Sommaire

MODBUS TCP....2

Rappels....2

Architecture du réseau....3

Architecture d'un service de messagerie Modbus TCP....4

Communication TCP....5

Exemple : requête du client....7
Exercice n°1....7

Taille maximale des ADU et PDU....7
Exercice n°2....7

Étude de cas : Système d'Exploitation du Tramway....8
Exercice n°3....11
Exercice n°4....12
Exercice n°5....12
Exercice n°6....12
Exercice n°7....13
Exercice n°8....13
Exercice n°9....14
Exercice n°10....14
Exercice n°11....14

MODBUS TCP

Rappels

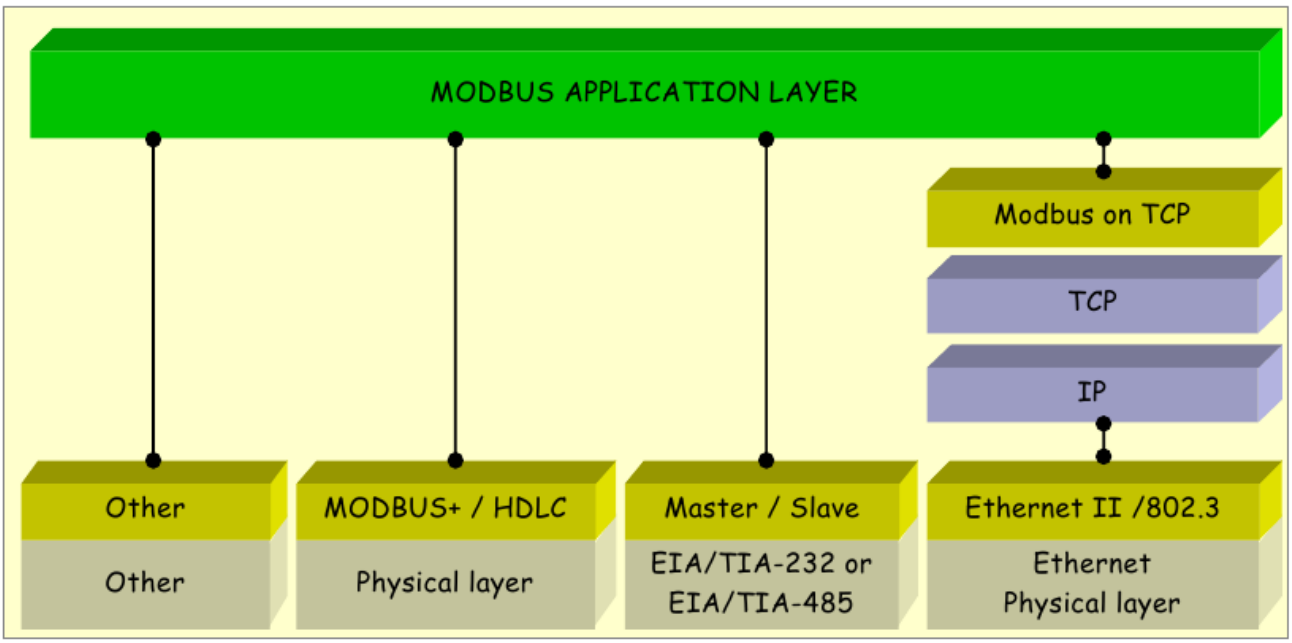
Modbus (marque déposée par Modicon) est un protocole de communication utilisé pour des réseaux d'automates programmables (API). Il fonctionne sur le mode maître / esclave(s). Il est constitué de trames contenant l'adresse de l'automate concerné, la fonction à traiter (écriture, lecture), la donnée et le code de vérification d'erreur appelé contrôle de redondance cyclique sur 16 bits ou CRC16.

Les trames sont de 2 types :

- mode RTU (Remote Terminal Unit) : les données sont sur 8 bits
- mode ASCII : les données sont codées en ASCII (il faut deux caractères pour représenter un octet, exemple 0x03 sera codé '0' et '3')

Le protocole Modbus peut être implémenté :

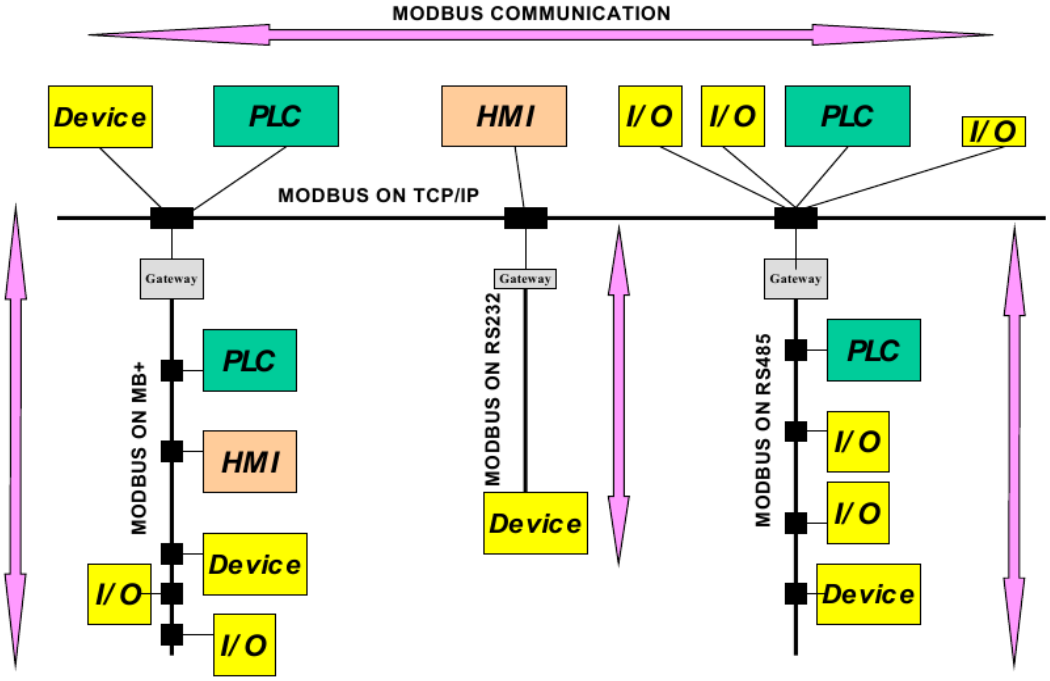
- sur une liaison série asynchrone de type RS-422 ou RS-485 ou TTY (boucle de courant), avec des débits et sur des distances variables : on parle **Modbus série** ou **Modbus maître/esclave** ;
- sur TCP/IP et Ethernet : on parle alors de **Modbus TCP** ;
- sur **Modbus+** : Modbus Plus est un réseau à passage de jetons à 1 Mb/s, pouvant transporter les trames Modbus et d'autres services propre à ce réseau.



Remarque : selon des études récentes, Modbus TCP serait le protocole Ethernet Industriel le plus utilisé au monde. Modbus TCP est la variante "encapsulée" dans TCP/IP du protocole Modbus.

Architecture du réseau

Le protocole MODBUS permet une communication entre toutes ses variantes au sein de toutes les architectures de réseaux :



Légende :

- PLC : *Programmable Logic Controller*
- HMI : *Human Machine Interface* (Interface Homme-Machine ou IHM)
- I/O : *Input/Output* (Entrée/Sortie)
- *Device* : équipement, appareil

Architecture d'un service de messagerie Modbus TCP

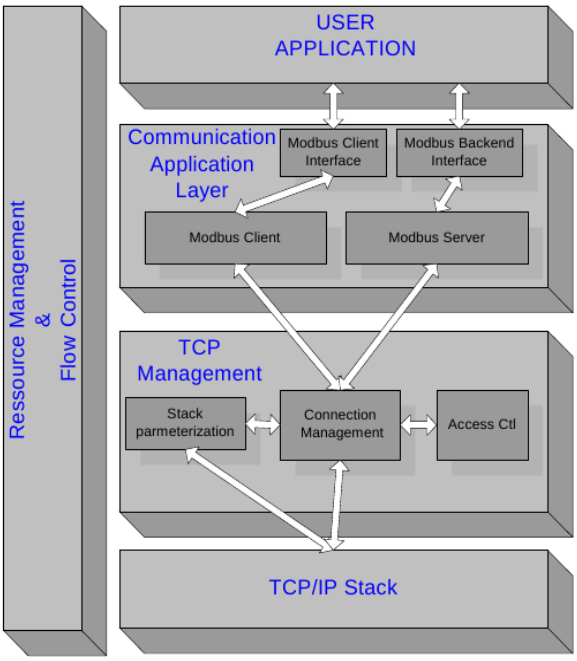
Évidemment la communication Modbus TCP est basée sur le modèle **client/serveur**. Un équipement Modbus peut donc intégrer à la fois un module client et un module serveur, mais cela n'est pas obligatoire. Un équipement peut très bien n'intégrer qu'un seul de ces deux rôles.

Parmi les principales fonctions implémentées par un service de messagerie Modbus TCP figurent l'établissement et la terminaison des communications, ainsi que la gestion des flots de données (contrôle de flux) parcourant les connexions TCP établies.

La communication entre un client et un serveur Modbus requiert la mise en place d'un système de gestion des connexions TCP. Deux options sont envisageables :

- soit c'est l'application qui se charge de cette tâche (programmation par sockets et gestion des mécanismes TCP/IP),
- soit cette gestion est réalisée au travers d'un module dédié, baptisé *TCP Connection Management*, inclus au niveau de la couche *TCP Management* de l'architecture composant Modbus. Dans ce cas, la gestion des connexions devient totalement transparente pour l'application, qui se contente d'envoyer et recevoir les messages Modbus.

Pour permettre l'établissement des connexions et l'échange de données entre équipements, le service de messagerie Modbus TCP doit fournir une socket d'écoute sur le port 502. Il est important de noter que le **port d'écoute 502 TCP** est réservé aux communications Modbus.



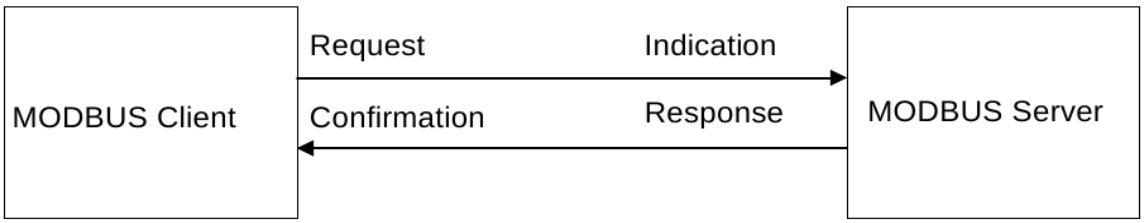
Dans certains contextes critiques, l'accès aux données internes des équipements doit être interdit aux hôtes indésirables. C'est pourquoi un module de contrôle d'accès (*Access Ctl*) peut être implémenté si nécessaire.

Si le nombre de connexions client et serveur est supérieur au nombre de connexions autorisées (de 1 à 16), la plus ancienne connexion non utilisée est fermée. Les mécanismes de contrôle d'accès peuvent par ailleurs être activés pour vérifier que les adresses IP des clients distants sont bien autorisées. Par défaut, lorsque le mode de sécurité est activé, les adresses IP non-configurées par l'utilisateur sont interdites d'accès.

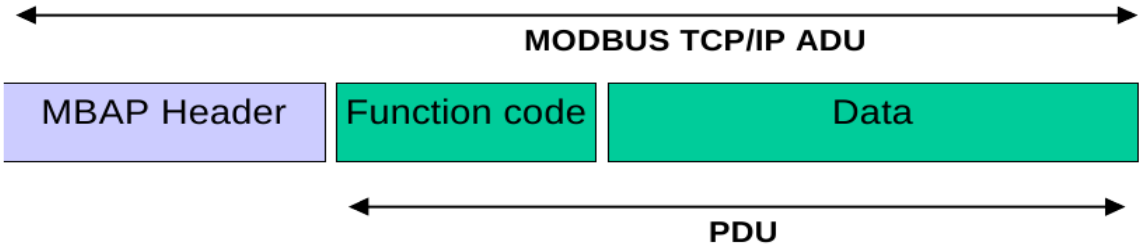
Le module client Modbus construit une requête sur la base des informations transmises par l'application au travers de l'interface client Modbus. Cette interface fournit une API (*Application Programming Interface*) permettant à l'application de construire des requêtes pour accéder à différents services Modbus.

Un module serveur Modbus est, quant à lui, chargé de recevoir les requêtes et de mettre en œuvre des actions (de lecture et d'écriture notamment) afin d'y répondre. L'exécution de ces actions est réalisée de façon totalement transparente pour le programmeur de l'application. Les fonctions principales d'un serveur Modbus sont l'attente de requêtes sur le port 502 TCP, ainsi que le traitement de ces requêtes et la construction de réponses Modbus en fonction du contexte dans lequel se trouve l'équipement. Entre le module serveur Modbus et l'application on peut également trouver une interface baptisée *Modbus Backend*, qui permet un accès indirect aux objets de l'application.

Communication TCP



Le protocole Modbus définit une « unité de données de protocole », ou PDU (*Protocol Data Unit*), indépendante des autres couches de communication. L'encapsulation du protocole Modbus sur TCP/IP introduit un champ supplémentaire (*MBAP Header*) au niveau de l'unité de donnée d'application, ou ADU (*Application Data Unit*).



Le contenu du *MBAP Header* est le suivant :

Fields	Length	Description -	Client	Server
Transaction Identifier	2 Bytes	Identification of a MODBUS Request / Response transaction.	Initialized by the client	Recopied by the server from the received request
Protocol Identifier	2 Bytes	0 = MODBUS protocol	Initialized by the client	Recopied by the server from the received request
Length	2 Bytes	Number of following bytes	Initialized by the client (request)	Initialized by the server (Response)
Unit Identifier	1 Byte	Identification of a remote slave connected on a serial line or on other buses.	Initialized by the client	Recopied by the server from the received request

Description d'un échange :
C'est le client (qui initie la transaction Modbus) qui construit l'ADU. Le champ Function Code du PDU indique au serveur le type d'action à mener.

A la réception d'une requête du client, le serveur analyse l'entête MBAP de l'ADU Modbus. Si celle-ci correspond bien à une entête Modbus, une transaction Modbus est instanciée.

Dans le cas où le nombre maximum de transactions simultanée autorisé est dépassé, le serveur construit une réponse d'exception (Exception Code 6 : Server Busy).

Dans le cas contraire, une transaction est instanciée et initialisée avec les informations suivantes : l'identifiant de la connexion TCP utilisée (fournie par le module TCP Management), l'ID de la transaction Modbus (contenu dans l'entête MBAP) et l'Unit Identifier (contenu également dans l'entête MBAP). Ensuite, c'est au tour du champ PDU d'être analysé. Si celui-ci est reconnu comme un champ valide, le serveur est prêt à exécuter le service demandé.

Une fois que la requête a été traitée, le serveur Modbus construit une réponse qu'il doit ensuite envoyer au module TCP Management. En fonction du résultat du traitement, deux types de réponses sont envisageables : une réponse positive ou une réponse d'exception, dont l'objectif est de fournir au client des informations concernant les erreurs détectées durant le traitement de la requête. Le PDU de la réponse Modbus doit être préfixé d'une entête MBAP construite à partir des informations mémorisées lors de la réception de la requête : Unit Identifier, Protocol Identifier et Transaction Identifier. Le serveur renseigne également le champ « Length », indiquant la longueur du champs PDU+Unit Identifier. Enfin, la réponse Modbus est envoyée au client.

A la réception de la réponse, le client Modbus utilise le Transaction Identifier pour retrouver la requête correspondante, précédemment envoyée sur la connexion TCP. Si celui-ci ne correspond à aucune requête connue du client, la réponse est ignorée et mise au rebut. Dans le cas contraire, la réponse est analysée et utilisée par le client Modbus pour construire la confirmation qui sera ensuite envoyée à l'application.

L'analyse de la réponse consiste à vérifier l'entête MBAP ainsi que le PDU de la réponse Modbus. Si celle-ci provient d'un serveur Modbus directement connecté au réseau TCP/IP, l'identification de la connexion TCP utilisée est suffisante pour identifier le serveur distant sans aucune ambiguïté. Dans ce cas l'Unit Identifier n'est pas significatif et doit être ignoré. Par contre, si le serveur est connecté à un sous-réseau sur liaison série, et que la réponse provient d'un routeur, l'Unit Identifier (qui dans ce cas ne sera pas égal à 0xFF) devra être pris en compte pour l'identification du serveur par le client. Pour ce qui est de l'analyse du PDU de la réponse, le client vérifie la conformité du Function Code ainsi que du format de la réponse. Si le Function Code est le même que celui contenu dans la requête, et si le format de la réponse est correct, le client envoie une confirmation positive à l'application. Si au contraire le Function Code contenu dans la réponse est différent de celui contenu dans la requête originale, ou si le format de la réponse est incorrect, le client signale une erreur à l'application en lui envoyant une confirmation négative. Enfin, si le Function Code correspond à un code d'exception (Function Code + 80H), une réponse d'exception Modbus est transmise à l'application.

Il est important de noter qu'une confirmation positive indique seulement à l'application que le serveur a bien reçu la requête et qu'il y a répondu. Une telle confirmation ne la renseigne en aucun cas sur le succès ou l'échec des actions menées, ceci étant du ressort des réponses d'Exception Modbus.

Exception Code	MODBUS name	Comments
01	Illegal Function Code	The function code is unknown by the server
02	Illegal Data Address	Dependant on the request
03	Illegal Data Value	Dependant on the request
04	Server Failure	The server failed during the execution
05	Acknowledge	The server accepted the service invocation but the service requires a relatively long time to execute. The server therefore returns only an acknowledgement of the service invocation receipt.
06	Server Busy	The server was unable to accept the MB Request PDU. The client application has the responsibility of deciding if and when to re-send the request.
0A	Gateway problem	Gateway paths not available.
0B	Gateway problem	The targeted device failed to respond. The gateway generates this exception

Exemple : requête du client

Soit une requête Modbus ADU envoyée par le client (en héxadécimal) :

15	01	00	00	00	06	FF	03	00	04	00	01

Exercice n°1

Décoder seulement l'en-tête *MBAP Header* de cette requête en complétant le tableau suivant.
Réponse :

Champs	**Valeur**	**Signification/Décodage**

Remarques :
Le champ "Unit Identifier" est utilisé pour le routage lorsqu'on s'adresse à un périphérique sur un réseau Modbus+ ou Modbus série. Dans ce cas, le champ "Unit Identifier" contient l'adresse esclave de l'appareil distant. Si le serveur MODBUS est relié à un réseau Modbus+ ou Modbus série par un pont ou une passerelle, le champ "Unit Identifier" est nécessaire pour identifier le dispositif esclave connecté sur le sous-réseau derrière le pont ou la passerelle. L'adresse IP de destination identifie le pont lui-même et non l'esclave. Les adresses MODBUS esclave sur la liaison série sont affectés de 1 à 247 (décimal). L'adresse 0 est utilisé comme adresse de diffusion.

Sur les réseaux TCP/IP, le serveur MODBUS est adressée en utilisant son adresse IP, par conséquent, le champ "Unit Identifier" est inutile. La valeur 0xFF doit être utilisé.

Taille maximale des ADU et PDU

La taille du PDU MODBUS est limitée par la contrainte de taille héritée de la première implémentation du protocole MODBUS mise en oeuvre sur liaison série RS485 : **max. ADU = 256 octets.**

Par conséquence, la taille maximale du MODBUS PDU pour une communication sur liaison de série est : **max. PDU = 256 - Adresse du serveur (1 octet) - le CRC (2 octets) = 253 octets (Function Code + Data).**

Exercice n°2

Déterminer la taille maximale du ADU pour MODBUS TCP.
Réponse :

Étude de cas : Système d'Exploitation du Tramway

Le Système d'Exploitation du Tramway d'une grande ville française est constitué de nombreux sous-systèmes. Ceux qui font l'objet de cette étude sont :

- le SAEIV constitué par le Système d'Aide à l'Exploitation (SAE) et le Système d'Information Voyageur (SIV) ;
- le Réseau Multi Service (RMS).

Le Système d'Aide à l'Exploitation (SAE) désigne globalement un SAE Temps Différé (SAE-TD) et un SAE Temps réel (SAE-TR). C'est un système informatique implanté au Poste de Commande Centralisé (PCC), en liaison avec les Systèmes d'Exploitation Embarqués (SEE) des tramways et Systèmes d'Information Station (SIV Station) des stations.

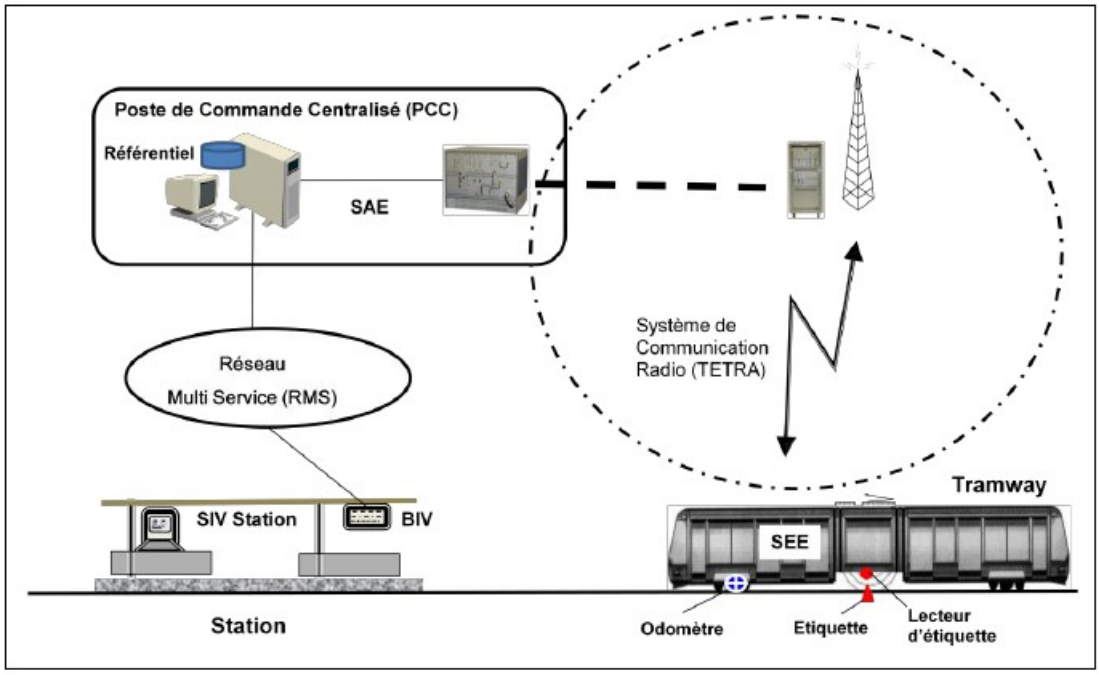


Figure 1 : Organisation du SAEIV et du RMS

Le SAE assiste le personnel exploitant (opérateurs du PCC et conducteurs) dans ses principales missions d'exploitation du réseau de tramway.
Le Système d'Information Voyageur (SIV) désigne de façon commode les moyens d'information à destination des voyageurs en station et à bord des tramways.

Le SIV est composé :

- du Système d'Information Voyageur Station (SIV Station) qui gère les Bornes d'Information Voyageur (BIV) de chaque station ;
- du Système d'Information Voyageur Tramway (SIV Tramway) qui gère l'information des voyageurs à bord des tramways.

Pour l'information des voyageurs aux stations, le serveur de communication SCOM dialogue avec les Bornes d'Information Voyageur (BIV) des stations au travers d'un réseau TCP/IP pour transmettre les messages à diffuser dans chaque station.

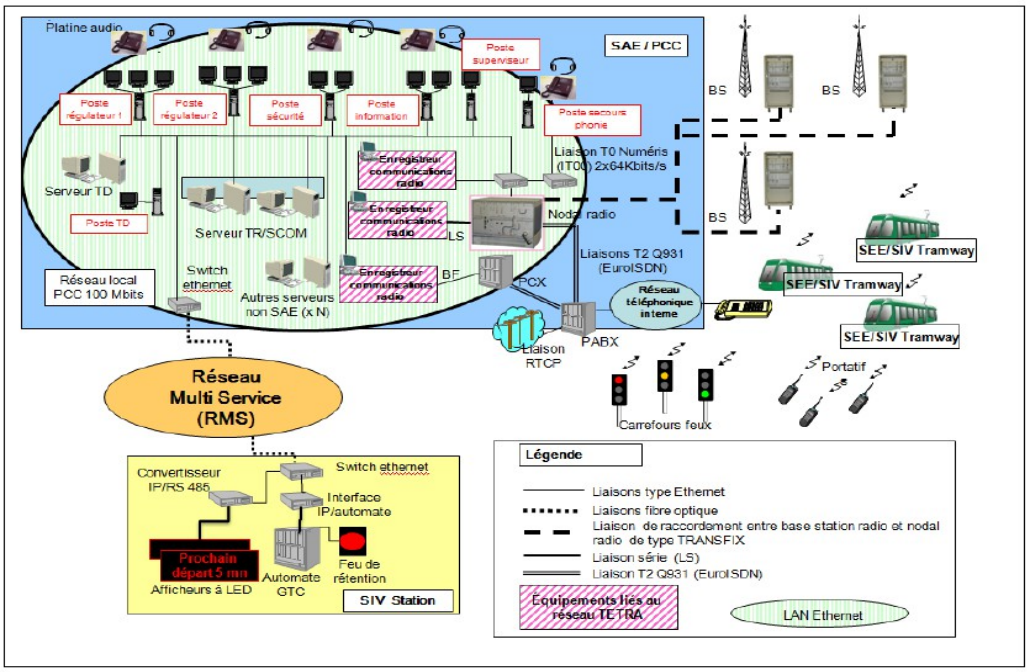
Le Poste de Commande Centralisé (PCC) est composé :

- d'un système d'aide à l'exploitation en Temps Réel (SAE-TR) en relation avec le Système d'Exploitation Embarqué (SEE) de chaque rame ;
- d'un système d'aide à l'exploitation en Temps Différé (SAE-TD) en relation avec le Système d'Information Station (SIV Station) de chaque station de voyageurs ;
- d'un Système de COMmunication (SCOM) qui dialogue avec les Bornes d'Information Voyageur (BIV) du SIV d'une station ;
- d'un Système Radio Numérique (TETRA) qui permet de dialoguer avec le Système d'Exploitation Embarqué (SEE) de chaque rame de tramway.

Le Système d'Exploitation Embarqué (SEE) est composé :

- pour chacune des deux loges d'une rame :
  - d'une Unité Centrale Embarquée (UCE) qui intègre un écran/clavier ;
  - d'un Tiroir de Phonie Embarqué (TPE) ;
  - d'un contrôleur de feux qui permet de demander le passage au rouge des feux de signalisation des carrefours ;
- pour l'ensemble de la rame, d'un Système d'Information Voyageurs Tramway (SIV Tramway) formé :
  - d'un annonceur vocal ;
  - de divers dispositifs d'affichage de messages pour les utilisateurs.

Les informations fournies portent principalement sur les destinations, les arrêts desservis par les rames, et les temps d'attente en station voyageur. La gestion du SIV est effectuée depuis le PCC. Cette gestion centrale dialogue avec des équipements informatiques en station via le Réseau Multi Service (RMS) essentiellement constitué d'un réseau filaire de communications TCP/IP et avec les équipements embarqués à bord des rames via le réseau radio numérique.



Architecture générale du Système d'Aide à l'Exploitation et d'Information des voyageurs (SAEIV)

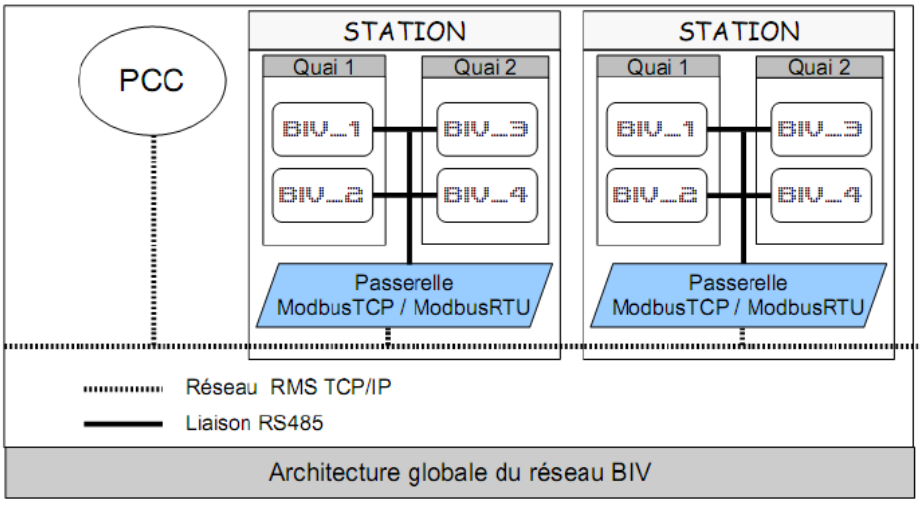


Figure 8 : Architecture globale du réseau SIV Station

Le Système d'Information Voyageurs en Station (SIV Station) repose sur les Bornes d'Information Voyageur (BIV). Le système central du SIV au PCC réalise les fonctions :

- de dialogue avec les équipements informatiques des BIV en stations via le réseau TCP/IP ;
- d'affichage de messages à destination des voyageurs.

Le sous-système Borne d'Information Voyageur (BIV) est un des composants du Système d'Information Voyageurs (SIV). Il est installé dans chaque station constituant les points d'arrêts du réseau de tramway. Sa fonction principale est de renseigner l'utilisateur sur les conditions de desserte des véhicules.

En pratique, les informations diffusées sont la destination, le temps d'attente pour le prochain tramway, l'heure et des informations commerciales du réseau.
Chaque quai de station est équipé de deux afficheurs permettant l'affichage de 4 lignes de 40 caractères, de hauteur 33 mm. La hauteur des caractères et le réglage de la luminosité permettent une bonne lisibilité de jour comme de nuit à une distance de 20 mètres environ.

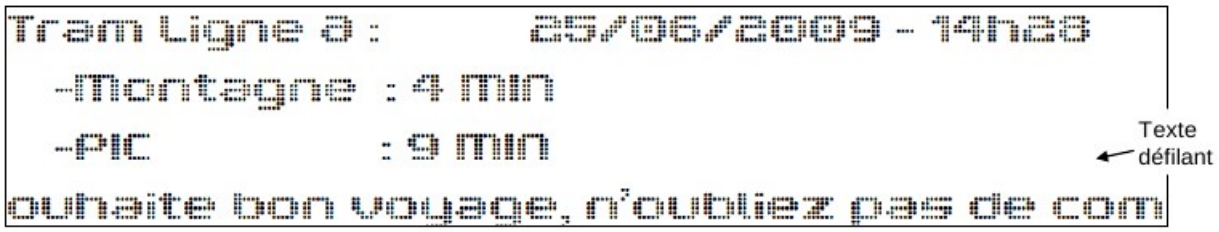


Figure 3 : Exemple d'affichage

Le pilotage des BIV s'effectue depuis un poste opérateur situé dans la salle du Poste de Commande Centralisé (PCC). L'information transite ainsi depuis le sous-système SAE-TR jusqu'aux BIV via le serveur de communication (SCOM), et le réseau multiservice (RMS).

En station, elle est restituée aux BIV à travers une **passerelle ModBusTCP / ModBusRTU avec une liaison série RS485**.

En cas de rupture de liaison avec le serveur central, une situation de repli permet d'afficher un message fixe et configurable.

Les messages à afficher sur les BIV sont élaborés depuis le Poste de Commande Centralisé et envoyés aux BIV sous forme de trame modbusTCP via le réseau RMS. Des passerelles ModbusTCP / Modbus RTU relaient ces messages aux BIV.

Le modèle des passerelles est : UDS1100IAP de la société Lantronix.

Le modèle des BIV est : SX502 de la société SIEBERT.

Les BIV sont interconnectés à la passerelle ModbusTCP/ModbusRTU (UDS1100IAP) par une liaison RS485 deux fils.

Pour répondre aux questions suivantes, il vous faudra consulter les documents : « Annexe 7 : Afficheur SX502», « Annexe 8 : Modbus ».

Exercice n°3

La passerelle ModbusTCP / ModbusRTU est-elle maître ou bien esclave sur le réseau RS485 des BIV ?
Réponse :

On réalise la capture suivante :

```
Source       Destination  Protocol Info

10.98.0.254  10.98.0.3    Modbus/TCP  query [ 1 pkt(s)]: trans:1;
unit: 5, func: 16: Write Multiple Registers.

Ethernet II, Src: (00:16:d3:64:8e:14), Dst: (00:20:4a:b2:38:6c)

Internet Protocol, Src: (10.98.0.254), Dst: (10.98.0.3)

Transmission Control Protocol, Src Port: (30261), Dst Port: (502),
Seq: 0, Ack: 0, Len: 65

Modbus/TCP

    transaction identifier: 1
    protocol identifier: 0
    length: 59
    unit identifier: 5
    Modbus
        function 16:  Write Multiple Registers
        reference number: 0
        word count: 26
        byte count: 52
        Data


00 20 4a b2 38 6c 00 16 d3 64 8e 14 08 00 45 00   . J.8l...d....E.

00 69 26 56 40 00 80 06 be 74 0a 62 00 fe 0a 62   .i&V@....t.b...b

00 03 76 35 01 f6 87 5a 7a 9b 04 2d 9a b8 50 18   ..v5...Zz..-..P.

ff ff 16 20 00 00 00 01 00 00 00 3b 05 10 00 00   ... .......;....

00 1a 34 24 46 31 24 4d 31 24 4c 30 34 49 6e 66   ..4$F1$M1$L04Inf

6f 72 6d 61 74 69 6f 6e 20 76 6f 79 61 67 65 75   ormation voyageu

72 20 3a 20 6c 69 67 6e 65 20 43 20 65 6e 20 70   r : ligne C en p

61 6e 6e 65 24 46 30                              anne$F0

——— ModbusTCP

——— Modbus
```

Tableau 2: Capture d'une trame Modbus TCP

Exercice n°4

Indiquer sur quel réseau cette trame a été capturée.
Réponse :

Exercice n°5

Quel est le code fonction Modbus utilisé pour piloter l'afficheur (Annexe7) ? Donner sa signification.
Réponse :

Exercice n°6

Quels sont les noms des champs de la requête Modbus associés à ce code fonction (Annexe 8) ?
Réponse :

Exercice n°7

Donner la commande à envoyer à l'afficheur permettant l'affichage de la 3ième ligne de l'exemple Figure 10 (Annexe 7)
Réponse :

Sur la capture de la trame Modbus TCP (Tableau 2), la partie Modbus TCP est soulignée en trait plein et la partie Modbus en trait pointillé.

Analyse de l'entête MBAP Header la trame Modbus TCP du tableau 2 :

Exercice n°8

Dans l'entête ModbusTCP (MBAP Header, Annexe 8), quel est le nom du champ qui contient l'adresse Modbus de l'afficheur ? Donner l'adresse Modbus de l'afficheur. Qui qui va recevoir cette trame (le serveur modbus ou l'afficheur) ?
Réponse :

Exercice n°9

Interpréter les 6 premiers octets de la requête Modbus contenus dans la trame ModbusTCP du tableau 2 : Capture d'une trame ModbusTCP.
Réponse :

Champs	Valeur	Signification/Décodage

Exercice n°10

Extraire de la requête Modbus les commandes de manipulation de texte envoyées à l'afficheur (Annexe 7).
Réponse :

Exercice n°11

Donner le résultat de cette commande en terme d'affichage sur le document réponse.
Réponse :