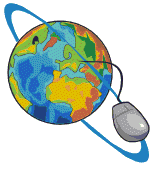
|  |  |
| --- | --- |
| RASoftware | |
| Rapport de consultation |
|  | |

|  |
| --- |
| Rachid AKKOUCHE  05/06/2018 |



|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Ecris par** | **Vérifié par** | **Date** |
| Initial | Rachid AKKOUCHE |  | 05 06 2018 |
|  |  |  |  |

Table des matières

[1. Introduction 2](#_Toc515982981)

[2. ccTalk 4](#_Toc515982982)

[3. J-xFX 4](#_Toc515982983)

[4. LE MATÉRIEL 5](#_Toc515982984)

[4.1. Version 1 5](#_Toc515982985)

[4.2. Version 2 5](#_Toc515982986)

[4.3. Equipements 6](#_Toc515982987)

[5. FONCTIONNEMENT 6](#_Toc515982988)

[5.1. Initialisation 6](#_Toc515982989)

[5.2. Repos 6](#_Toc515982990)

[5.3. Usager 7](#_Toc515982991)

[5.4. Opérateur 9](#_Toc515982992)

[6. Compteurs espèces 12](#_Toc515982993)

[7. Paramètres 12](#_Toc515982994)

[8. Statuts 14](#_Toc515982995)

[9. Défauts 14](#_Toc515982996)

## Introduction

Dans le cadre de la création d’une nouvelle borne de paiement aux péages, nous avons étudié les différentes solutions et contraintes.

Cette étude ne porte pas sur le paiement par badge ou par carte bancaire ou privative.

Afin de préserver les investissements, 2 versions sont en projet.

Une version que nous appellerons V1 ne prévoyant pas le recyclage de la monnaie, mais compatible avec l’existant.

Une version que nous appellerons V2 prévoyant le recyclage de la monnaie. Cette option permettra de réduire drastiquement les opérations de recharge de la monnaie en recyclant les pièces introduites et en permettant le rendu de billets recyclés ou non.

Pour la partie pièces de monnaie, les 2 versions intègreront le protocole ccTalk, tandis que la partie billets intègrera le protocole XFS.

## ccTalk

Le protocole ccTalk est un protocole série multi-drop permettant de sécuriser la communication entre les périphériques et l’application.

Ses avantages sont nombreux. On peut citer parmi ceux-ci :

* Sa simplicité de mise en œuvre : 3 fils seulement sont nécessaires pour alimenter et communiquer avec les périphériques. Mais pour simplifier encore la connectique, dans le cas présent, nous utiliserons des connecteurs HE10-10 montés sur un câble en nappe.
* Sa versatilité : La majorité des fonctions sont applicables sur tous les périphériques quelques soit la marque ou le modèle. Par exemple un monnayeur Eagle ccTalk pourra être, dans la majorité des cas, remplacé par un modèle G13 ccTalk sans aucune modification du programme. De plus l’ajout ou le retrait d’un périphérique sur le bus, s’effectue sans modification HW.
* Le real-time : Le comportement des périphériques peut être modifié, en fonction des nécessités, sans interventions humaines, ni extinction de la machine. Par exemple, l’activation d’un canal d’acceptation se fait par software et peut être décidé par l’intelligence de la machine. De plus, les périphériques peuvent informer l’application de leur état, à tout moment.
* La possibilité de mettre à jour les datasets de monnaie à distance, dans le cas de la découverte d’une nouvelle pièce frauduleuse ou la création d’une nouvelle monnaie.
* Sa sécurité : ccTalk est un protocole série, ce qui rend plus difficile les attaques par rapport au protocole parallèle. De plus sa sécurité peut-être augmenter par des systèmes de chiffrement tel que DES.

## J-xFX

Standard utilisé dans le protocole de communication du BNR – BNA.

## LE MATÉRIEL

### V1

* 1 monnayeur Eagle ccTalk
* 4 hoppers MK IV ccTalk
* 1 lecteur de billet BNA



### V2

* Pelicano (à confirmer)
* 1 lecteur de billet BNR.



### Equipements

Alimentation 24V 12A (48W).

Convertisseur RS232 ccTalk. Pouvant être fourni par RASoftware.

Les connecteurs des hoppers devront être de type HE10-10, voir MEI.

Câble en nappes 10 fils.

## FONCTIONNEMENT

Pour chaque transaction, un numéro unique sera généré par l’application. Il sera utilisé et transmit dans chaque message concernant cette transaction aussi bien par l’application que par la dll.

En cas de distribution, la dll optimisera le rendu afin de maintenir un niveau optimal pour chaque périphérique. Cependant un paramètre sera prévu pour rendre le moins de pièces et de billets possibles.

Lors d’une opération en cas d’échec de celle-ci, la dll enverra immédiatement tous les détails concernant celui-ci.

En cas de défaut la dll enverra toutes les informations permettant de localiser le défaut sur le périphérique afin de permettre une intervention rapide si le défaut peut être supprimé sur site.

### Initialisation

Lors du lancement de la dll, celle-ci lancera l’initialisation physique des périphériques, effectuera la lecture des caractéristiques internes de ceux-ci et affectera les paramètres d’utilisation en fonction d’un fichier xml.

Puis le système passera au mode repos.

### Repos

Lorsqu’aucune activité n’est en cours la dll effectue périodiquement un poll sur chaque périphérique. Si un changement d’état se produit ou si un défaut est détecté la dll enverra un message à l’application provoquant un évènement.

En cas de défaut la dll pourra, en fonction des paramètres contenus dans le fichier xml ou de la gravité de celui-ci, soit continuer en mode dégradé, soit mettre la borne hors-service.

L’application pourra aussi demander à la dll d’interrompre ses activités ou de fonctionner en mode dégradé.

Parallèlement, la dll lira périodiquement le fichier de paramètres pour détecter un éventuel changement de configuration et affecter aux périphériques ceux-ci ou modifier les déclencheurs d’événements tels que les alertes de niveaux ou autres états.

Cependant, pour une mise à jour des datasets des pièces et des billets, un message devra demander expressément à la dll d’effectuer celle-ci.

### Usager

#### Cas 1 :

*L’usager se présente avec l’appoint.*

Il introduit les pièces ou/et les billets. Quand le montant est atteint, la transaction est terminée.

A chaque dénomination introduite, la dll transmettra sa valeur.

Quand la transaction est terminée, la dll transmettra le nombre et le montant de chaque dénomination utilisée ainsi que le montant total.

#### Cas 2 :

*L’usager se présente avec un montant supérieur au montant du péage.*

Il introduit les pièces et/ou les billets, dès que le montant est supérieur, les moyens de paiement sont désactivés, le système retourne le montant trop perçu. La transaction est terminée.

A chaque dénomination introduite, la dll transmettra sa valeur.

Pour chaque dénomination rendue, la dll transmettra le nombre et le montant. Dans le cas d’un BNR le rendu se faisant en lot, le détail du lot sera transmis.

Quand la transaction est terminée, la dll transmettra le nombre et le montant de chaque dénomination introduite ainsi que le montant total introduit et transmettra aussi le nombre et le montant de chaque dénomination rendue ainsi que le montant total rendu.

#### Cas 3 :

*L’usager a introduit un montant inférieur au montant du péage et ne dispose plus d’espèces.*

Il pourra utiliser une carte pour compléter le règlement (paiement multiple). Dans ce cas l’application informe les moyens de paiement qui seront désactivés. La transaction est terminée.

Pour chaque dénomination introduite, la dll transmettra sa valeur.

Quand la transaction est terminée, la dll transmettra le nombre et le montant de chaque dénomination introduite ainsi que le montant total.

#### Cas 4 :

*L’usager a introduit un montant inférieur au montant du péage et ne dispose plus d’espèces et n’a pas d’autre moyens de paiement.*

Il appelle l’assistance, qui une fois la procédure appliquée, informera par l’application les moyens de paiement qui seront désactivés. La transaction est terminée.

Pour chaque dénomination introduite, la dll transmettra sa valeur.

Quand la transaction est terminée, la dll transmettra le nombre et le montant de chaque dénomination introduite ainsi que le montant total.

#### Cas 5 :

*L’usager a introduit un montant inférieur au montant du péage et demande le retour du montant introduit.*

Le système retourne le montant introduit selon la procédure suivante :

* + - Les billets introduits seront retournés.
    - Dans la mesure du possible, des pièces avec une valeur identique à celles introduites seront retournées.

Après voir retirés les billets du bezel du lecteur de billet, l’usager peut recommencer l’acquittement du montant par un autre moyen de paiement ou réintroduire des espèces. Attention, les moyens de paiement ne seront réactivés que si les billets retournés sont retirés du bezel du lecteur de billets.

Dans le cas où l’usager utilise un autre moyen de paiement et laisse les billets dans le bezel, l’application avisera les moyens de paiement du franchissement de la barrière, les billets seront envoyés en caisse, la transaction est terminée.

Pour chaque dénomination introduite, la dll transmettra sa valeur.

Pour chaque dénomination rendue, la dll transmettra le nombre et le montant. Dans le cas d’un BNR le rendu se faisant en lot, le détail du lot sera transmis.

Quand la transaction est terminée, la dll transmettra le nombre et le montant de chaque dénomination introduite ainsi que le montant total introduit et transmettra aussi le nombre et le montant de chaque dénomination rendue ainsi que le montant total rendu.

Dans le cas où l’usager ne reprend pas les billets rendus, la dll transmettra le nombre et le montant des billets théoriquement retournés en caisse ainsi que le montant total. Cependant, rien ne permet de vérifier qu’un billet n’a pas été prélevé dans le lot.

#### Cas 6 : (V2 uniquement)

*L’usager a introduit un montant supérieur au montant du péage et des billets lui sont retournés qui ne sont pas retirés du bezel au moment du franchissement de la barrière.*

Dès le franchissement de la barrière, l’application avise les moyens de paiement et les billets sont envoyés en caisse, la transaction est terminée.

Pour chaque dénomination introduite, la dll transmettra sa valeur.

Pour les pièces, la dll transmettra le nombre et le montant de chaque type rendu.

Pour les billets, le détail du lot rendu indiquant le nombre et le montant de chaque dénomination et le montant total sera transmis.

Quand la transaction est terminée, la dll transmettra le nombre et le montant de chaque dénomination introduite ainsi que le montant total introduit et transmettra aussi le nombre et le montant de chaque dénomination rendue ainsi que le montant total rendu.

Dans le cas où l’usager ne reprend pas les billets rendus, la dll transmettra le nombre et le montant des billets théoriquement retournés en caisse ainsi que le montant total. Cependant, rien ne permet de vérifier qu’un billet n’a pas été prélevé dans le lot.

#### Cas 7 :

*Un changement de montant est appliqué et l’usager a introduit une somme inférieur au nouveau montant réclamé.*

Les procédures des cas précédents s’appliquent pour le nouveau montant. Le premier montant étant abandonné.

### Opérateur

##### Retrait espèces

###### Retrait caisse pièces :

*L’opérateur retire le sac contenant les pièces.*

Quel que soit la raison du retrait, le compteur de caisse de pièces est remis à zéro dès la remise en place du sac. La dll transmettra le nombre et le montant de chaque type de pièce ainsi que le montant total contenu dans le sac.

##### Retrait caisse billets :

*L’opérateur retire la caisse contenant les billets.*

Quel que soit la raison du retrait, le compteur de la caisse est remis à zéro dès la remise en place de la caisse. La dll transmettra le nombre et le montant de chaque type de billet ainsi que le montant total contenu dans la caisse.

##### Rechargement espèces

###### Pièces

*L’opérateur met en place le système de rechargement et effectue la procédure de rechargement.*

L’application avise la dll de l’opération et du numéro du hopper concerné. Celle-ci met à jour les compteurs en fonction des paramètres lus dans le fichier. A chaque numéro de hopper correspond un nombre par défaut de pièces rechargées et une valeur de dénomination.

Cependant l’opérateur pourra corriger le montant. Dans ce cas la dll calculera le nombre de pièces correspondant au montant corrigé.

###### Billets (V2 uniquement)

Loader

*L’opérateur retire un loader et en remet un en place.*

La dll avise l’application et met à jour les compteurs en fonction des paramètres lus dans le fichier. Une valeur et un nombre par défaut de billets rechargés sont indiqués dans le fichier de paramétrages. L’opérateur pourra corriger le montant. Dans ce cas la dll calculera le nombre de billet correspondant au montant corrigé.

Recycler

*L’opérateur souhaite mettre à niveau un billet à recycler.*

L’application avise la dll de l’opération. Celle-ci active le BNR.

Pour chaque dénomination introduite, la dll avisera l’application de la valeur de celle-ci et du module dans lequel elle aura été stockée.

A noter :

* Il n’est pas nécessaire d’indiquer la dénomination ou son module de stockage. Le BNR reconnaissant la dénomination, il l’affectera au module le plus adéquate.
* Dans le cas ou une dénomination ne peut être recycler, soit parce qu’elle n’est pas prévue pour être recyclé, soit parce-que les recyclers concernés sont plein, elle sera stockée en caisse.

Une fois le nombre de billets souhaité introduit, l’application avise la dll qui désactivera le BNR.

##### Distribution espèces

*L’opérateur souhaite forcer une distribution d’espèces.*

###### Par montant

L’application indique le montant à distribuer. La dll provoque la distribution.

Pour chaque dénomination distribuée, la dll transmettra le nombre et le montant. Dans le cas d’un BNR le rendu se faisant en lot, le détail du lot sera transmis.

Quand la transaction est terminée, la dll transmettra le nombre et le montant de chaque dénomination distribuée ainsi que le montant total.

###### Par périphérique

Pièces

L’application indique le numéro du hopper concerné et le nombre de pièces à distribuée. La dll provoque la distribution.

Quand la distribution est terminée, la dll transmettra le nombre et le montant distribué et désactive les périphériques.

Billets (Version 2 uniquement)

L’application indique le nombre et la dénomination souhaités. La dll provoque la distribution

Quand la distribution est terminée, la dll transmettra le nombre et le montant distribués et désactive les périphériques.

##### Vidage espèces

Pièces

L’application indique le numéro du hopper concerné. La dll provoque le vidage.

Lorsque le hopper ne peut plus distribuer de pièces, la dll transmettra le nombre et le montant distribués. Les compteurs sont mis à jour.

Billets (Version 2 uniquement)

Par module

L’application transmet le nom du module concerné à la dll qui provoque le transfert des billets de celui-ci dans la caisse.

Par dénomination

L’application transmet la dénomination à la dll qui provoque le transfert de tous les billets de cette dénomination vers la caisse.

Complet

L’application transmet la requête à la dll qui provoque le transfert de tous les billets contenu dans les modules vers la caisse.

Quand le transfert est terminé, la dll transmettra le nombre et le montant transférés en caisse pour chaque module et pour chaque dénomination ainsi que le montant total. Le BNR sera désactivé.

## Compteurs espèces

Monnayeur

* Un compteur par dénomination du nombre de pièces introduites
* Un compteur total.

Lecteur de billet

* Un compteur par dénomination du nombre de billets contenus
* Un compteur par dénomination du nombre de billets introduits.
* Un compteur par dénomination du nombre de billets rendus.
* Un compteur par dénomination du nombre de billets trop-perçus.
* Un compteur par modules du nombre de billets contenues.
* Un compteur par modules du nombre de billets introduits.
* Un compteur par modules du nombre de billets rendus.
* Un compteur global du montant contenu.
* Un compteur global du montant introduits.
* Un compteur global du montant rendus.
* Un compteur global des montants trop-perçus.
* Un compteur du dernier montant trop-perçu.

Par hopper

* Un compteur du nombre de pièces contenues.
* Un compteur du nombre de pièces introduites.
* Un compteur du nombre de pièces rendues.

L’application pourra à tout moment demander la valeur de ces compteurs, soit individuellement soit globalement.

## Paramètres

###### General

Type d’optimisation

Taux de conversion

###### Lecteur de billets

Pour devises primaires acceptées

* Code ISO
* Valeur
* Variant
* Activée
* Montant à partir duquel la devise sera acceptée.

Pour devises secondaires acceptées

* Code ISO
* Valeur
* Variant
* Activée
* Montant à partir duquel la devise sera acceptée.

Pour chaque module

* Nom du module
* Float level

Pour le loader

* Nombre de billet de recharge par défaut.

###### Monnayeur

Pour chaque canal

* Activé

###### Hopper

* Numéro du hopper
* Nombre de pièces de recharge par défaut.
* Valeur de la dénomination.
* Niveau pré-alarme de vide.
* Niveau alarm vide.
* Critique (mode dégradé).

## Statuts[[1]](#footnote-1)

L’application pourra à tout moment demander les différents états des périphériques tel que décrit dans les manuels de ceux-ci. Tout changement d’état d’un périphérique fera l’objet d’un message provoquant un événement.

## Défauts[[2]](#footnote-2)

L’application pourra à tout moment demander les différents défauts des périphériques tel que décrit dans les manuels de ceux-ci. Toute détection de défaut fera l’objet d’un message provoquant un événement.

La dll définira si la borne peut fonctionner en mode dégradé ou être déclarée hors service. L’application pourra forcer l’application de l’un ou l’autre état.

1. Les statuts des périphériques sont définis dans la documentation de ceux-ci. [↑](#footnote-ref-1)
2. Les défauts des périphériques sont définis dans la documentation de ceux-ci. [↑](#footnote-ref-2)