

protocoles de communication graphique ou Codes  
matriciels

## **Introduction**

Les protocoles de communication graphique [...]

Les codes barres [...]

Les codes 2D [ Qr code...]

## **Sujet**

## RESEAUX 2

### TP PROTOCOLE GRAPHIQUE



L'objectif de ce TP est de concevoir un protocole de communication « graphique ». Un exemple de protocole de communication graphique est le QR-CODE (voir en haut à droite de cette feuille pour un exemple de message graphique).

Vous devrez concevoir un nouveau protocole de A à Z. Vous pouvez vous inspirer du QR-CODE mais devrez proposer vos propres solutions aux différents problèmes posés.

Les informations seront « dessinées » dans une matrice dont vous proposerez vous même les dimensions (plusieurs tailles « fixes » ou formes peuvent être proposées en fonction de la taille du message). Chaque « cellule » de votre matrice constitue l'unité d'information de base (par exemple, pour le QR-CODE, cette unité d'information est le bit).

Votre matrice devra contenir trois types d'informations :

- Le message à transmettre en lui même
- Les informations permettant de détecter et éventuellement corriger des erreurs
- Les informations spécifiques à votre protocole

N'oubliez pas que votre message graphique sera imprimé et que lors de l'impression la taille des cellules (en pixel) peut être modifiée et les couleurs altérées (si vous utilisez des couleurs différentes). De même, l'orientation de votre dessin peut être modifiée.

Exemple d'informations spécifiques à votre protocole (non exhaustif) :

- Information permettant de repérer la matrice au milieu d'une page pouvant contenir d'autres informations
- Information permettant de calculer la taille d'une cellule en pixel
- Information permettant de connaître la position du premier bit du message
- Information permettant de connaître la taille du message
- Information permettant de connaître combien de bits sont codés dans chaque cellule.
- Si vous utilisez des couleurs, informations permettant de retrouver chacune des couleurs utilisées et les bits qu'elles codent
- ...

#### Travail à rendre

- 1) Rapport détaillé présentant votre protocole graphique et justifiant vos choix. Vous devrez en particulier expliquer le rôle de chacune des cellules de votre matrice ainsi que la taille maximum du message pouvant être transmis à l'aide de votre protocole.
- 2) Un programme permettant de saisir un message texte quelconque et qui générera la matrice correspondante respectant votre protocole (vous pouvez utiliser le langage de votre choix, en justifiant éventuellement ce choix).
- 3) Optionnel : proposer une méthode permettant d'ajouter un logo dans votre matrice sans que cela n'altère la fonction de décodage et sans perdre de cellules « codantes ».

## Part I

# Fonctionnement

- 1 - Analyse des données à encoder et paramétrage du niveau de code correcteur. Si pas de niveau de code correcteur spécifié → plus petite version de QR Code.

- 2 - Conversion des données dans un flux de bytes.  
encodage des données (+ concatenation)  
ajout complémentaire  
indicateur de mode (alpha numeric,...) 4bits  
nombre de caractere [...]  
Edition du code de correction d'erreur → choisir niveau de correction  
creation structure messages finale (block d'info/bits)

- 3 - Implémentation de la correction des erreurs. séparation en blocs des bits de données et génération du codes correcteurs.

(Capacité à corriger les erreurs :

Niveau L : environ 7 % de redondance

Niveau M : environ 15 %

Niveau Q : environ 25 %

Niveau H : environ 30 %

-Code correcteur, code de Redd Solomon (et Code de Hamming ))

- 4 - Insérer les données avec le code correcteur dans la matrice.

Utilisation de masque de patterns (Timing pattern, pattern de detection, pattern d'alignement)

Placement des éléments:

→ motifs de positionnement

les séparateur (pour distinguer les motifs de pos)

→ motifs de synchronisation/Timming patterns permettant de percevoir les contrastes entre les modules (clairs et foncés) et il permettent de determiner la version du Qr code (avec ça taille)

→ motifs de d'alignement (pareil mais plus petit que les motifs de positionnement), facilite la lecture en cas de deformation de la matrice.

→ zone tranquille autour du symbol/matrice.

→Placement des info/codewords

- 5 - Génération de la matrice et évaluation du résultat retourné.  
optimisation de la balance entre les modules noirs et les modules blancs et  
minimisation des occurrences de patterns indésirables

Creation du masque

Application du masque  
Informations de format  
Informations de version

- 6 - Génération du QR Code au format image.

(lecture

1 - Reconnaître les bits 1 ou 0.

Le but est de différencier les modules noirs des modules blancs.

2 - Identifier le taux de code correcteur.

3 - Identifier la version du QR Code.

4 - Découvrir la région à décoder.

5 - Lire les données et le code correcteur.

6 - Détecter/Corriger les erreurs.

7 - Décoder les données.

8 - Afficher le résultat.

)

Exemple : ZXing Open source en java

## Part II

# Mise en place

Le langage utilisé est java, avec swing. Le logiciel est développé sous Intel ij. (a mettre dans utilisation ?)

### 0.1 Encodage et versionnage:

...

### 0.2 Conversion:

...

### 0.3 Code correcteur:

...

#### **0.4 Insertion (les differents motifs):**

Contour faisant zone tranquille/tampon

...

#### **0.5 generation matrice ( masque,format,version) :**

...

#### **0.6 Generation Code 2D:**

...

### **Part III**

## **Utilisation**