Victor Gosselin

Lilian Poignant

Antoine Verfaillie

# Compte rendus TP Cloches

## Sommaire

Comptes Rendus

[Compte rendus TP Cloches 1](#_Toc24037251)

[1. Sommaire 2](#_Toc24037252)

[2. But 3](#_Toc24037253)

[3. Principe 3](#_Toc24037254)

[4. Réponses aux questions spécifiques 3](#_Toc24037255)

[Question 1 3](#_Toc24037256)

[Question 2 4](#_Toc24037257)

[Question 3 4](#_Toc24037258)

[Question 4 4](#_Toc24037259)

## But

Piloter un carillon de 4 cloches avec une application en C++.

## 4. Réponses aux questions spécifiques

### Question 1

L’ensemble du code comprend l’ADU (Application Data Unit) qui est l'encapsulation des données à travers les paramètres suivants :

l'Address

PDU (Protocol Data Unit) : Le numéro de la transaction, La longueur du message en octets,

Le code la fonction

Error Check.

### Question 2

Ce principe repose sur une communication d’égal à égal entre les applications ; communication réalisée par dialogue entre processus deux à deux :

- un processus client

- un processeur serveur

Les processus ne sont pas identiques mais forment plutôt un système coopératif se traduisant par un échange de données :

-le client réceptionne les résultats finaux délivrés par le serveur.

Le client initie l’échange, le serveur est à l’échange, le serveur est à l’écoute d’une requête cliente éventuelle.

Le service rendu = traitement effectué par le serveur

Pour TCP/IP, ETZ 510 est un serveur.

### Question 3

0000 0000 0006 0006 0002 0001

Valeur hexadécimal de la cloche 1

Code de la Fonction

Longueur du message

### Question 4

Pour effectuer la connexion entre l’application C++ et la carte ETZ, nous avons utilisés des objets de la librairie Windows ainsi que des sockets.