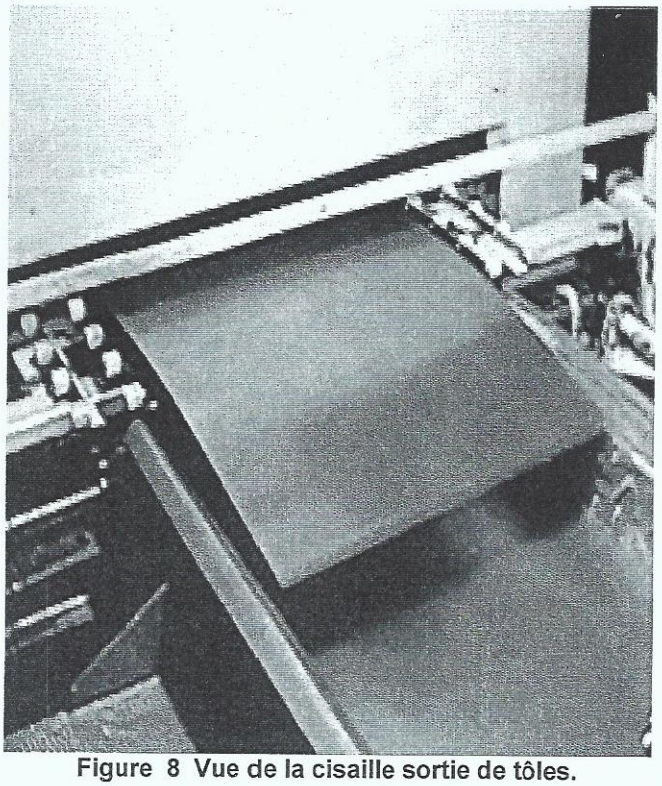


**Figure 8 Vue de la cisaille sortie de tôles.**

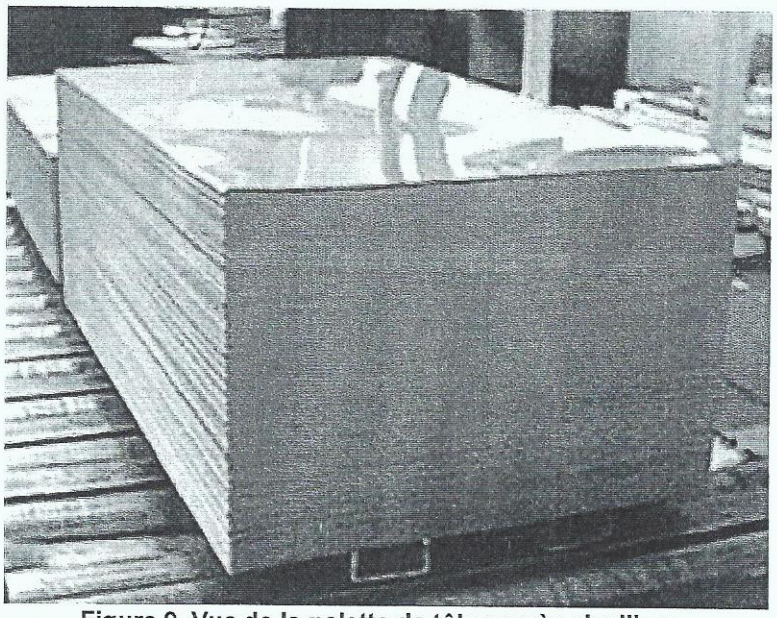


**Figure 9 Vue de la palette de tôles après cisaillage.**

# B. ANALYSE DU CONTEXTE

## Présentation générale

Les annexes à consulter pour cette partie sont notées : « Annexe 1 et 2 ».

### Le laminoir à froid :

L'enrouleur du laminoir accepte des coils d'acier de diamètre extérieur de 1830 mm.
Le diamètre intérieur des coils est de 610 mm.
L'acier respecte la norme DC01 EN 10130

Réponses à inscrire dans le document réponse.

**Question B 1 :**
Calculer la vitesse de rotation minimum et maximum de l'enrouleur pour obtenir une vitesse linéaire de la bande de métal de 90 m/min.
Exprimer les résultats en tours par minute.

**Question B 2 :**
Un coil destiné à la fabrication de tôles a une largeur de 1500 mm. Sa longueur totale déroulée est de 400 m et l'épaisseur de la feuille est égale à 3 mm.
La masse volumique de l'acier de ce coil est de 7850 kg / $m^3$.
Calculer la masse d'acier enroulé autour de ce coil.

**Question B 3 :**
La détection de la bande d'acier sur le laminoir se fait par des capteurs de proximité inductifs. Justifier le choix de tels capteurs

**Question B 4 :**
Les capteurs de proximité :
- sont fixés sur le bâti de la machine et ne doivent pas dépasser de ce châssis,
- la distance entre le capteur et la tôle n'est jamais supérieure à 8 mm,
- le traitement de l'information nécessite une sortie PNP normalement fermée.

Sachant que, pour des raisons économiques, le choix se portera sur un boîtier standard et de portée juste suffisante, donner la référence du capteur de proximité inductif (voir annexe 2).

# C. CONCEPTION ET CODAGE

## Diagramme des cas d'utilisation du système

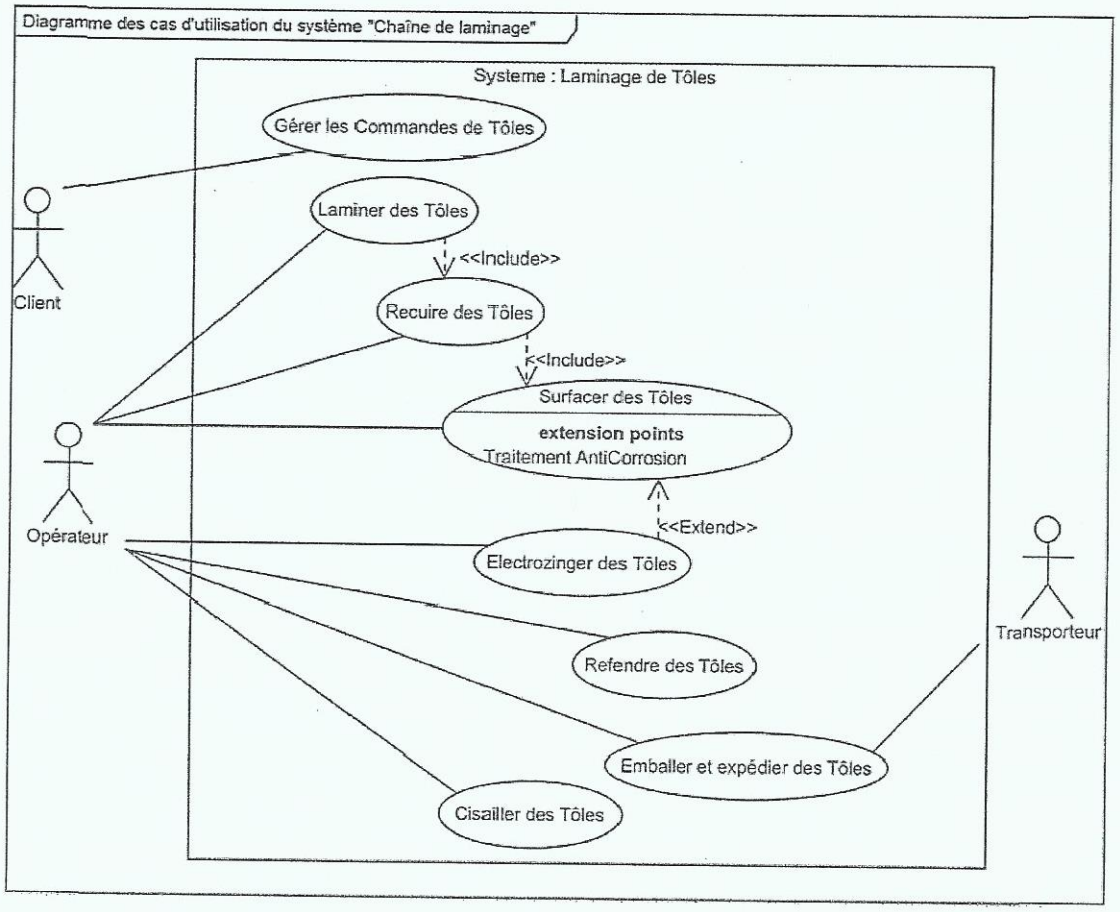



**Figure 10 : Diagramme des Cas d'utilisation du système « Chaîne de laminage »**

## Scénario du processus de refendage

Le refendage consiste à dérouler une bobine, la recouper dans le sens de la longueur et à la rebobiner. On peut voir, figure 6 du sujet, le schéma de la ligne de refendage.

Le scénario de refendage est le suivant :

1) une bobine à refendre est montée sur le dérouleur ;
2) les informations de la bobine sont extraites de la base de donnée ;
3) le dérouleur démarre en petite vitesse ;
4) le refendeur pince la tôle ;
5) le système d'asservissement de l'entraineur est mis en marche ;
6) l'enrouleur démarre en petite vitesse ;
7) la feuille d'acier déroulée est maintenue tendue en position horizontale entre le pinceur et l'entraineur, la refendeuse taille la feuille d'acier à la largeur souhaitée, la feuille retaillée passe par la fosse à boucle (dans cette zone tampon la feuille n'est plus sous tension).
   Dans la zone tampon, tant que le déroulage n'est pas terminé :
   - le dérouleur se met en grande vitesse ;
   - l'enrouleur accélère si la fosse est pleine et ralentit si la fosse n'est pas pleine.
8) A la fin de l'enroulage de la bobine, les informations sont mises à jour dans la base de données.

### Remarque :

Au niveau de l'entraineur, la feuille a une vitesse linéaire supérieure à celle en sortie du dérouleur afin de compenser le retard dû à l'arrêt pour le cisaillage.
Les fosses à boucle servent de tampon.

Document à consulter :

- **Figure 6 :** Schéma de la ligne de refendage

Réponses à inscrire dans le document réponse.

## C.1 Étude de la ligne de refendage

**Question C 1.1**
Expliquer les termes « *extend* » et « *include* » utilisés dans le diagramme des cas d'utilisation (figure 10)

**Question C 1.2**
A partir du scénario du processus de refendage ci-dessus, et de l'extrait du package « Découpage » de la figure 11, compléter le diagramme de séquence du cas d'utilisation « Refendre des tôles ».

## C.2 Étude des classes du système

Le système regroupant de multiples classes, elles ont été ordonnées à l'aide de packages. Les classes issues des diagrammes de séquences des processus de refendage et de cisaillage sont regroupées dans un package nommé « Découpage »

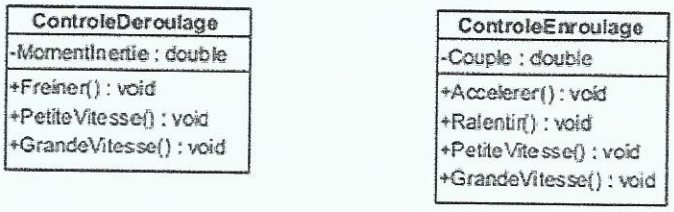



**Figure 11 Extrait du package « Découpage »**

**Question C 2.1**
Dans le package « Découpage » du diagramme de classes (annexe 3 et figure 11) on remarque que les classes *ControleEnroulage* et *ControleDeroulage* ont deux méthodes communes.
Proposer une solution utilisée en P.O.O pour éviter cette redondance.

Le constructeur de la classe GestionRefendage est le suivant :

```
GestionRendage::GestionRefendage(GestionBDD *RefGestionBDD)
{
    GestionBase = RefGestionBDD;
    Deroulage = new ControleDeroulage();
    Enroulage = new ControleEnroulage();
}
```

**Question C 2.2**
Compléter l'extrait du diagramme de classes en faisant apparaître les liaisons et les cardinalités entre la classe GestionRefendage et les classes ControleDeroulage et ControleEnroulage ainsi que la liaison et les cardinalités entre la classe GestionRefendage et la classe GestionBDD.

**Question C 2.3**
Proposer une déclaration en C++ de la classe *GestionRefendage* du package « Découpage » représentée ci-après (gestionrefendage.h).

| GestionRefendage |
| --- |
| -LargeurFeuille : double<br>-LargeurRefend : double<br>-DeroulageNonFini : boolean<br>-FossePleine : boolean<br>-Deroulage : ControleDeroulage*<br>-Enroulage : ControleEnroulage*<br>-GestionBase : GestionBDD* |
| +GestionRefendage(RefGestionBdd : GestionBDD*)<br>+GestionRefendage()<br>+~GestionRefendage()<br>+ChargerBobine() : void<br>+Refendre() : void<br>-Pincer() : void<br>-DemarrerAsservissementEntraineur() : void |

**Figure 12: classe GestionRefendage du package Decoupage.**

**Question C 2.4**
A partir du diagramme de séquence du processus de refendage traité à la question C1.2 ( voir document réponse C1.2 ), proposer la définition en C++ de la méthode *ChargerBobine()* de la classe *GestionRefendage* du package « Découpage » (gestionrefendage.cpp).

Le processus de Cisaillage utilise une balance permettant de peser les tôles découpées.
Le dialogue de la balance s'effectue au moyen de trames selon le protocole Modbus RTU (voir Annexe 4)

**Question C 2.5**
D'après le diagramme de classes et de la classe Modbus (annexe 4), justifier l'utilisation du type unsigned short utilisé pour l'attribut CRC16

**Question C 2.6**
A partir de l'algorithme du CRC de l'annexe 4, proposer une définition en C++ de la méthode *CalculCRC()* de la classe *Modbus* du package « Découpage » (modbus.cpp)

# D. ORGANISATION DES DONNÉES

Les informations sont stockées dans une base de données de type relationnel conforme à la norme SQL 92 (voir extrait du langage SQL annexe 5).
Ci-dessous est donné un extrait du MCD (modèle conceptuel de données) global du système. Seuls apparaissent ici les tables et les champs nécessaires au questionnement.

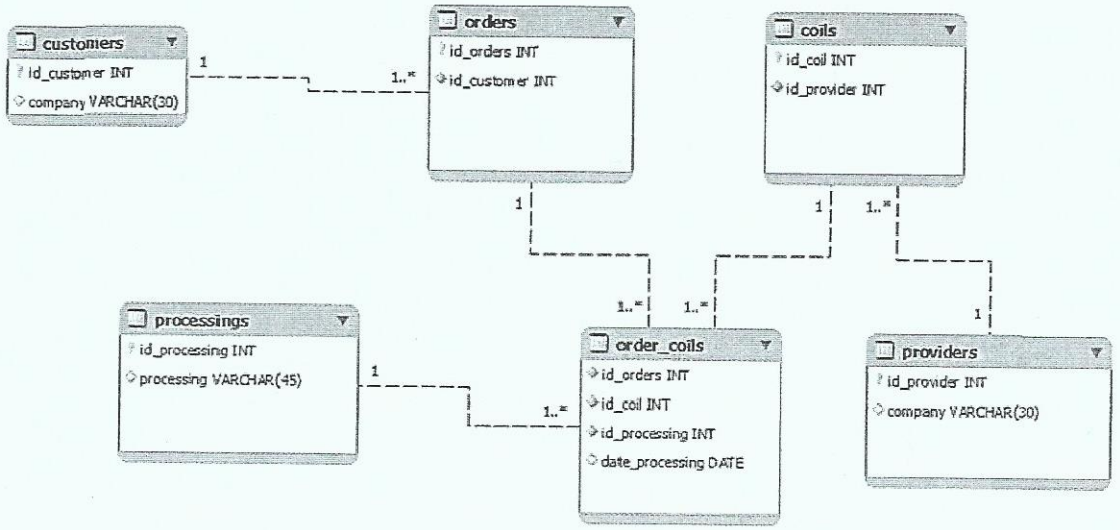



**Figure 13 : extrait du MCD**

***Question D.1***
Mise à part la table **orders_coils** (commande de bobines), le premier champ de chaque table, commençant par id_, est repéré par l'icône d'une clé. Ce champ est appelé clé primaire.
Compléter le tableau du document réponse.

***Question D.2***
La relation entre les tables **customers** (clients) et **orders** (commandes) étant de type 1 - 1..*.
Compléter le tableau du document réponse.

***Question D.3***
Que retourne la requête suivante ?

```
SELECT id_customer FROM customers WHERE (company = "FORD" OR company = "PSA") ;
```

Compléter le tableau du document réponse.

***Question D.4***
Que fait la requête suivante ?

```
UPDATE providers SET company = "ArcelorMittal" WHERE (company = "Arcelor")
```

Compléter le tableau du document réponse.

***Question D.5***
Ecrire la requête qui permet de supprimer dans la table **orders** la commande ayant l'identifiant "123"
Rédiger la requête sur le document réponse.

## E. COMMUNICATION

Documents à consulter : **Annexe 4 :** MODBUS

Le PC « **PSI de Supervision des ateliers** » est connecté à la « **bascule** » de pesage des tôles via une « **passerelle Modbus/TCP** ».

La bascule de pesage implémente le **protocole Modbus** pour communiquer avec la passerelle Modbus/TCP

**Question E.1**
Quel est le principe de l'arbitrage maître / esclave utilisé par Modbus ?

La liaison entre la bascule et la passerelle Modbus/TCP est assurée par une RS422 utilisant un câble en paires torsadées.

**Question E.2**
Indiquer l'avantage apporté par ce type de câble.

**Question E.3**
Donner le rôle du CRC dans la trame Modbus RTU

Le PC « **PSI de Supervision des ateliers** » accède au résultat du pesage des tôles effectué par la balance avec la **fonction 0x03 Modbus** : Read Holding Register (Lecture de N mots de sortie consécutifs)

Trame de requête Modbus RTU permettant d'interroger la bascule :

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 04 | 03 | 00 | 02 | 00 | 01 | 25 | 9F |

**Question E.4**
Souligner en traits continus le PDU Modbus.
Souligner en traits pointillés l'ADU Modbus.
Analyser la trame de requête ci-dessus en donnant le nom, la valeur et la signification de chacun des champs.
Documentation à utiliser :
- Format trame RTU : Annexe 4
- Code fonction Modus : Annexe 4

Le résultat du pesage est stocké dans un registre Modbus (2 octets) à l'adresse 0x02 en kg au format d'un entier 16 bits.

**Question E.5**
Donner la trame de réponse de la balance (sans calculer le CRC) pour une pesée de 150 kg. Indiquer la valeur des octets en hexadécimal.
Commenter chacun des champs de la trame de réponse.

**Question E.6**
Calculer le temps de transmission de la trame de requête en considérant les paramètres suivants de la liaison série : Vitesse : 9600 bits/s, Longueur des données : 8 bits, Parité : Paire, Nombre de bits de stop : 1

# F. RÉSEAUX

## F.1 Architecture du réseau

L'architecture des réseaux de l'entreprise est présentée en annexe 6.
Le secteur PRODUCTION comporte 6 sous-réseaux :

- le LAN PROCESS, relié au réseau de l'entreprise via le serveur de supervision PSI.
- les LANs QUARTO, EBNER, FABRICOM, SKIN PASS et ZINGAGE, reliés au LAN PROCESS par l'intermédiaire du routeur PROCESS.

## F.2 Ethernet

**Question F.2.1**
Le LAN PROCESS utilise la technologie de réseau Ethernet. Peut-on qualifier ce réseau de probabiliste ou déterministe ? Justifier votre réponse.

**Question F.2.2**
Le support de transmission est de type 100 base T.
Expliquer les termes :
100 ?
T ?

## F.3 Adressage IP

Les sous-réseaux du secteur PRODUCTION font partie d'un même bloc d'adresses 192.168.0.0/16
Ce bloc d'adresses est découpé en plusieurs sous-réseaux de masque /19

**Question F.3.1**
Exprimer les masques en binaire et en décimal pointé.
Compléter le tableau dans le document réponse.

**Question F.3.2**
Avec un masque /19, combien de sous-réseau au total est-il possible de découper dans le bloc d'adresses de masque /16 ? Justifier votre réponse.

Avec le découpage obtenu, les 6 premiers sous-réseaux sont alloués aux sous-réseaux du secteur PRODUCTION.

**Question F.3.3**
Pour chacun des sous-réseaux, donner l'adresse du sous-réseau, la plage d'adresses utilisables et l'adresse de diffusion.
Compléter le tableau dans le document réponse.

**Question F.3.4**
Expliquer ce qu'est une adresse de « *broadcast* » ?

## F.4 Accès Internet et routage

Pour l'accès au Web, la société a acquis un routeur WAN pare-feu avec un abonnement spécifique à un fournisseur Internet.
Après l'installation du routeur, il faut configurer les stations clientes.
Pour des raisons de sécurité, les stations du secteur PRODUCTION ne doivent en aucun cas avoir accès à Internet. Seules les stations du LAN ADMINISTRATION peuvent avoir accès à Internet.

**Question F.4.1**
Donner l'adresse de passerelle par défaut à écrire dans les paramètres réseau des stations souhaitant accéder à Internet.

Le serveur PSI peut effectuer du routage IP entre le LAN ADMINISTRATION et le LAN PROCESS.

**Question F.4.2**
Renseigner l'entrée de table de routage d'un hôte du LAN ADMINISTRATION afin de joindre tout hôte du LAN PROCESS.

L'acronyme *Dmz* signifie « demilitarized zone » ou « zone démilitarisée »

**Question F.4.3**
Expliquer la fonction de cette zone Dmz.

**Question F.4.4**
Quelles seraient les fonctionnalités apportées par l'installation d'un serveur proxy HTTP ?
Répondre en remplissant le tableau du document réponse.

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

## E4 - ÉTUDE D'UN SYSTÈME INFORMATISÉ U4

SESSION 2015

---

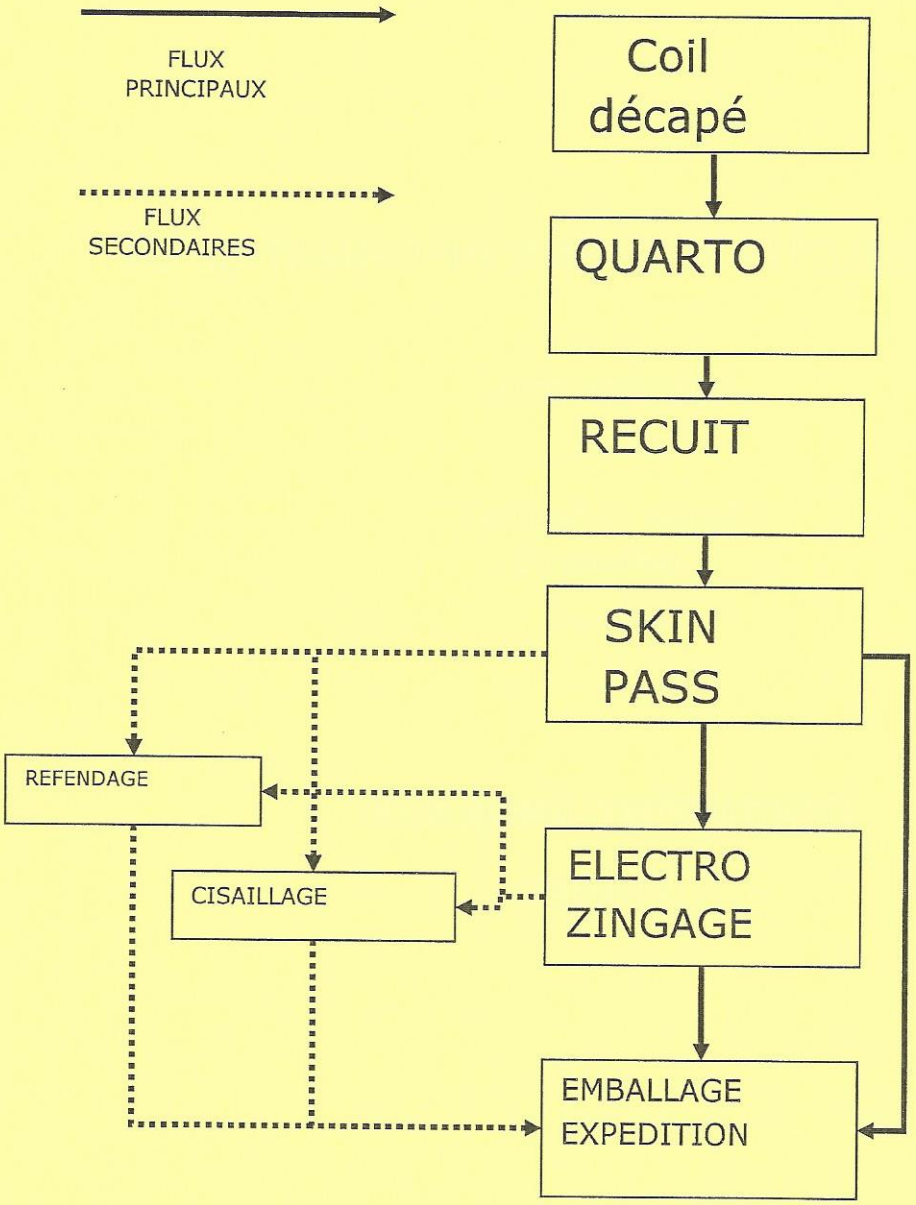
Durée : 6 heures
Coefficient 5

---

## DOCUMENT ANNEXES

(12 PAGES)

# Annexe 1 Flux de matière




**Glossaire :**

COÏL : bobine de métal enroulé

QUARTO : cage de laminage permettant de réduire l'épaisseur de la tôle

RECUIT : Traitement thermique permettant de redonner les caractéristiques de dureté au métal

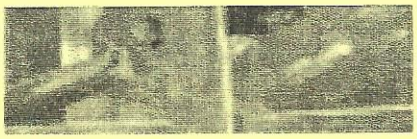
SKIN PASS : Traitement de surface

ELECTRO-ZINGAGE : Dépôt d'une pellicule de zinc sur la tôle

REFENDAGE : Découpe de la feuille de tôle d'une bobine dans le sens de la longueur (de la tôle déroulée) pour obtenir en bout de chaîne deux ou plusieurs bobines de largeur plus faible.

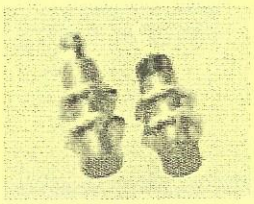
CISAILLAGE : Découpe de la feuille de tôle dans le sens de la largeur (de la tôle déroulée) pour obtenir des plaques.

# Annexe 2 Capteurs inductifs



## Capteurs standards

**Détecteurs de proximité inductifs IME 18**



**Avantages**

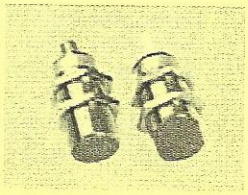
- Indice de protection IP 67
- Portée de 2 à 10 mm
- Fréquence de commutation jusqu'à 1 kHz
- Fonction NO ou NC
- Sortie NPN ou PNP
- Câble ou connecteur
- Version courte ou longue

| Caractéristiques techniques | |
|---|---|
| Diamètre du filetage x pas [mm] | M18 x 1 |
| Matériau du boîtier | Laiton plaqué nickel |
| Tension d'alimentation | CC 10 ... 30 V |
| Sorties | PNP/NPN |
| Fréq. de commutation max. | 1 000/s |
| Indice de protection | IP 67/IP 68 |
| Raccordement | Câble / connecteur |

| Référence | Désignation | Description | Portée | Boîtier court 50 mm | Boîtier std 69 mm | Sortie PNP | Raccordement |
|---|---|---|---|---|---|---|---|
| 1040937 | IME18-05BPOZC0K | Gamme standard | 5 mm noyable | ✔ | | normalement fermée | connecteur M12, 4 pôles |
| 1040938 | IME18-05BPOZC0S | | | | ✔ | normalement fermée | |
| 1040933 | IME18-05BPSZC0K | | | ✔ | | normalement ouverte | |
| 1040934 | IME18-05BPSZC0S | | | | ✔ | normalement ouverte | |
| 1040953 | IME18-08NPOZC0K | | 8 mm non noyable | ✔ | | normalement fermée | |
| 1040954 | IME18-08NPOZC0S | | | | ✔ | normalement fermée | |
| 1040949 | IME18-08NPSZC0K | | | ✔ | | normalement ouverte | |
| 1040950 | IME18-08NPSZC0S | | | | ✔ | normalement ouverte | |
| 1040969 | IME18-08BPOZC0K | | 8 mm noyable | ✔ | | normalement fermée | |
| 1040970 | IME18-08BPOZC0S | | | | ✔ | normalement fermée | |
| 1040965 | IME18-08BPSZC0K | | | ✔ | | normalement ouverte | |
| 1040966 | IME18-08BPSZC0S | | | | ✔ | normalement ouverte | |
| 1040985 | IME18-12NPOZC0K | Gamme advanced | 12 mm non noyable | ✔ | | normalement fermée | |
| 1040986 | IME18-12NPOZC0S | | | | ✔ | normalement fermée | |
| 1040981 | IME18-12NPSZC0K | | | ✔ | | normalement ouverte | |
| 1040982 | IME18-12NPSZC0S | | | | ✔ | normalement ouverte | |

# Capteurs standards

**Détecteurs de proximité inductifs IME 30**



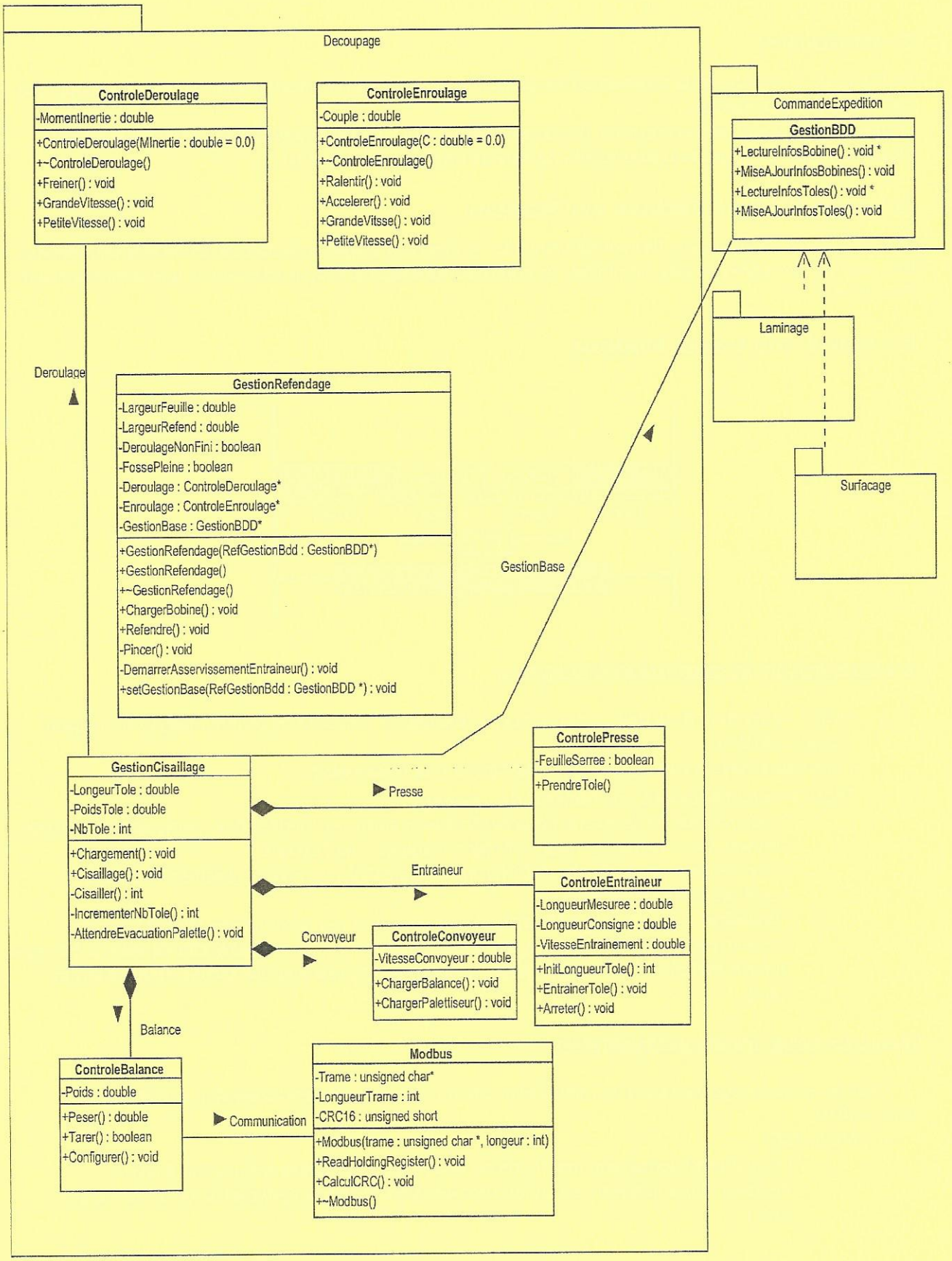
**Avantages**

- Indice de protection IP 67
- Portée de 2 à 10 mm
- Fréquence de commutation jusqu'à 500 kHz
- Fonction NO ou NC
- Sortie NPN ou PNP
- Câble ou connecteur
- Version courte ou longue

| Caractéristiques techniques | |
|---|---|
| Diamètre du filetage x pas [mm] | M30 x 1 |
| Matériau du boîtier | Laiton plaqué nickel |
| Tension d'alimentation | CC 10 ... 30 V |
| Sorties | PNP/NPN |
| Fréq. de commutation max. | 500/s |
| Indice de protection | IP 67/IP 68 |
| Raccordement | Câble / connecteur |
| Température d'utilisation | -20 °C ... +65 °C |

| Référence | Désignation | Description | Portée | Boîtier court 52 mm | Boîtier std 71 mm | Sortie PNP | Raccordement |
|---|---|---|---|---|---|---|---|
| 1041001 | IME30-10BPOZC0K | Gamme standard | 10 mm noyable | ✔ | | normalement fermée | connecteur M12 4 pôles |
| 1041002 | IME30-10BPOZC0S | | 10 mm noyable | | ✔ | normalement fermée | |
| 1040997 | IME30-10BPSZC0K | | 10 mm noyable | ✔ | | normalement ouverte | |
| 1040998 | IME30-10BPSZC0S | | 10 mm noyable | | ✔ | normalement ouverte | |
| 1041017 | IME30-15NPOZC0K | | 15 mm non noyable | ✔ | | normalement fermée | |
| 1041018 | IME30-15NPOZC0S | | 15 mm non noyable | | ✔ | normalement fermée | |
| 1041013 | IME30-15NPSZC0K | | 15 mm non noyable | ✔ | | normalement ouverte | |
| 1041014 | IME30-15NPSZC0S | | 15 mm non noyable | | ✔ | normalement ouverte | |
| 1041033 | IME30-15BPOZC0K | | 15 mm noyable | ✔ | | normalement fermée | |
| 1041034 | IME30-15BPOZC0S | | 15 mm noyable | | ✔ | normalement fermée | |
| 1041029 | IME30-15BPSZC0K | | 15 mm noyable | ✔ | | normalement ouverte | |
| 1041030 | IME30-15BPSZC0S | | 15 mm noyable | | ✔ | normalement ouverte | |
| 1041049 | IME30-20NPOZC0K | Gamme advanced | 20 mm non noyable | ✔ | | normalement fermée | |
| 1041050 | IME30-20NPOZC0S | | 20 mm non noyable | | ✔ | normalement fermée | |
| 1041045 | IME30-20NPSZC0K | | 20 mm non noyable | ✔ | | normalement ouverte | |
| 1041046 | IME30-20NPSZC0S | | 20 mm non noyable | | ✔ | normalement ouverte | |

# Annexe 3 : Diagramme de classes partiel du système



# Annexe 4 : Extrait documentation MODBUS

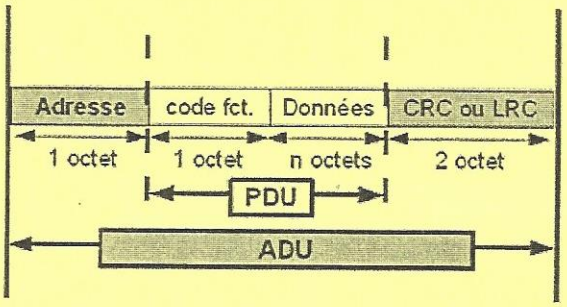
## Présentation

MODBUS est un protocole de communication couramment utilisé dans l'industrie pour faire communiquer des automates programmables.
Le protocole MODBUS est un protocole de dialogue basé sur une structure hiérarchisée entre un maître et plusieurs esclaves. Les esclaves possèdent une adresse comprise entre 1 et 64.

## Échange Maître/ Esclave sur Modbus

Le maître interroge un esclave de numéro unique sur le réseau et attend de la part de cet esclave une réponse. Lorsque l'esclave envoie sa réponse, il place sa propre adresse dans le champ adresse afin que le maître puisse l'identifier.

## Format d'une trame Modbus




<u>Une trame Modbus est composée de deux parties :</u>

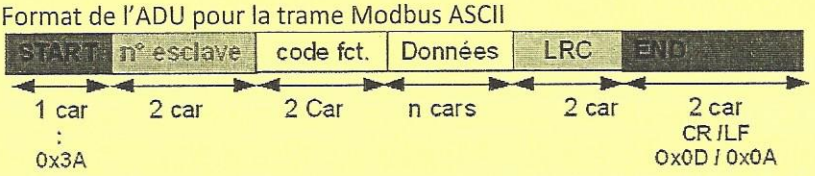
- **"ADU" Application Data Unit** : Cette partie est fonction de la couche de communication sur laquelle s'appuie Modbus.
  L'exemple ci-dessus correspond à un ***ADU Modbus sur liaison série***.
- **"PDU" Protocol Data Unit** : Est composé des champs :
  - ***"Code fonction"*** : Ce champ est prédéfini par le protocole Modbus (voir paragraphe 2 de cette annexe). Ce code fonction définie l'action à exécuter sur l'esclave.
    Par exemple le code fonction 0x0F (write Multiple Coils) permet de modifier les sorties numériques sur un esclave.
  - ***"Données"*** : Dans le champ "Données" sont présentes des informations relatives au code fonction (exemple : adresse de registre pour l'écriture sur les sorties) et des données à échanger entre le maître et l'esclave (exemple : les informations relatives aux sorties à mettre à l'état haut).

<u>Codage des trames Modbus sur liaison série :</u>

Deux types de codage peuvent être utilisés pour communiquer sur un réseau Modbus sur liaison série : **ASCII** ou **RTU**

- Mode ***ASCII*** :
  - Chaque octet composant une trame est codé avec 2 caractères ASCII (2 fois 8 bits).
    Par exemple, la valeur 01 est codée avec les codes ASCII des symboles '0' et '1' (soit 0x30 et 0x31).

- Format de l'ADU pour la trame Modbus ASCII

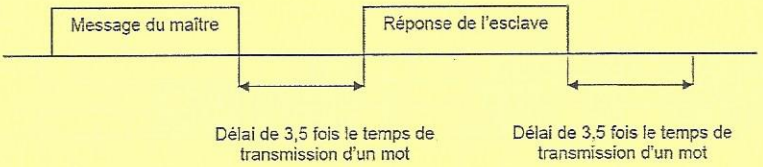



**LRC** : C'est la somme en hexadécimal modulo 256 du contenu de la trame hors délimiteurs (START et END), complémentée à 2. Le LRC est transmise en ASCII.

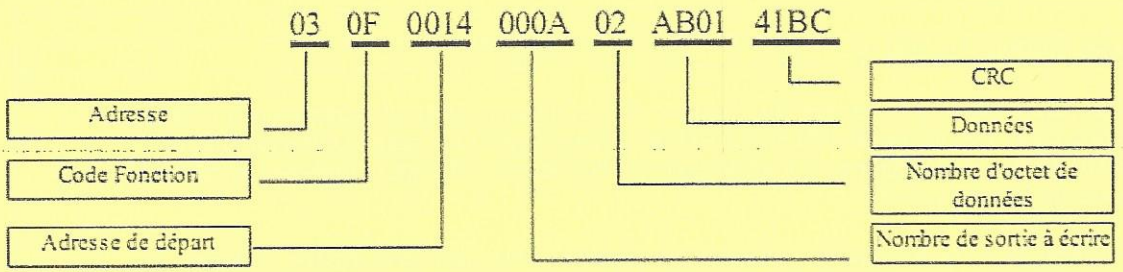
- Mode ***RTU*** :
  - Les octets composants la trame ne sont pas codés.
    Par exemple, la valeur $(15)_{10}$ aura pour octet 0x0F ou $(00001111)_2$.
  - Format de l'ADU pour la trame Modbus RTU




**Silence** : Les trames RTU ne comportent ni entête, ni délimiteur de fin.
La station réceptrice contrôle le temps séparant deux caractères consécutifs, s'il est supérieur à 3 caractères et demi, elle considère que le prochain caractère est un début de trame.




**Exemple d'une requête modbus RTU pour écrire une série de 10 sorties commençant à l'adresse 20 sur l'esclave à l'adresse 3:**



## Fonctions Modbus

### 15 (0x0F) Write Multiple Coils

This function code is used to force each coil in a sequence of coils to either ON or OFF in a remote device. The Request PDU specifies the coil references to be forced. Coils are addressed starting at zero. Therefore coil numbered 1 is addressed as 0.

The requested ON/OFF states are specified by contents of the request data field. A logical '1' in a bit position of the field requests the corresponding output to be ON. A logical '0' requests it to be OFF.

The normal response returns the function code, starting address, and quantity of coils forced.

**Request PDU**

| | | |
|---|---|---|
| Function code | 1 Byte | 0x0F |
| Starting Address | 2 Bytes | 0x0000 to 0xFFFF |
| Quantity of Outputs | 2 Bytes | 0x0001 to 0x07B0 |
| Byte Count | 1 Byte | N* |
| Outputs Value | N* x 1 Byte | |

*N = Quantity of Outputs / 8, if the remainder is different of 0 ⇒ N = N+1

**Response PDU**

| | | |
|---|---|---|
| Function code | 1 Byte | 0x0F |
| Starting Address | 2 Bytes | 0x0000 to 0xFFFF |
| Quantity of Outputs | 2 Bytes | 0x0001 to 0x07B0 |

**Error**

| | | |
|---|---|---|
| Error code | 1 Byte | 0x8F |
| Exception code | 1 Byte | 01 or 02 or 03 or 04 |

### 03 (0x03) Read Holding Registers

This function code is used to read the contents of a contiguous block of holding registers in a remote device. The Request PDU specifies the starting register address and the number of registers. In the PDU Registers are addressed starting at zero. Therefore registers numbered 1-16 are addressed as 0-15.

The register data in the response message are packed as two bytes per register, with the binary contents right justified within each byte. For each register, the first byte contains the high order bits and the second contains the low order bits.

**Request :**

| | | |
|---|---|---|
| **Function code** | 1 Byte | **0x03** |
| **Starting Address** | 2 Bytes | 0x0000 to 0xFFFF |
| **Quantity of Registers** | 2 Bytes | 1 to 125 (0x7D) |

**Response :**

| | | |
|---|---|---|
| **Function code** | 1 Byte | **0x03** |
| **Byte Count** | 1 Bytes | 2 x **N*** |
| **Register Value** | N* x 2 Bytes | 1 to 125 (0x7D) |

*N = Quantity of Register

**Error :**

| | | |
|---|---|---|
| **Function code** | 1 Byte | **0x83** |
| **Exception code** | 1 Bytes | 01 or 02 or 03 or 04 |
| **Register Value** | N* x 2 Bytes | 1 to 125 (0x7D) |

## Algorithme pour le calcul du CRC 16 bits d'une Trame:

```
Debut

    CRC = FFFFh
    OctetPointe = Premier octet de la trame
    Pour (chaque octet de la trame)
        CRC = CRC OU_Exclusif OctetPointe
        Pour Cpt variant de 1 à 8
            Si ((CRC AND 0001h) non NUL) alors
                CRC = CRC décalé d'un bit à droite
                CRC = CRC OU_Exclusif A001h
            sinon
                CRC = CRC décalé d'un bit à droite
            finSi
        finPour
        OctetPointe = Octet suivant de la trame
    FinPour

Fin.
```

# Annexe 5 Aide-mémoire sur le langage S.Q.L.

## 1 DEFINITION

S.Q.L. (Structured Query Language) est un Langage normalisé de Requêtes Structurées et un standard d'accès aux bases de données relationnelles.

## 2 LES INSTRUCTIONS DE SQL

### 2.1 L'instruction 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 fonctions de critères.
- D'utiliser des fonctions arithmétiques et de groupements pour des calculs.

Syntaxe générale de l'instruction **SELECT** :

**SELECT** *Liste des champs séparés par une virgule*
**FROM** *Liste des tables concernées, séparées par une virgule* **WHERE** *Liste des critères de choix*

### 2.2 La projection

**SELECT** [DISTINCT] nomcol1 [, nomcol2 , ...] **FROM** nomtable1 [ , nomtable2 , ....] ;

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.

### 2.3 La restriction (ou sélection)

**SELECT** nom_col1 [ , nom_col2, ....] **FROM** nom_table1 [, nom_Table2,....] **WHERE** conditions ;

La clause WHERE permet de sélectionner dans la table obtenue par SELECT ... FROM ... les tuples correspondants à des critères précis.

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 n_dep FROM LIVRAISON WHERE (prise > 25 AND prise < 50) ;
```

Extrait les numéros de dépositaires avec la restriction des quantités livrées comprises entre 26 et 49.

### 2.4 La jointure (ou sélection sur plusieurs tables)

En SQL, il est possible d'enchaîner plusieurs jointures dans la même instruction SELECT.

- En SQL de base

**SELECT** * **FROM** table1, table2, table3, ...

**WHERE** table1.attribut1=table2.attribut1 **AND** table2.attribut2=table3.attribut2 **AND** ...;

Exemple :

SELECT * FROM Produit, Détail_Commande

WHERE Produit.CodePrd=Détail_Commande.CodePrd ;

ou en utilisant des alias pour les noms des tables :

SELECT * FROM Produit A, Détail_Commande B

WHERE A.CodePrd=B.CodePrd ;

- Avec la clause **INNER JOIN** (jointure dite interne) à partir du SQL2, supportée aujourd'hui par tous les SGBDR :

**SELECT** *

**FROM** table1 **INNER JOIN** table2 **ON** table1.attribut1=table2.attribut1 **INNER JOIN** table3 **ON** table2.attribut2=table3.attribut3... ;

Le mot clé INNER est facultatif sur la plupart des SGBDR (sauf MS Access).

Cette notation rend plus lisible la requête en distinguant clairement les conditions de jointures, derrière ON, et les éventuelles conditions de sélection ou restriction, derrière WHERE.

De plus, l'oubli d'un ON (et donc de la condition de jointure) empêchera l'exécution de la requête, alors qu'avec l'ancienne notation, l'oubli d'une condition de jointure derrière WHERE, n'empêche pas l'exécution de la requête, produisant alors un bien coûteux produit cartésien entre les tables !

Le même exemple que précédemment en utilisant aussi les alias :

SELECT *

FROM Produit A INNER JOIN Détail_Commande B ON A.CodePrd=B.CodePrd ;

La norme SQL2 définit aussi l' équi-jointure naturelle, joignant les 2 tables sur l'ensemble des attributs qu'elles ont en commun, mais en ne gardant qu'une seule colonne pour chaque attribut joint, contrairement aux 2 expressions précédentes :

**SELECT** *

**FROM** table1 **NATURAL JOIN** table2 ;

Il est aussi possible de restreindre (ou préciser) le ou les attributs de jointure avec USING :

**SELECT** *

**FROM** table1 **INNER JOIN** table2 **USING (**attribut1**)** ;

NATURAL JOIN et USING ne sont pas supportés par tous les SGBDR.

### 2.5 L'instruction INSERT

INSERT permet d'ajouter un ou plusieurs enregistrements dans une table. ***Insertion d'un enregistrement :***

| |
|---|
| **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') ;
```

Insert un nouvel enregistrement dans la table dépositaire avec les valeurs :

N_dep=68 ; nom_dep='Quentin' ; adr_dep='Marseille'

### 2.6 L'instruction UPDATE

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

| |
|---|
| **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'.

### 2.7 L'instruction DELETE

Elle permet de supprimer un enregistrement d'une table.

| |
|---|
| **DELETE FROM** nom_table **WHERE** conditions... ; |

# Annexe 6 : Architecture informatique.

