

Automatique et Informatique Industrielle
Systèmes Automatisés et Réseaux Industriels

Module RCI : Réseaux de Communication Industrielle

Pr. Eddy BAJIC
IUT Nancy Brabois
Université de Lorraine

Introduction aux Réseaux dans les Automatismes

Pr. Eddy BAJIC
IUT Nancy Brabois
Nancy Université

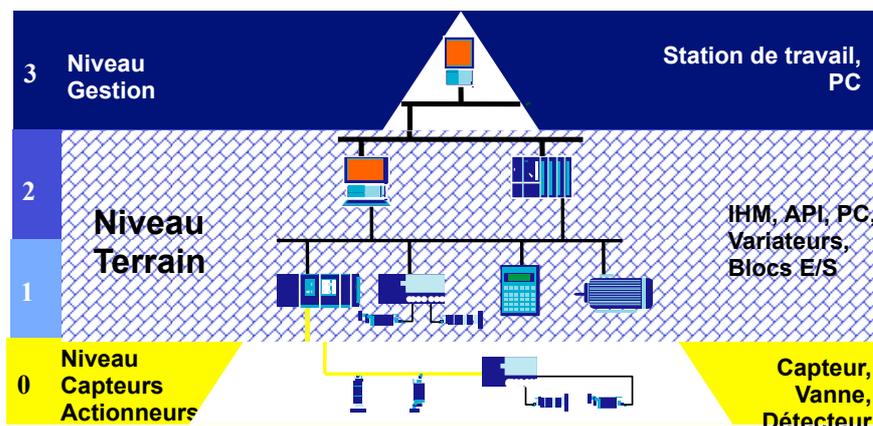
- *Pyramide CIM*
- *Evolution des Automatismes*
- *Comparatif Câblage classique / Réseau*
- *Réseaux de Terrain et Modèle OSI*
- *Marché des Réseaux de Terrain*
- *Normalisation*
- *Standards de Câblage : TéléAlimentation et Sécurité Intrinsèque*

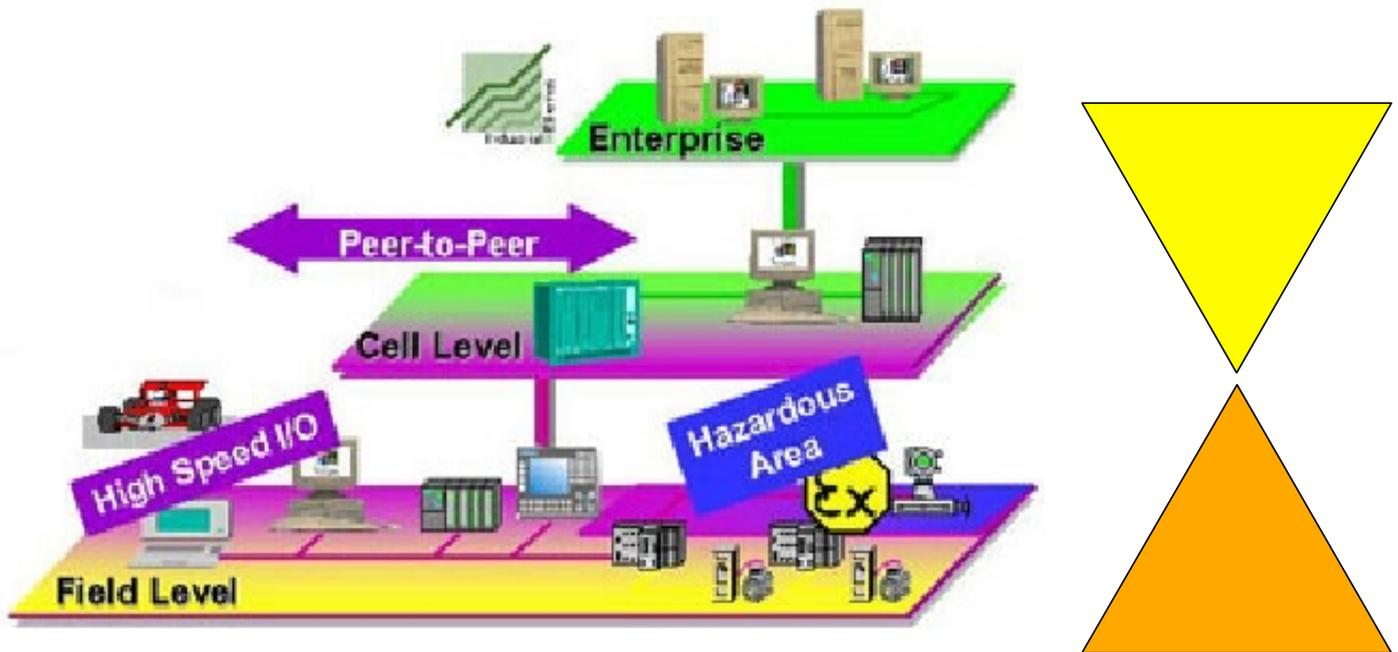
La Pyramide CIM de l'Automatisation

Modèle d'architecture de communication dans une entreprise datant des années 1980, basé sur Plusieurs niveaux fonctionnels hiérarchisés

Chaque niveau dispose de fonctions et de besoins d'échanges d'information distincts selon sa **hiérarchie** dans la pyramide appelée aussi

Les réseaux de communication seront aussi classifiés selon ces Niveaux.





Les Contraintes des Communications Industrielles sont :

Hétérogénéité des équipements ⇒



Environnement « dur » ⇒

Rapidité de fonctionnement ⇒



Sûreté de fonctionnement ⇒



Non propriétaire : Qualité d'un réseau à pouvoir accepter des équipements provenant de différentes sources ou constructeurs.

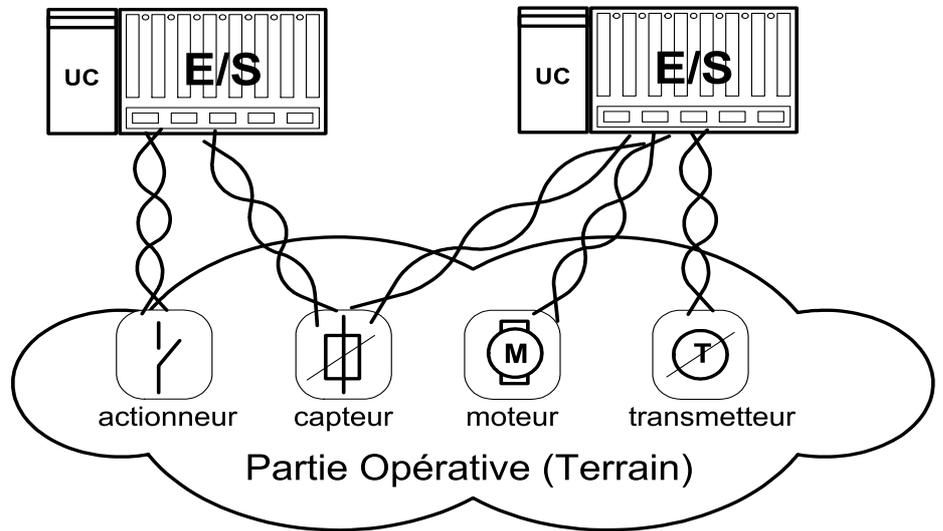
Interopérabilité : Capacité d'un équipement à pouvoir réaliser des actions coordonnées avec d'autres équipements au moyen de communication réseau

Interchangeabilité : Capacité d'un équipement à pouvoir remplacer fonctionnellement un autre équipement par simple remplacement physique sans reconfiguration ou adaptations nécessaires.

Evolution des Automatismes (<1980)

Evolution des Structures de Contrôle / Commande des Automatismes Programmés

< 1980 Automatismes Indépendants

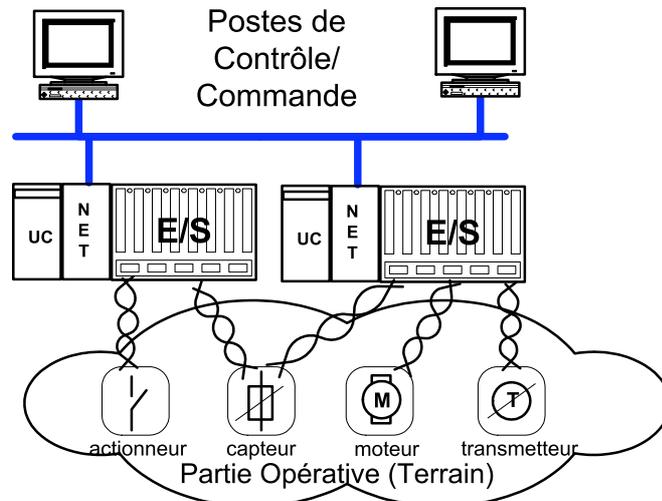


Evolution des Automatismes (>1985)

Evolution des Structures de Contrôle / Commande des Automatismes Programmés

1985 Automatismes en Réseau

- Supervision des UC

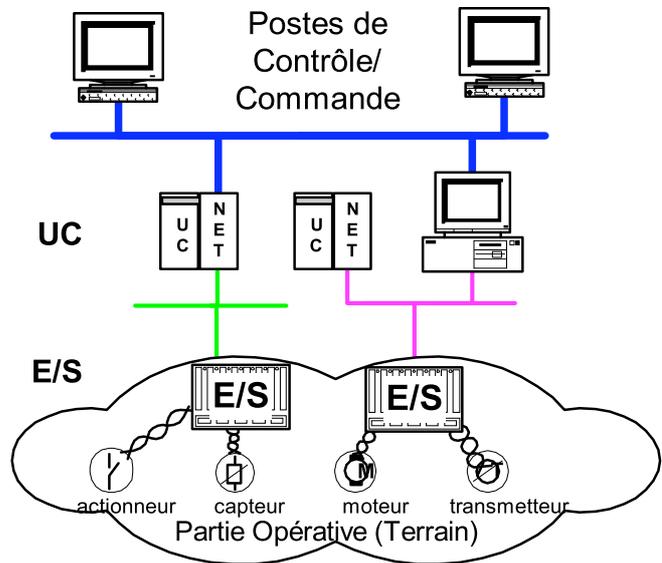


↔ Réseau « d'Automates »
↔ Réseau « d'Atelier »

Evolution des Structures de Contrôle / Commande des Automatismes Programmés

1993 Automatismes Hiérarchisés

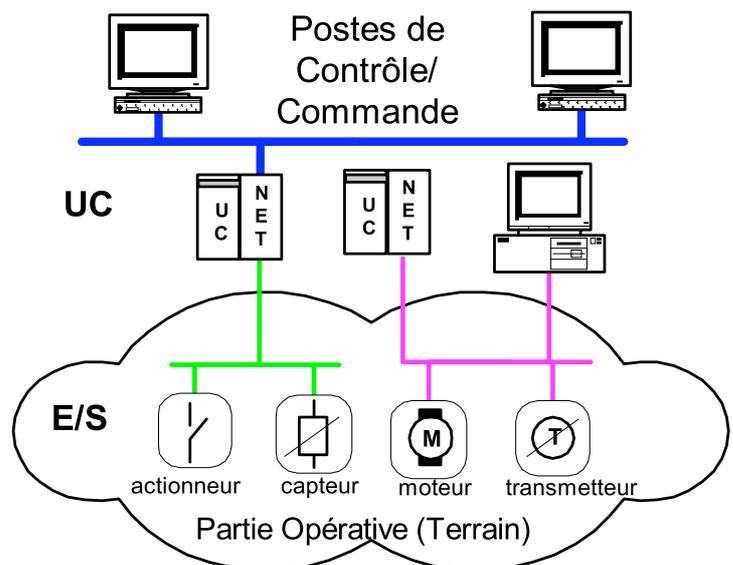
- Contrôle Commande Distribué
- Distance de Câblage E/S



↳ Réseau d'Entrées / Sorties Déportées

1996 Automatismes Répartis et Distribués en réseau

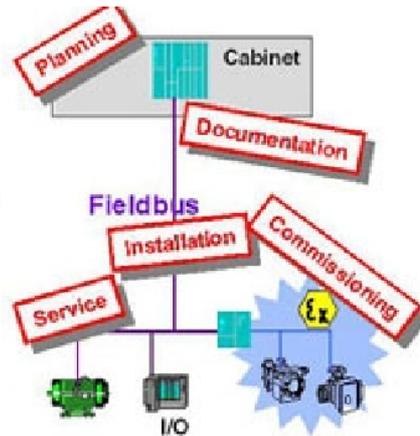
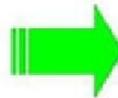
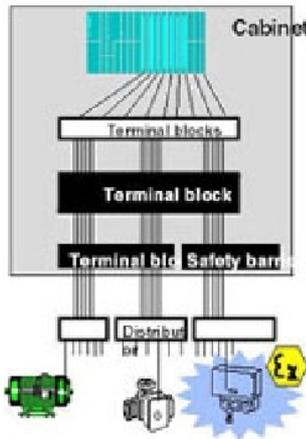
- Gestion Hiérarchisée de la commande
- Contrôle Distribué
- Les Capteurs Actionneurs sont directement sur le Réseau



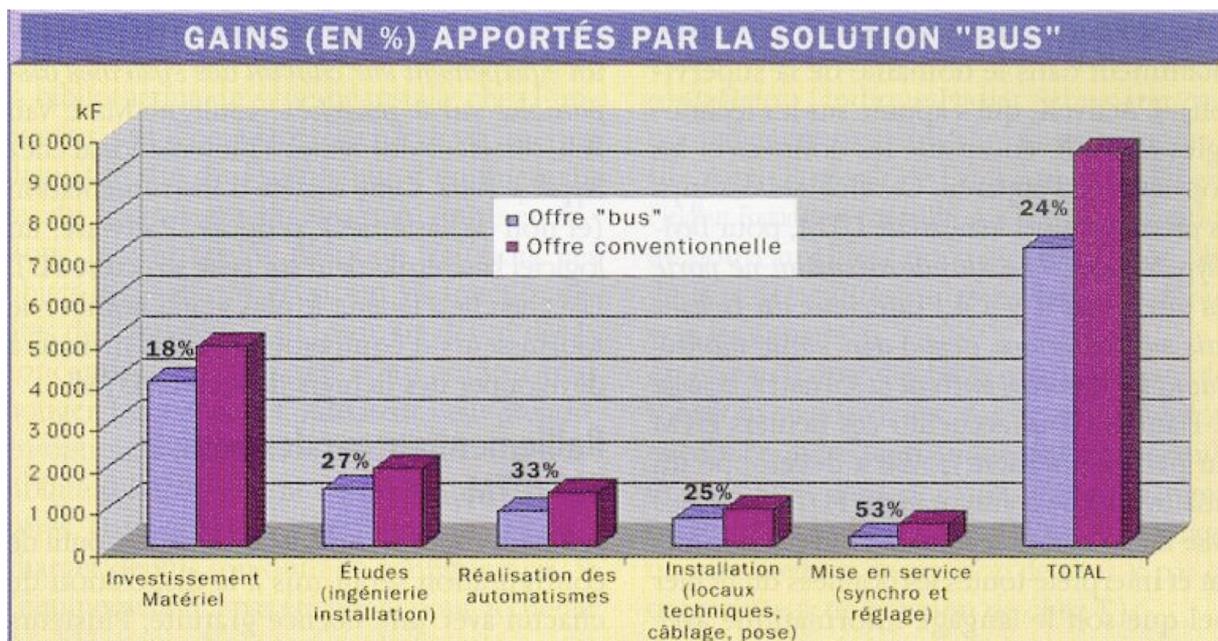
Du câblage Analogique au Câblage numérique



- Réduction du câblage
- Distribution de la commande sur le réseau
- Connexion des Capteurs Actionneurs au plus près du Réseau
- Communication numérique / communication analogique



Comparatif Conventiionnelle / Réseau



Etude réalisée par Fisher-Rosemount, publiée dans Mesure Avril 2001, rapporte une évaluation des coûts d'automatisation pour un grand groupe industriel chimique. Deux approches ont été évaluées et comparées :

- l'une correspondant à une solution dite "**conventionnelle**" : câblage 4-20 mA, protocole HART, E/S TOR
- l'autre correspondant à une solution **bus de terrain** Fieldbus Foundation FF-H1 à 31,25 KBps

Licence Professionnelle AII-SARI :

Automatique et Informatique Industrielle Systèmes Automatisés et Réseaux Industriels

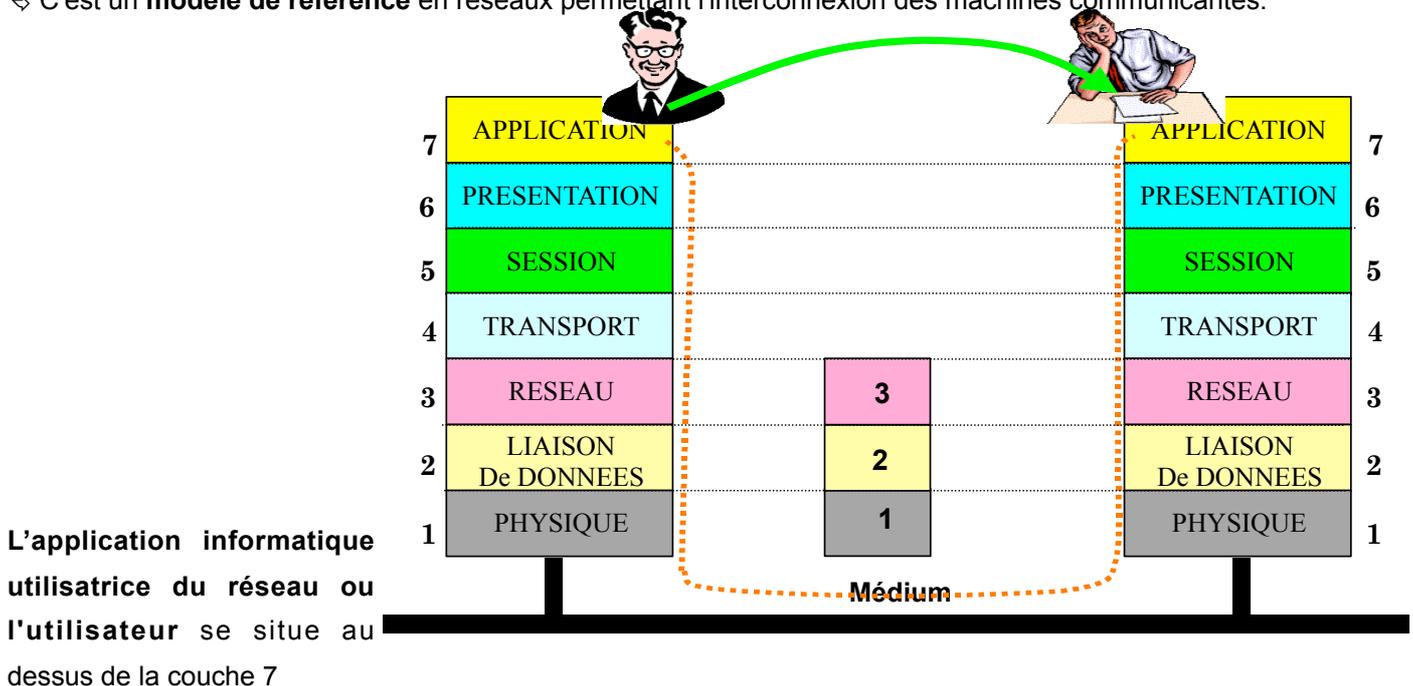
LE MODELE OSI

Pr. Eddy BAJIC
IUT Nancy Brabois
Nancy Université

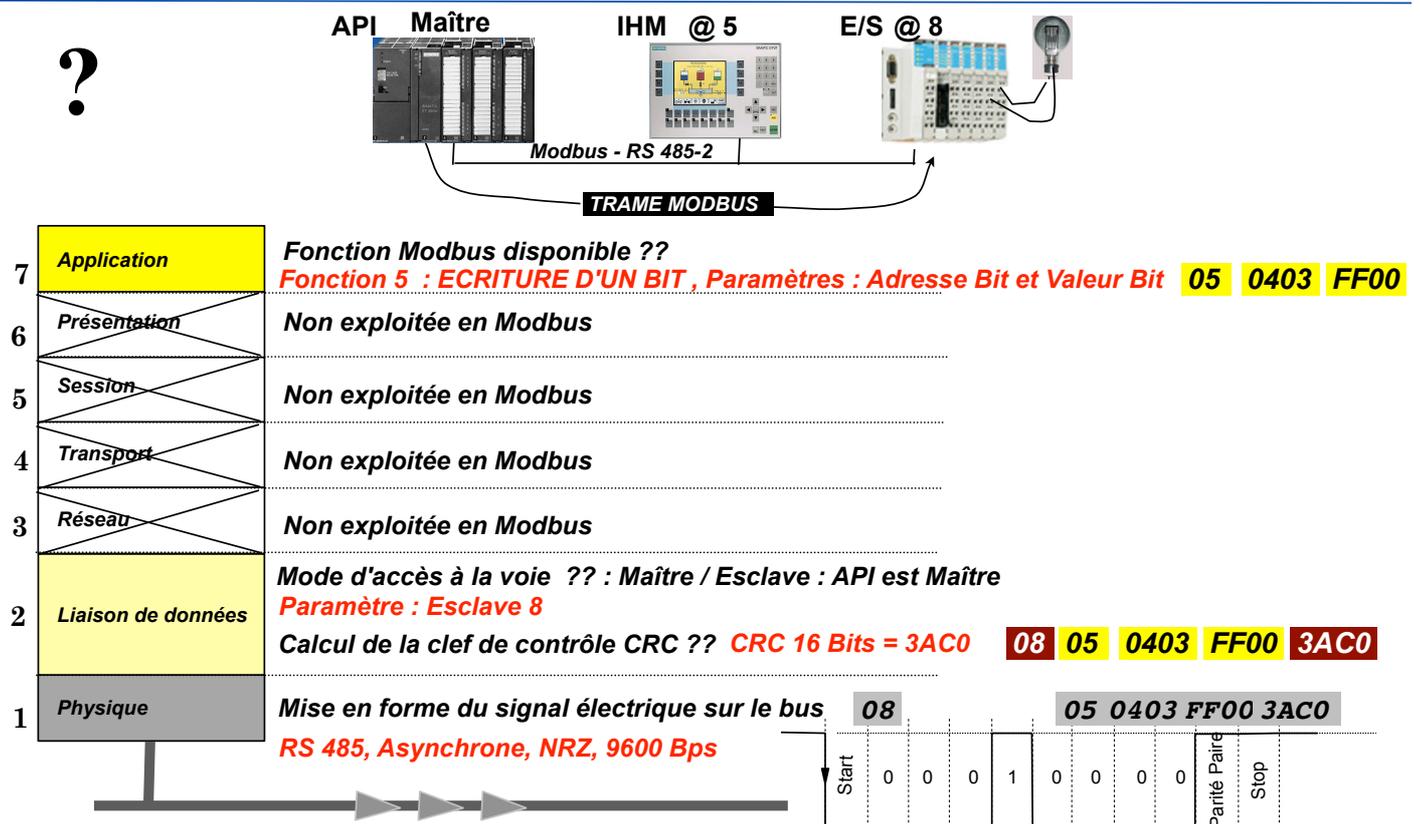
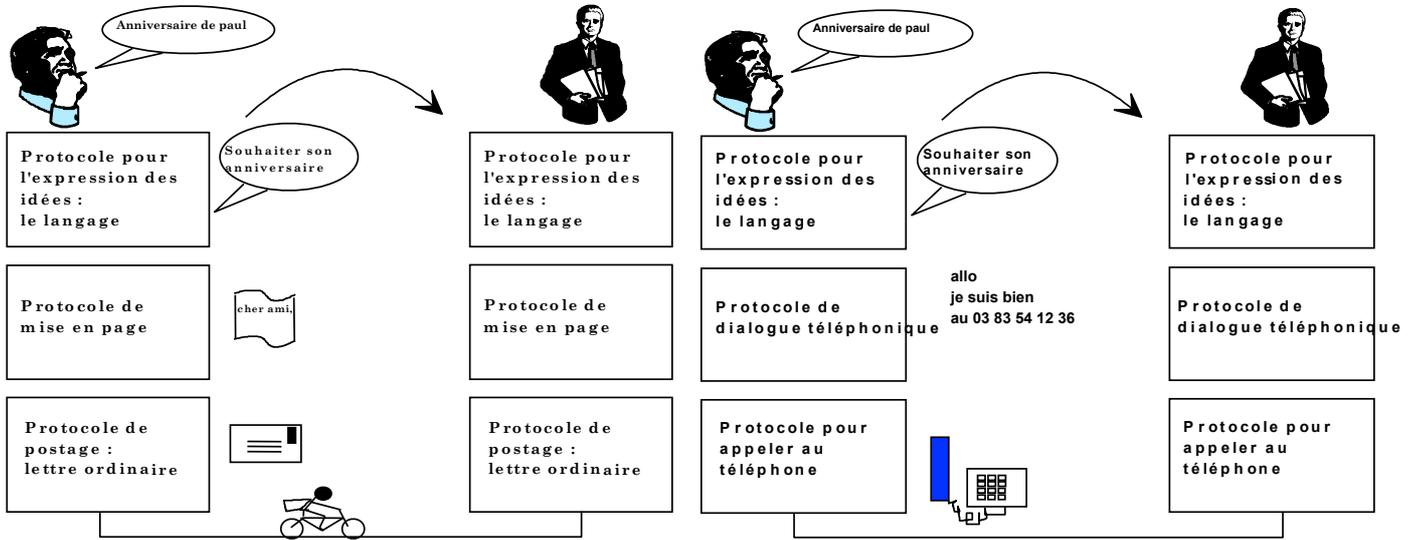
Le Modèle OSI : Open System Interconnection

Le modèle **OSI** (Open System Interconnection) développé en 1979 par l' **ISO** (International Standard Organisation) a pour objectif de déterminer un standard pour la description, la spécification et le développement des systèmes de communication ouverts.

↳ C'est un **modèle de référence** en réseaux permettant l'interconnexion des machines communicantes.



Tout être humain est station d'un réseau mondial car il peut communiquer avec tout le monde. Voyons comment



Analogie du Modèle OSI pour Modbus

API Maître

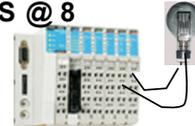


Allumer la lampe depuis l'API avec le réseau MODBUS

IHM @ 5

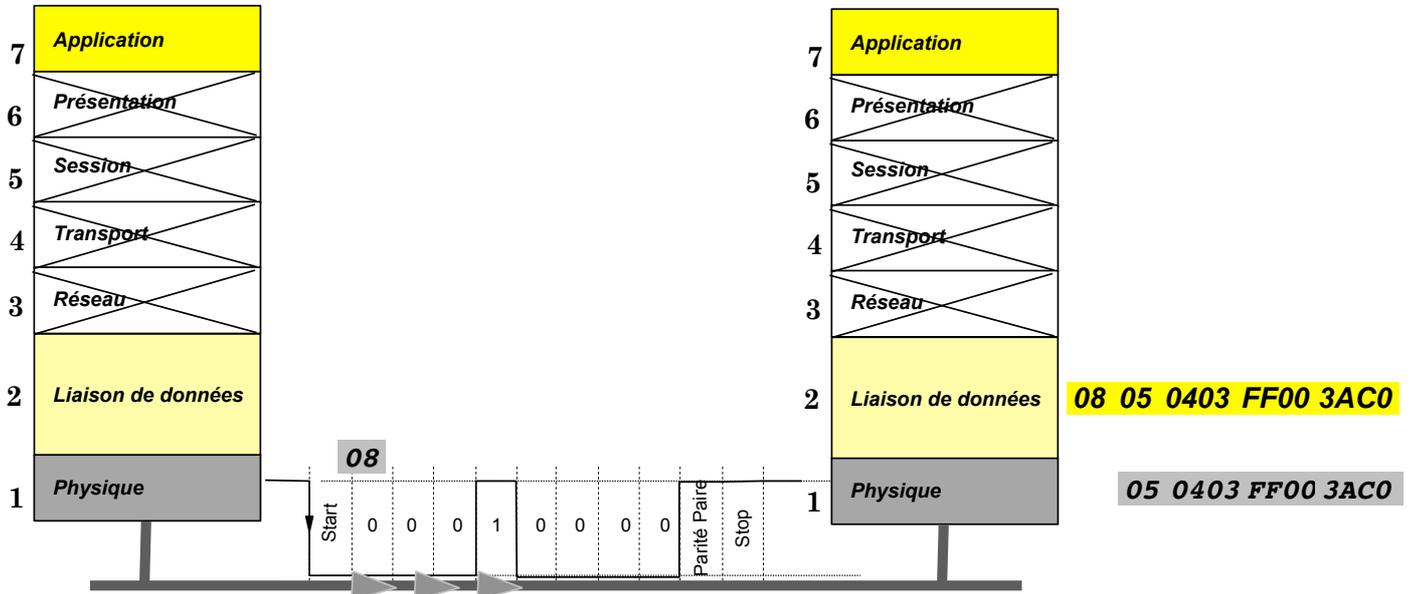


E/S @ 8



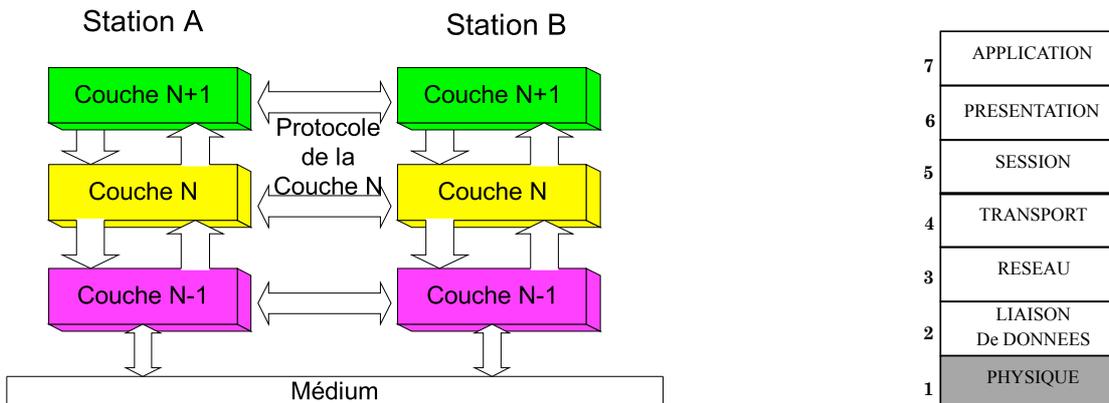
la lampe est raccordée à une sortie TOR (Bit N°3 mot 0x40)

Modbus - RS 485-2



Avantages de la structuration en couches

Modèle de communication en Couches : **modularité** et **simplicité des interfaces d'échanges**.

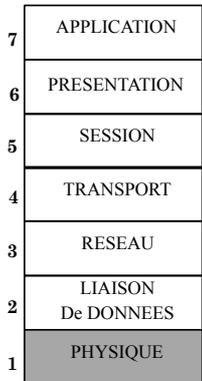


❶ La couche N du système A ne connaît que les interfaces :

⇒ avec les couches (N - 1) et (N + 1) adjacentes, d'où une limitation des interfaces nécessaires.

⇒ avec la couche N du système B.

Les Couches du Modèle OSI



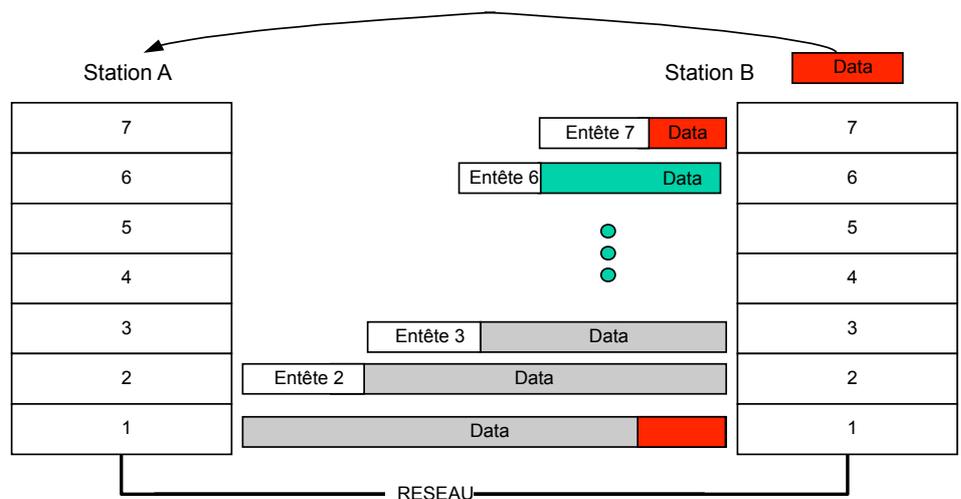
- 7 **Fournit à l'utilisateur** les fonctions qui lui permettent d'exploiter le réseau : transfert de fichier, télécommande, messagerie, navigation hypertexte, accès aux variables mots et bits, etc ...
- 6 **Représentation des données:** cryptage, compression de données, assurer la sécurité et la confidentialité par authentification
Synchronise le dialogue entre les utilisateurs distants.
- 5 Une session est une connexion entre utilisateurs qui nécessite un établissement, un maintien et une clôture de la
- 4 **Garantie la fiabilité du transfert** des données de bout en bout.
Assure une Qualité de Service (QoS) en fonction des possibilités des couches inférieures (débit, niveau de priorité, taux d'erreurs) en négociant les options entre la source et le destinataire.
Découpe et ré-assemble les messages volumineux en paquets et vice-versa.
- 3 **Choisit le chemin optimum** pour acheminer les informations vers le destinataire. Gère l'interconnexion des réseaux lorsque l'émetteur et destinataire ne sont pas sur le même réseau local : ROUTAGE.
Routage par optimisation : trafic, coût, délai, ..., Fragmente les paquets (MTU)
- 2 **S'occupe de la transmission des trames en point à point** entre deux équipements sur un même réseau. Prend en charge la méthode d'accès à la voie et les mécanismes de détections d'erreurs (CRC) et de contrôle de flux, retransmission.
Transmission des Trames point à point sur le médium, Contrôle des erreurs de transmission CRC, Contrôle de Flux – Séquencement, Gestion de la Méthode d'Accès à la Voie
- 1 Gère la **connexion physique des équipements** sur le réseau, ainsi que la réalisation des signaux de transmission sur le Médium. Assure la transmission des bits .
Connexion physique au Médium, Forme des signaux et Codage, Débit - Mode de transmission, Moyens de raccordement Mécanique - Electrique – Optique, Type de Médium

Mécanisme d'Encapsulation

L'empilement des couches entraîne pour chaque couche, l'ajout ou la modification de données à celles qui lui sont fournies par la couche supérieure.

⇒ Ces informations sont ajoutées suivant un mécanisme dit d'"**Encapsulation**".

Les données traitées par la couche 2 sont constituées des données de la couche 3 qui ont été encapsulées par la couche 4.
La couche 2 prendra l'ensemble comme des données sans se soucier de leur contenu déjà traité par la couche 3.



Privilégier une architecture simplifiée à 3 couches

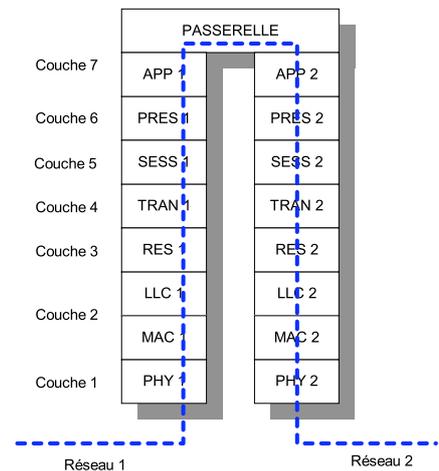
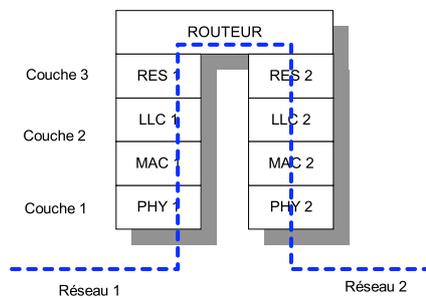
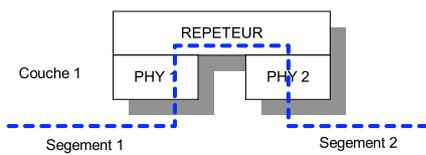
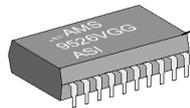
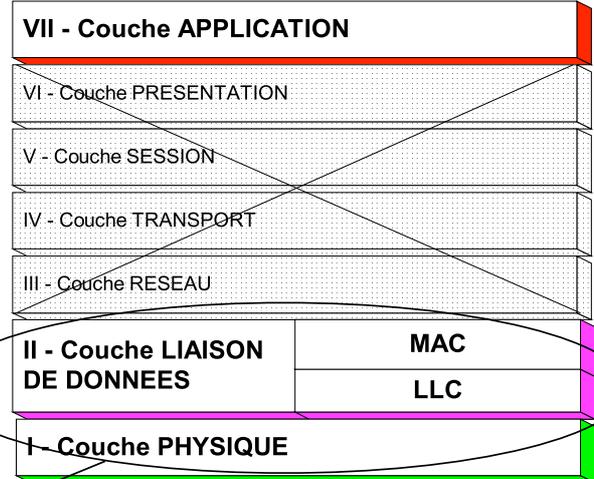
→ Respect des contraintes de Temps Réel :

Rapidité

→ Ne tenir compte que des contraintes industrielles

Réseau local privatif, Efficacité

→ Gestion des couches basses sur semiconducteur



Licence Professionnelle AII-SARI :

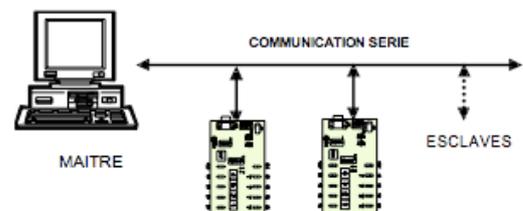
Automatique et Informatique Industrielle Systèmes Automatisés et Réseaux Industriels

LE RESEAU MODBUS

Pr. Eddy BAJIC
IUT Nancy Brabois
Nancy Université

Protocole MODBUS / JBUS

Défini par **Gould Modicon** (Constructeur d'API USA),
1ère mise en œuvre **1979**. C'est donc un vieux protocole industriel.



Il a été adopté depuis par de nombreux constructeurs, et même par tous les constructeurs d'équipement industriel destiné à communiquer. Il permet ainsi de **faire communiquer ensemble des systèmes d'origines et de constructeurs différents**.

La grande caractéristique de MODBUS est qu'il constitue un **standard de communication industrielle** entre API, Régulateur Numérique, Centrale d'Acquisition et de Mesure, etc La majorité des constructeurs ont inscrit à leur catalogue une option, une interface ou un coupleur de communication selon le protocole MODBUS.

Le constructeur APRIL a développé par la suite un protocole très fortement inspiré de MODBUS, et dénommé JBUS. **Le protocole JBUS est compatible MODBUS** (moyennant un simple décalage d'adresse car **JBUS commence ses adresses de données à 1** et non à 0, de plus JBUS intègre un **service périodique** de liaison de données), si bien que l'on parle souvent de protocole **MODBUS/JBUS**.

MODBUS est avant tout un Protocole de communication, donc un ensemble de règles d'échange et de dialogue entre des équipements, et non pas réellement un réseau. Mais par abus de langage, un système de communication mettant en œuvre le protocole Modbus est couramment appelé "réseau Modbus".

Type : Réseau Maître/Esclave de type point à point ou multi-point. 1 maître et de 1 à 63 esclaves

Topologie :

Transmission : Pas de spéc Modbus

- Transmission asynchrone Half-Duplex
- Bande de base de 50 à 19200 Bits/Seconde
- RS 232- V24, RS422 ou RS485

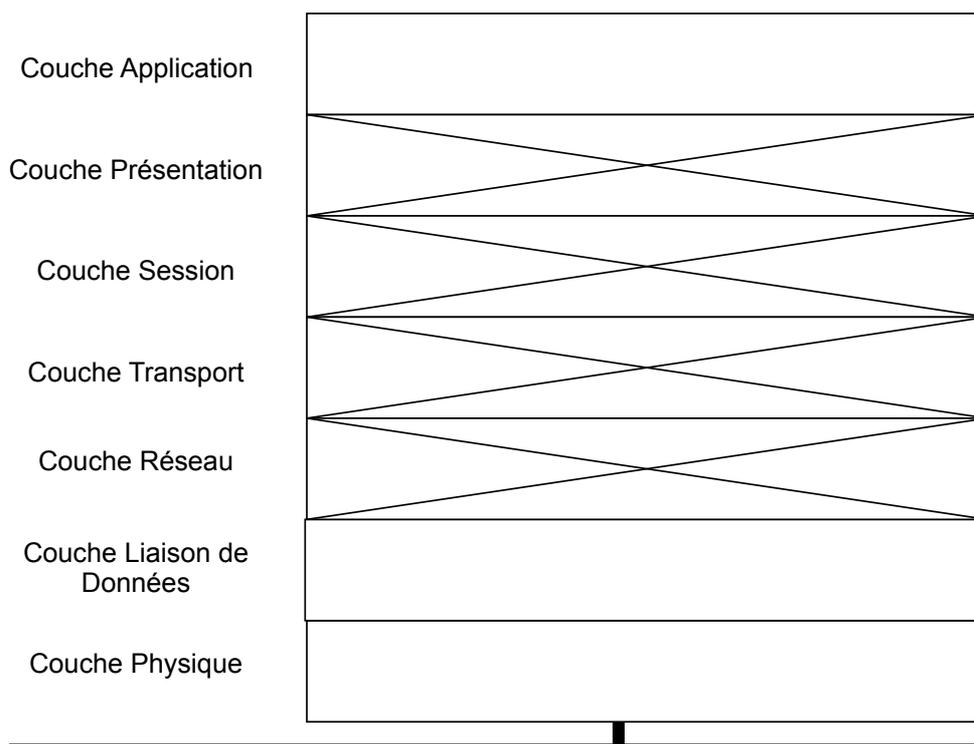
Protocole d'accès :

Fonctions de couche applicaion :

- Ecriture/Lecture de Mots de 16 bits
- Ecriture/Lecture de Bits
- Lecture Rapide d'un Octet Particulier de l'esclave
- * Téléchargement de programme
- * Run/Stop Automate

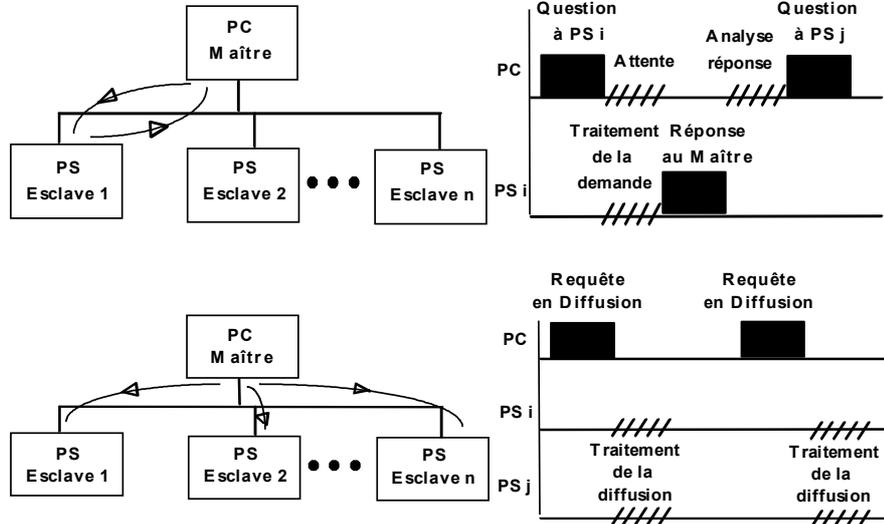
Surveillance : Chaque esclave gère de façon interne des compteurs de diagnostic accessible par le maître, et permettant de surveiller l'état de fonctionnement des coupleurs des esclaves.

MODBUS et le Modèle OSI



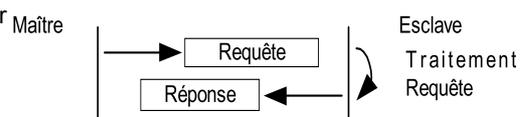
Type **Maître/esclave**, tel qu'il n'existe pas en permanence d'activité sur la ligne.

Seul le Poste Maître a la possibilité de lancer un échange. Un Esclave est toujours passif, il ne fait qu'attendre les requêtes en provenance du Poste Maître.

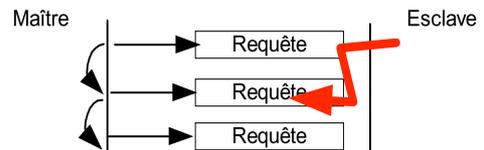


Toute communication horizontale, d'esclave vers esclave, ne peut exister que si le logiciel application du poste maître a été conçu pour recevoir des données et les renvoyer d'un esclave à l'autre.

La station esclave reçoit une demande correcte du maître : Après avoir réalisé la fonction demandée, il répond par une trame de réponse

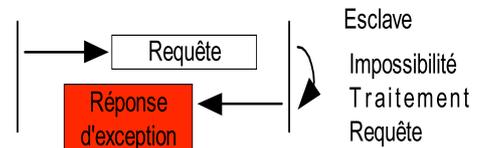


Le maître ne reçoit pas de réponse d'un esclave interrogé : Il attend un temps paramétrable (**Time-Out**), puis ré-émet la question maxi trois fois.

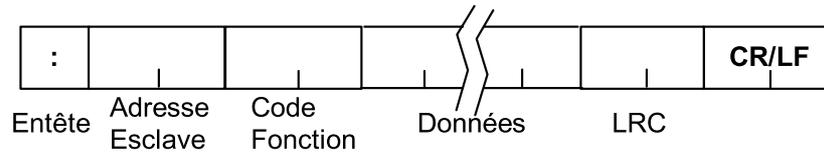


La trame reçue est physiquement erronée : Parasitages, coupures de ligne, défaut de coupleur, ...
L'esclave ne traite pas la trame et ne répond pas.

La trame est logiquement erronée : code fonction demandé non supporté par l'esclave, une adresse inexistante, ...
L'esclave destinataire répond par une **trame d'exception**



Une **trame ASCII** d'une longueur maximale de 525 octets, est composée de six champs distincts



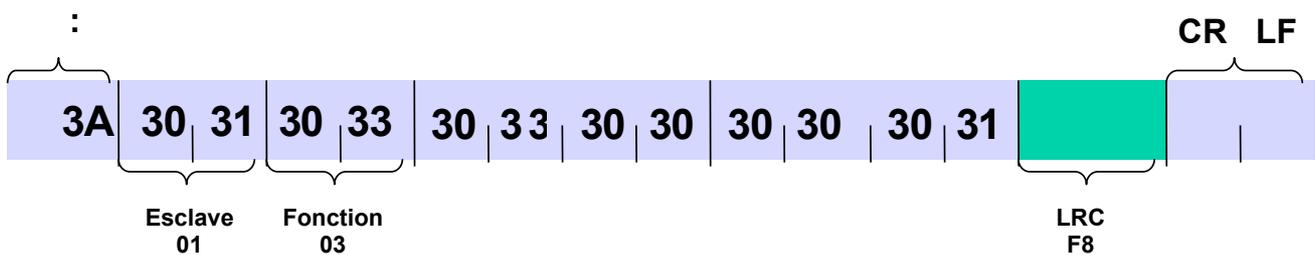
Entête	:Caractère ":" (3Ah) indiquant le début de la trame
Adresse	:Adresse sur deux octets en hexadécimal de l'esclave destinataire
Code Fonction	:Code représentant la fonction demandée à l'esclave
Données	:Paramètres et valeurs associés à la fonction demandée
LRC	:Clef de contrôle de la trame (Longitudinal Redundancy Check sur 8 bits)
Fin	:Caractères "CR" (0Dh) et "LF" (0Ah) délimitant la fin de la trame

Chaque octet est codé sur deux caractères ASCII, chacun contenant quatre bits d'information :

- le premier caractère émis contient le quartet de poids fort,
- le second caractère émis contient le quartet de poids faible

Ainsi, pour coder l'esclave d'adresse 01H, le champ adresse de la trame ASCII comprendra les deux octets 30H (caractère '0') et 31H (caractère '1').

Trame ASCII



Clef de Contrôle LRC 8 - CheckSum

En mode ASCII, la clef de contrôle est appelée **Checksum** ou **LRC8** (Longitudinal Redundancy Check) sur 8 bits. Il est calculé à partir de la somme modulo 255, des octets de la trame sans prendre en compte les délimiteurs ":", "CR", "LF".

Ce LRC est transmis après codage en ASCII, sous la forme de deux octets. Le LRC8 est calculé à partir de la forme binaire de la trame, avant codage ASCII.

unsigned char Calcul_LRC(unsigned char buffer[], unsigned char N) **Algorithme Calcul de Checksum**

TRAME = Tableau d'octets

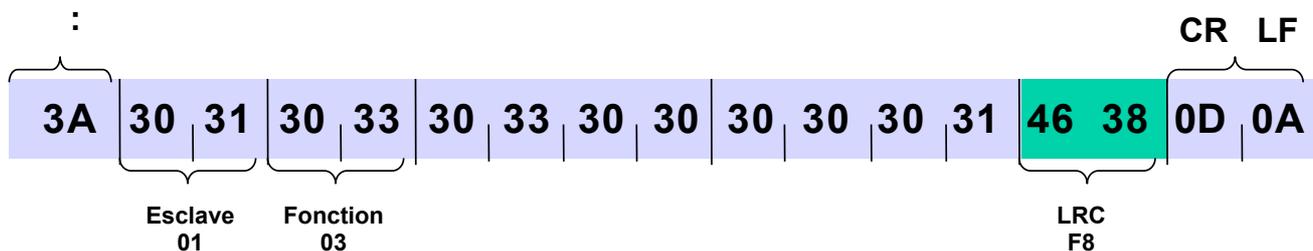
```
// Calcul le LRC sur les N premiers octets
// du tableau buffer[0]..[N-1]
```

```
{
int i;
unsigned char Somme=0; // Clef de contrôle

for (i=0;i<N;i++)
Somme = Somme + buffer[i];
Somme = ~Somme +1;
// Complémentation à deux de LRC
return (Somme);
}
```

Clef de Contrôle LRC 8 - CheckSum

Trame ASCII Modbus

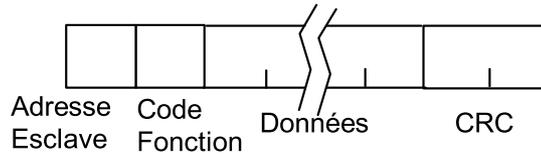


Le LRC est calculé sur les valeurs hexadécimal avant conversion en ASCII, en excluant l'entête et les délimiteurs de fin : Somme Modulo 256 de tous les octets de la trame sauf les délimiteurs, puis complémentée à 2.

LRC =

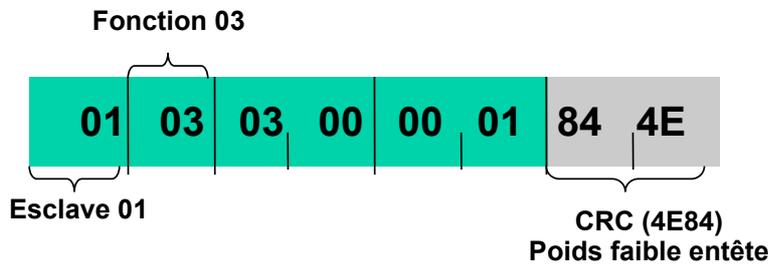
Format RTU des Trames MODBUS

Une **trame RTU d'une longueur maximale de 261 octets**, est composée de quatre champs distincts.



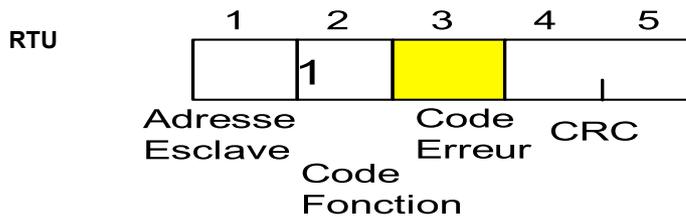
Chaque octet d'information est codé sur un caractère (valeur 00H à FFH).

- Adresse** :
- Code Fonction** :
- Données** : Paramètres et valeurs sur n octets associés à la fonction demandée
- CRC16** : Clef de contrôle de la trame sur deux octets (Cyclic Redundancy Check 16 bits)



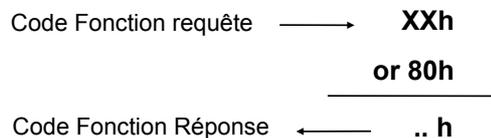
Trames de réponse d'exception MODBUS

L'esclave renvoi un **code d'erreur** sur un octet, signifiant le type d'erreur rencontrée. La signification précise de l'erreur est fonction de l'implémentation MODBUS faite par le constructeur du poste esclave.



Code erreur	Code Fonction erroné	Signification
1	Code Fonction erroné	Fonction non supportée par l'esclave
2	Adresse Incorrecte	Adresse non autorisée sur l'esclave
3	Données Incorrectes	Données non autorisées à l'adresse indiquée
4	Automate Non Prêt	Impossibilité d'échange coupleur avec CPU de l'esclave

L'esclave retourne le **code fonction** émis par le maître en positionnant le bit de poids fort à 1, pour signifier une réponse d'exception.

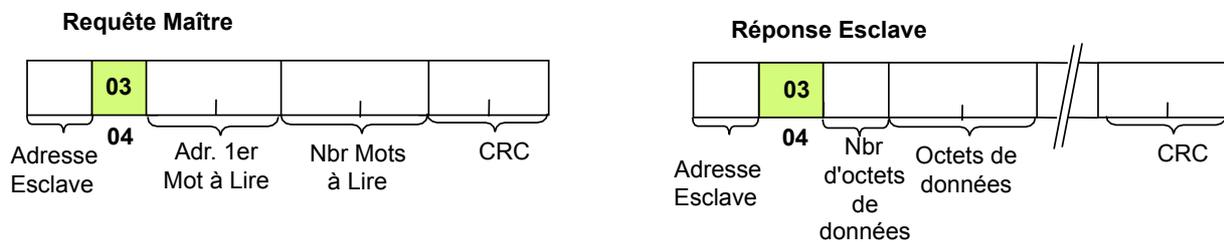


Fonction	Sous Fonction	Données	Fonction demandée
00	00 00	00 00	STOP automate
	00 01	00 00	RUN automate
	00 02	00 00	RUN automate avec initialisation
	00 03	00 XX	Télélecture de programme : XX est le N° de séquence
	00 04	YY XX	Téléchargement de programme dans l'esclave
01			Lecture de N Bits d'Entrée
02			Lecture de N Bits de Sortie
03			Lecture de N Mots d'Entrée
04			Lecture de N Mots de Sortie
05			Ecriture de 1 Bit
06			Ecriture de 1 Mot
07			Lecture Rapide d'un Octet
08	00		Echo : Vérification de la ligne de transmission
	03		Modif du Délimiteur de fin de trame : 0x0A par défaut
	0A		RAZ des compteurs de diagnostic
	0B		Lect du compt 1 Nb trames sans erreur CRC
	0C		Lect du compt 2 Nb trames avec erreur CRC
	0D		Lect du compt 3 Nb trames avec réponse d'exception
	0E		Lect du compt 4 Nb trames adressées à l'esclave
	0F		Lect du compt 5 Nb trames de non réponse (diffusion)
12		Lect du compt 7 Nb de caractères reçus et non traités	
0B			Lecture du compteur d'échange 8
0F			Ecriture de N Bits de sortie
10			Ecriture de N Mots de sortie

Demande de lecture de n mots

Le code fonction 03 correspond à une demande de lecture de N Mots, qui nécessite de spécifier le nombre de mots à lire codé sur 2 octets.

La réponse fournie par l'esclave, comprend quant à elle, un champ stipulant le nombre d'octets de données dans la trame lequel est codé sur un seul octet, puis les mots lus.

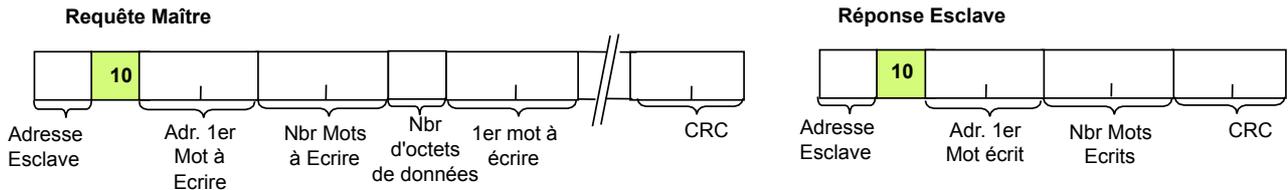


Le nombre de mots à lire en une seule requête est limité selon l'esclave (de 5 à 32 mots)

Demande d'écriture de n mots

Le code fonction 10 correspond à une demande d'écriture de N Mots, qui nécessite de spécifier le nombre de mots à écrire codé sur 2 octets, ainsi que le nombre d'octets de données (nombre de mots * 2), puis le smots à écrire sont rangés à la suite dans la trame poids fort en tête.

La réponse fournie par l'esclave, reprend l'adresse du 1^{er} mots écrit et le nombre de mots écrits.



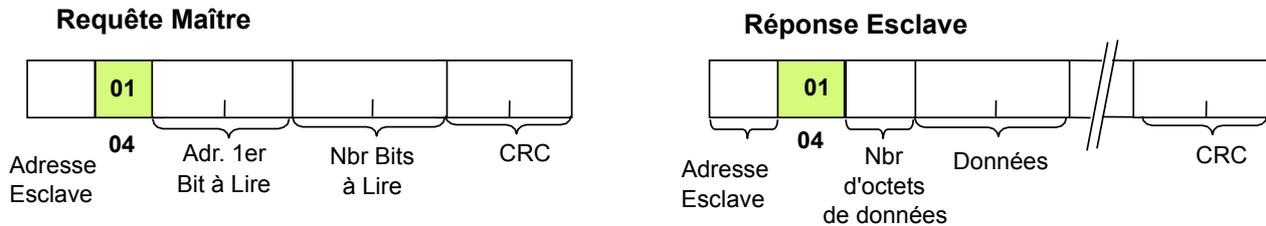
Le nombre de mots à écrire en une seule requête est limité selon l'esclave (de 5 à 32 mots)

Demande de lecture n bits

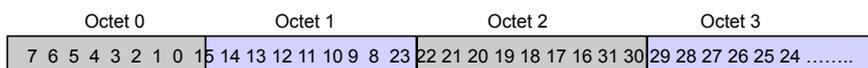
Le code fonction 01 correspond à une demande de lecture de N Bits, qui nécessite de spécifier le nombre de bits à lire codé sur 2 octets.

La réponse fournie par l'esclave, comprend quant à elle, un champ stipulant le nombre d'octets de données dans la trame, lequel est codé sur un seul octet, puis les mots lus.

Le bit de poids faible du premier octet de données de la réponse contient le premier bit lu, puis les bits sont rangés en commençant du poids faible de chaque octet de données de la réponse.



Les bits lus sont rangés selon l'ordre suivant

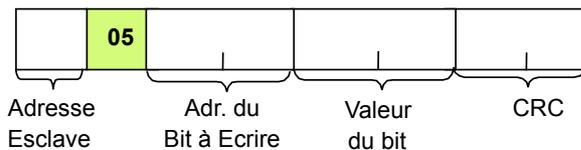


Demande d'écriture d'un bit

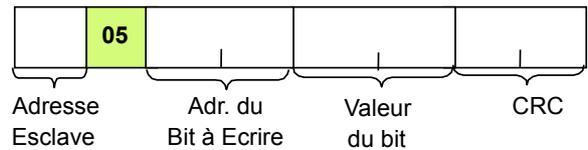
Le forçage d'un bit est effectué par la fonction 03, pour laquelle la valeur du bit est codée sur 1 mot de 16 bits : FF 00 le bit est forcé à 1, 00 00 le bit est forcé à 0.

La réponse de l'esclave est identique à la question du maître.

Requête Maître



Réponse Esclave



La valeur du bit est codée sur 1 mot :

Bit à 1 : **FF00**

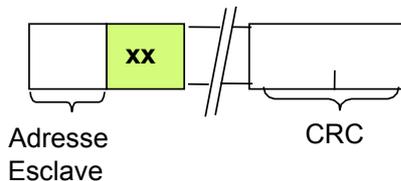
Bit à 0 : **0000**

Réponse d'Exception

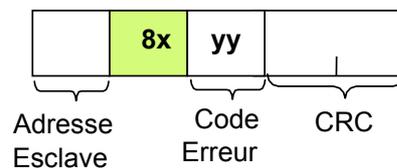
La détermine une réponse d'exception de type d'erreur 1, signifiant un code fonction demandé par le maître, erroné car non supporté par l'esclave.

On remarquera le code fonction renvoyé par l'esclave est celui émis par le maître auquel le bit de poids fort est forcé à 1 : 0x00 or 0x80 = 0x80

Requête Maître



Réponse d'Exception de l'Esclave



Code erreur		Signification
1	Code Fonction erroné	Fonction non supportée par l'esclave
2	Adresse Incorrecte	Adresse non autorisée sur l'esclave
3	Données Incorrectes	Données non autorisées à l'adresse indiquée
4	Automate Non Prêt	Impossibilité d'échange coupleur avec CPU l'esclave

Demande de lecture rapide d'un octet

Le code fonction 07 correspond à une demande d'écriture d'un octet spécifique de l'esclave. En effet, chaque implémentation esclave de Modbus, prévoit un octet accessible grâce à cette fonction, sans en connaître l'adresse en mémoire esclave.

Cette fonction est aussi appelée "lecture rapide d'octet", et correspond sur une implémentation Modbus esclave sur automate programmable, à l'accès à un octet spécifique, sans synchronisation avec le cycle de rafraîchissement des entrées-sorties du moniteur automate, donc dans un temps plus rapide.



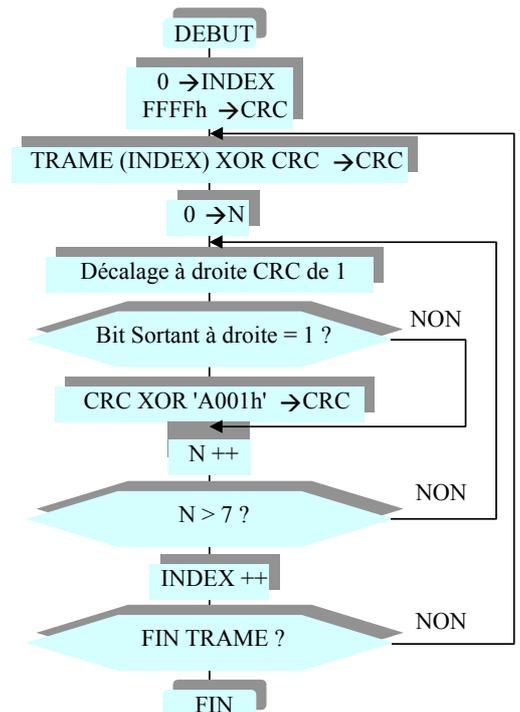
Algorithme de calcul de Clef de Contrôle CRC 16

```

unsigned short Calcul_CRC(byte buffer[], unsigned char N)
// Calcul le CRC16 sur les N premiers octets
// du tableau buffer[0]..[N-1]
{
const polynome_generateur = 0xA001;
//Polynôme générateur du CRC

int i,j,bit;
unsigned short CRC=0xFFFF; // Clef de contrôle

for (i=0;i<N;i++)
{
CRC^=buffer[i];
for (j=0;j<8;j++)
{
bit=CRC & 0x01; // Mémorisation bit N°0
CRC>>=1; // Décalage arithmétique a droite de 1
CRC&=0x7FFF; //Force à 0 le bit entré a gauche
//pour obtenir un décalage logique
if (bit)CRC^=(unsigned short)polynome_generateur;
}
}
return (CRC);
}
    
```



Le polynôme générateur spécifié par le protocole Modbus est $A001h = 2^{15} + 2^{13} + 2^0$:

Automatique et Informatique Industrielle Systèmes Automatisés et Réseaux Industriels

Câblage Réseaux de terrain

Pr. Eddy BAJIC
IUT Nancy Brabois
Nancy Université

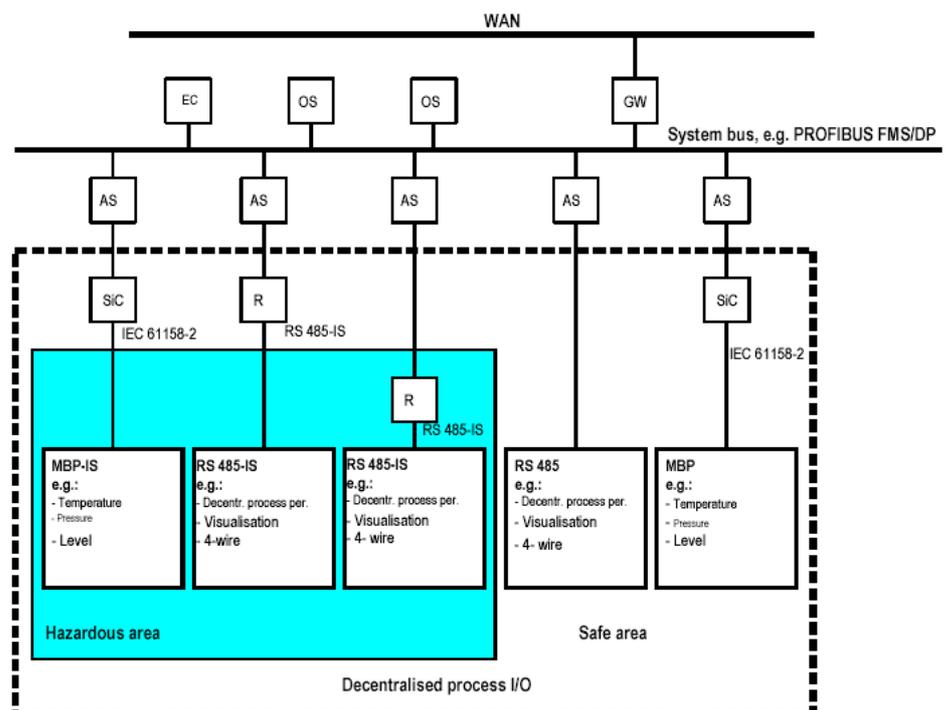
Câblage pour les Réseaux de Terrain

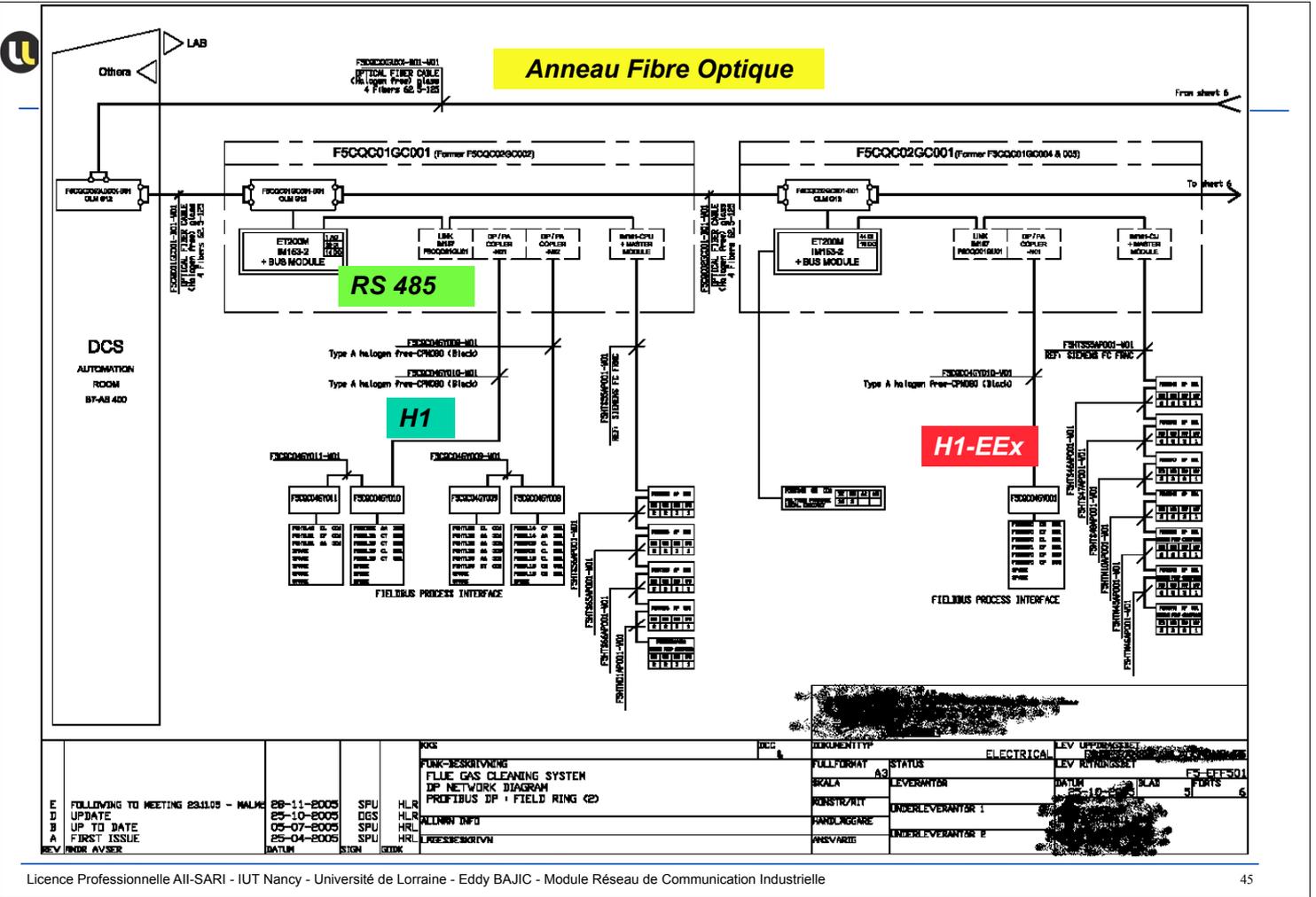
Environnement bruyé, Immunité aux parasites :

Télé-Alimentation par le Bus :

Environnement Explosif : **Limitation courant , barrière à sécurité intrinsèque**

OS: Operating Station
AS: Automation System
EC: Engineering console
GW: Gateway
SiC: Signal Coupler
R: Fieldbus Isolating Repeater
WAN: Wide Area Network
MBP: Manchester coded Bus Powered





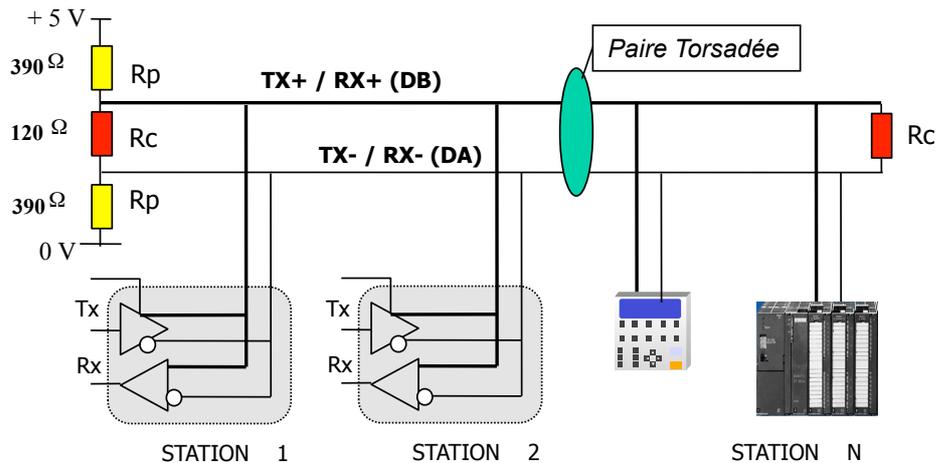
La Liaison RS 485

Caractéristiques :

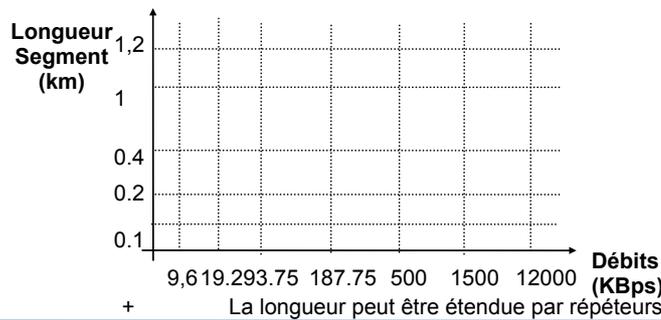
Les signaux différentiels RS485 sont gérés par des circuits spécialisés dits "Drivers RS485 »

Le Bus RS 485

Câblage Simple
Identique pour chaque station

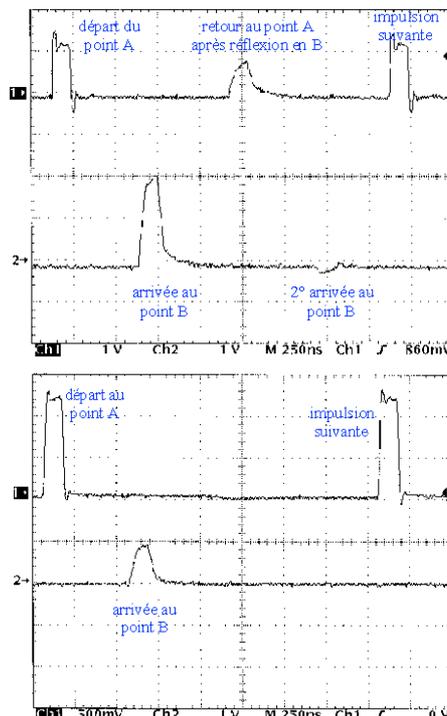
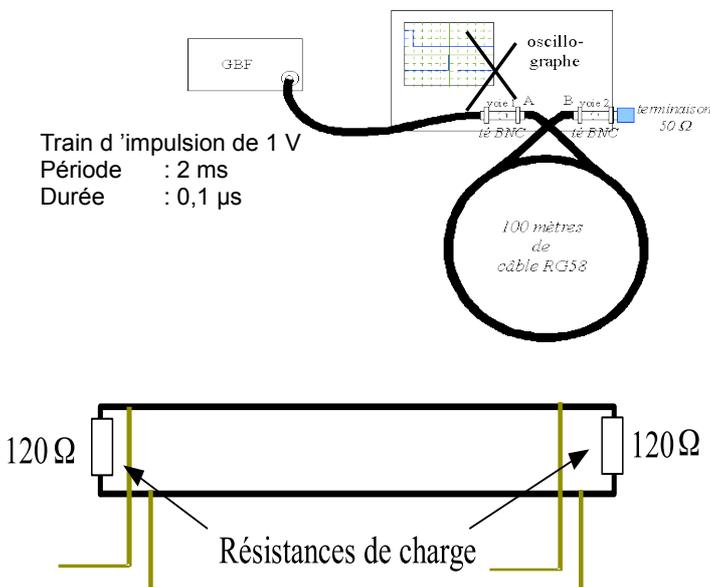


Relation Débit / Distance



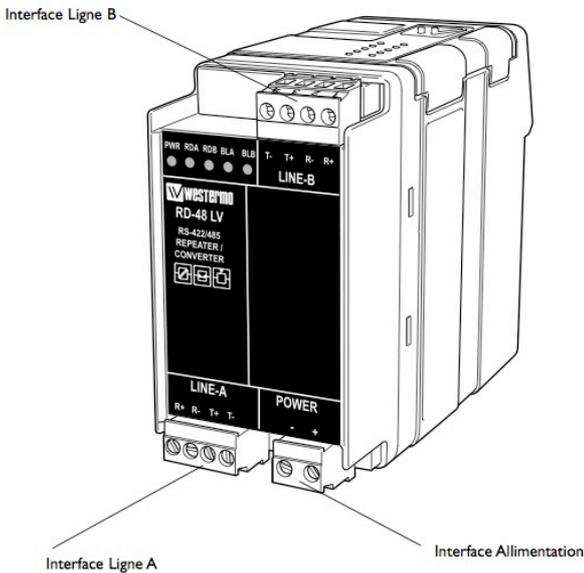
Adaptation de Fin de Ligne de transmission

Quand un signal circule sur un câble et rencontre une discontinuité en fin de ligne, il se produit une **réflexion de signal**. Une signal réfléchi circule en sens opposé, se superpose et produit une perturbation, un bruit qui provoque une distorsion de la ligne.

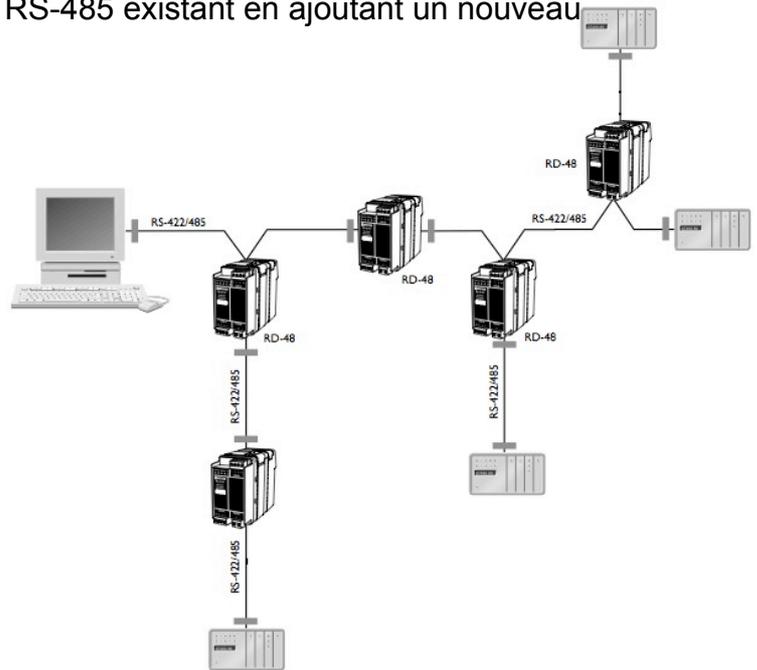


Répéteur RS485

Un répéteur RS485 permet d'étendre un bus RS-485 existant en ajoutant un nouveau segment de 1200 m /31 équipements



Répéteur RS485 Westermo RD48



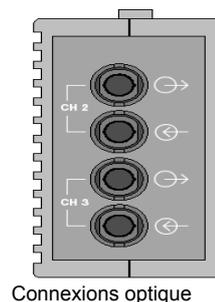
Dans les Bus RS485 la topologie en étoile n'est pas autorisée. L'utilisation d'un répéteur RS 485 permet de construire une topologie en étoile en créant des branches jusqu'à 1200m

Câblage Fibre Optique / RS 485

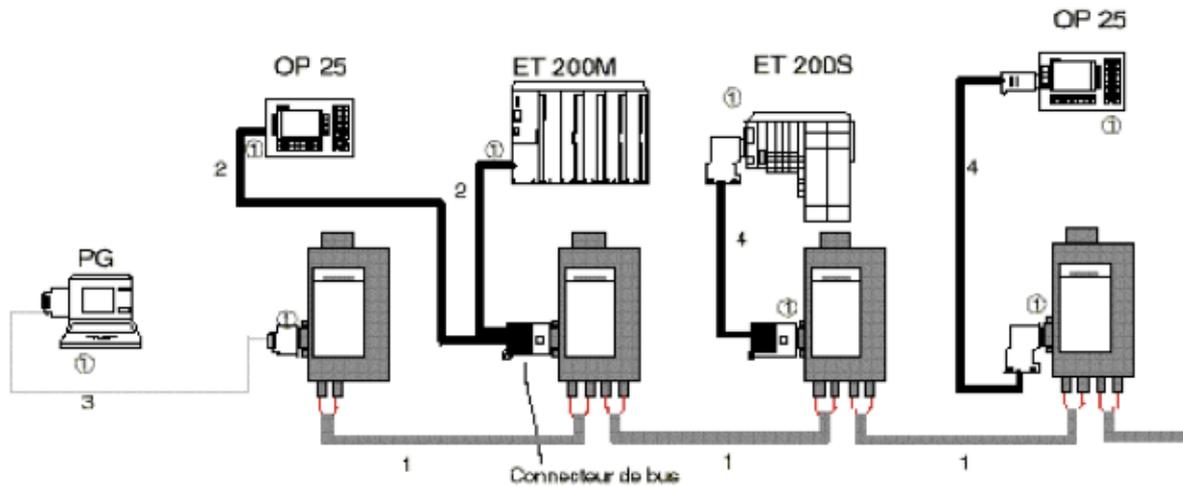
Les Modules liaison Optique dit OLM, sont des composants réseau qui convertissent des signaux cuivre en signaux optiques

- Extension de réseau RS 485 par Fibre Optique
- Distance jusqu'a 15km entre OLM
- Immunité totale aux parasites EM
- Topologie possible en Anneau

Type fibre	Distance
Plastique 900/1000µm	80 m
PCF 200/230µm	400 m
Verre 10/1250µm	3 à 15 km



Exemple de topologie linéaire avec OLM



① Résistance de terminaison activée

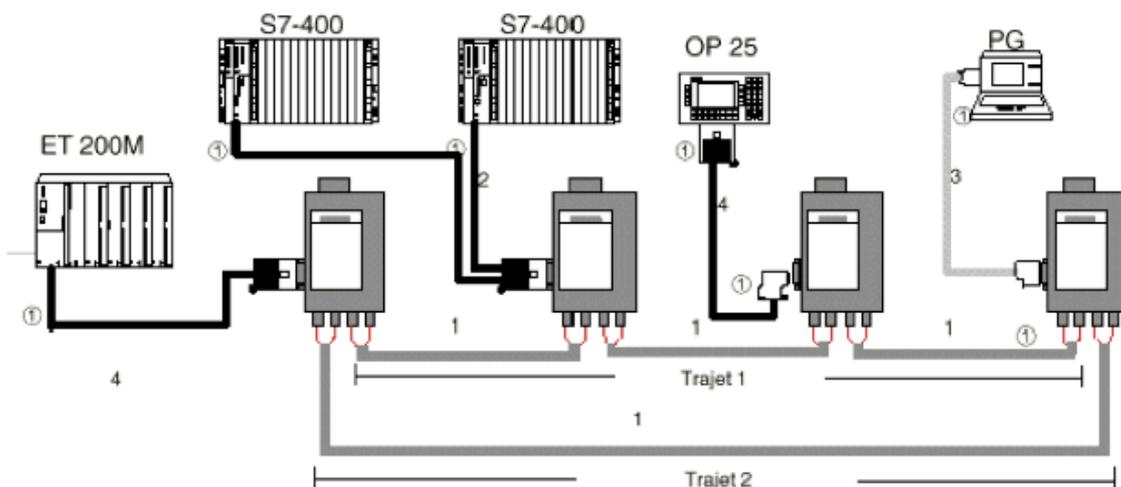
1 Câble FO

2 Câble-bus PROFIBUS

3 Câble de liaison PROFIBUS 830-1 T

4 Câble de liaison PROFIBUS 830-2

Les anneaux optiques redondants sont une forme particulière de la topologie linéaire. Le « bouclage » de la ligne optique en un anneau confère au réseau une grande sécurité de fonctionnement.



Monitoring et Segmentation du Bus optique :

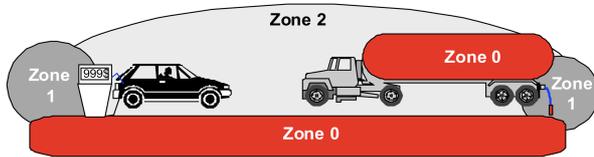
les OLM envoient des messages d'écho pour sonder la liaison optique

Si un OLM est détecté défaillant : le lien est interrompu et le bus est segmenté en deux parties indépendantes toujours opérationnelles

Zone **ATEX** : ATmosphère EXplosive

Norme CEI 61158-2 : répondre aux exigences de sécurité de la communication en zone explosive

FISCO : *Fieldbus Intrinsic Safety CO*ncept

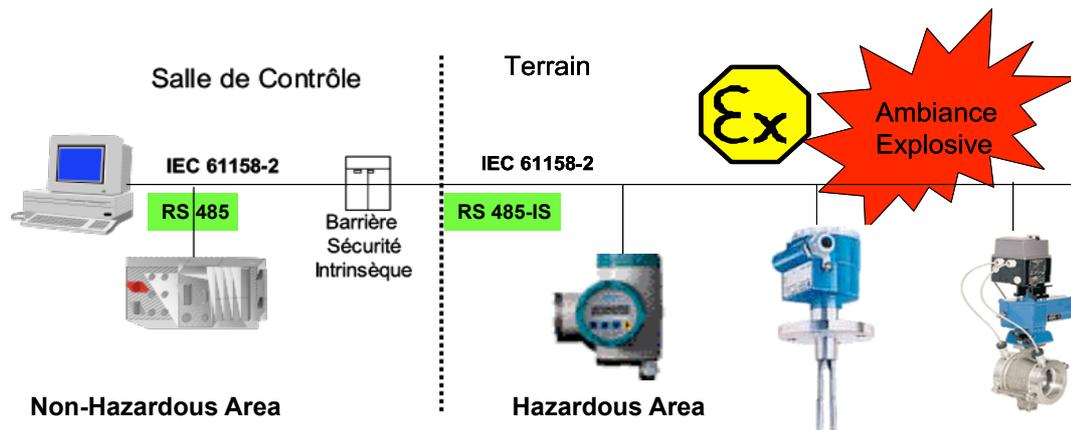


Zone 0 :

Zone 1 :

Classification des zones de sécurité explosives

Zone 2 :



Le Bus H1 : IEC 61158-2

IEC 61158-2, *Digital data communications for measurement and control - Fieldbus for use in industrial control systems - Part 2: Physical layer specification and service definition*

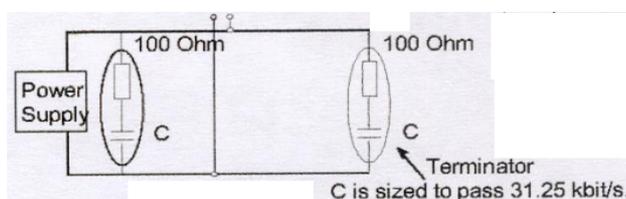
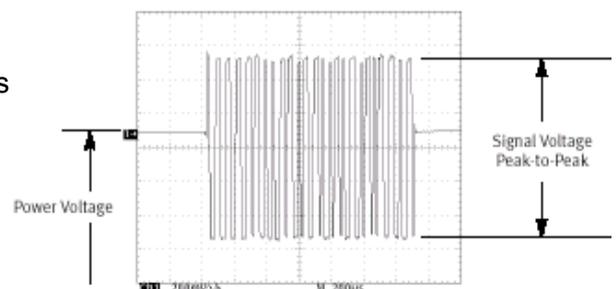
- **transmission sur courant porteur 9 – 32 V dc.**

(Signal numérique et puissance sur même câble + Orange / - Bleu)

- exploités en **ambiances explosives (EExi)**. (exigences des industries chimiques, pétrochimiques et pharmaceutiques)

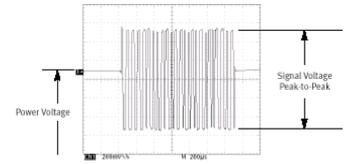
- Débit **31,25 KBps**, **Transmission synchrone**

- Termineur de Bus de type RC ($R=110 \Omega$, $C = 1\mu F$), nécessité par la transmission de puissance sur le bus.



Transmission par Variation de Charge

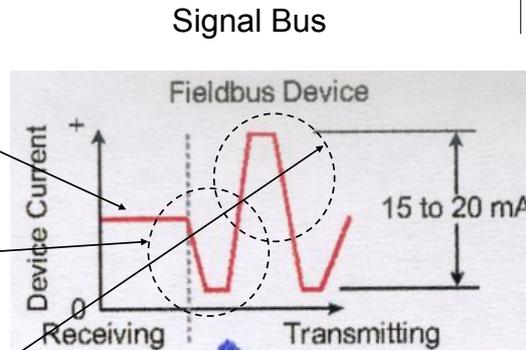
Chaque nœud réseau tire son alimentation nominale du bus (9V Min)
 +
 un courant de 10mA "gaspillé" par une charge interne variable 50 Ohms.



Tension nominale d'alimentation du nœud

Quand le nœud veut transmettre '0', il tire 10mA supplémentaire par une charge supplémentaire de 50 Ohms (signal de ligne descend).

Quand un nœud veut transmettre '1', il coupe les deux charges de 10mA (signal de ligne remonte).

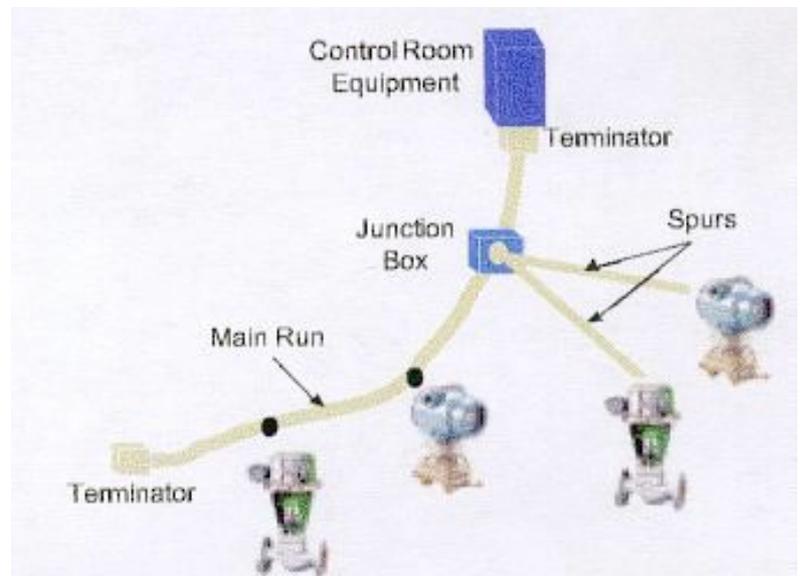


L'équipement fait varier son impédance d'entrée avec 2 charges de 50 Ohms, pour faire varier la tension sur le bus et ainsi transmettre des 0 et des 1 (sa trame réseau)

Topologie Réseau H1 : IEC 61158-2

Topologie classique **bus** ou **Arborescent** avec boîtier de dérivation

- ⇒ Longueur Maxi **1900 m**
- ⇒ **Paire torsadée blindée AWG 18**
- ⇒ **Termineur (RC)** à chaque extrémité
- ⇒ **Max 32 nœuds** dépend de leur consommation de courant, du type de câble, de l'usage de répéteur (+ 4 répéteurs maximum)



Validation de dimensionnement d'un réseau H1 par étude de consommation de courant (Téléalimentation des Nœuds, Résistance câble, ...)

