RÉALISATION D'UN SYSTÈME DE COMMUNICATION ET DE CONTRÔLE PAR COURANT PORTEUR

Ahmed CHEIKHROUHOU

Sommaire

Introduction

La réalisation d'un mini – projet, et ce qu'elle nécessite de connaissances théoriques et de savoir – faire pratique, est une opportunité intéressante pour apprendre à gérer un travail du début jusqu'à la fin.

L'organisation s'avère un élément déterminant pour la bonne conduite du projet et pour la réalisation des différentes phases qui le constituent.

Ce rapport présente d'une manière brève, les différents aspects techniques du travail effectué

Introduction à la technologie courant porteur :

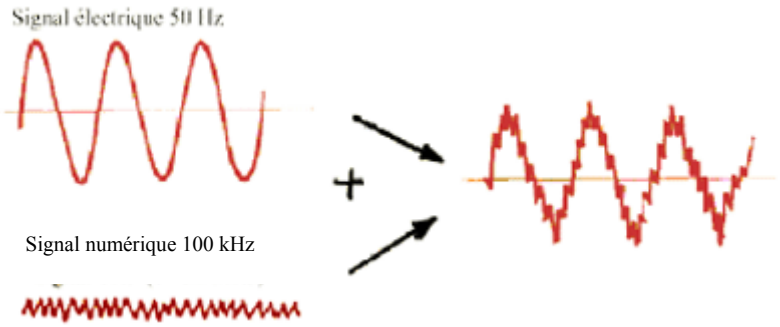
Le courant porteur était déjà utilisé en France dès 1950 pour le contrôle des éclairages publics en ville. Dans les années 1980, le CPL fut utilisé pour le transport d'informations à bas débit pour piloter à distance des appareils électriques (radiateurs, chaudières…).

Interface de communication par courants porteurs

1. Présentation du projet

L'objectif étant de pouvoir utiliser le réseau électrique acheminé par la STEG dans les domiciles comme un support de communication pour pouvoir commander à distance un appareil électrique, on a réalisé deux cartes électroniques : un émetteur et un récepteur.

L'émetteur génère un courant de fréquence élevée (100 kHz) qui sera superposé au courant véhiculé par les fils du secteur (de fréquence 50Hz). Le récepteur, en détectant ce courant, fournira une tension de 230 V aux bornes de la charge ou éliminera cette alimentation en fonction de la commande générée.



2. L'émetteur

2.1 Trame à générer

Le programme mémorisé dans le PIC de l'émetteur doit générer une porteuse à 100 kHz, porteuse modulée en amplitude par les données à transmettre : adresse du récepteur et ordre d'extinction ou d'allumage.

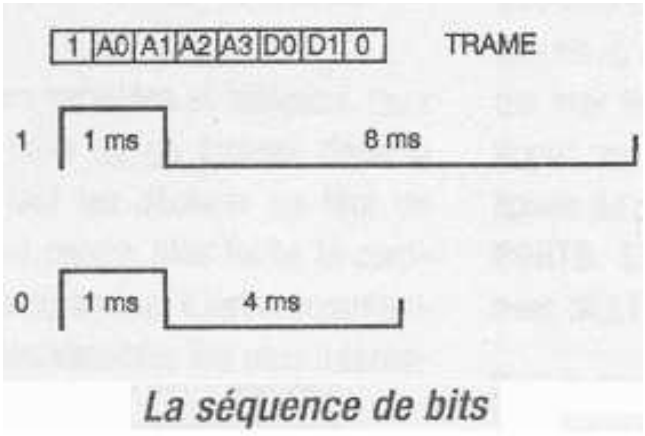


Figure 1 :

Les quatre bits d'adresse A0 à A3 sont déterminés par la position des 4 mini – interrupteurs de l'émetteur, et les deux bits de données D0 et D1 sont tous les deux égaux à 0 en cas d'appui sur le bouton marche (BP2) et tous les deux égaux à 1 en cas d'appui sur le bouton arrêt (BP1). Si le bouton reste appuyé, la même trame est envoyée 40 ms plus tard environ.

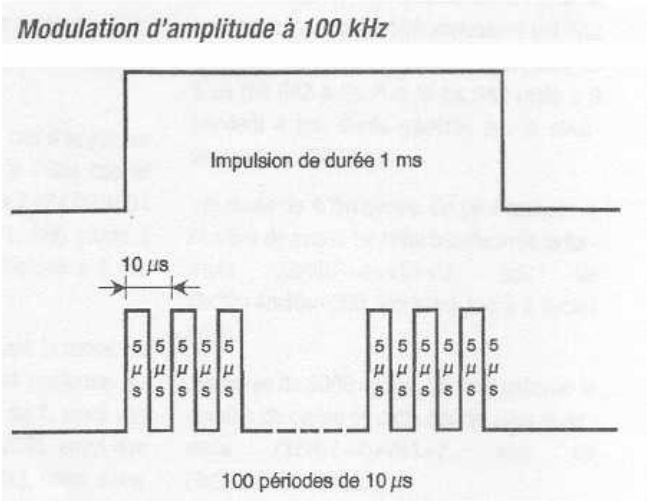


Figure 2 :

La figure 2 est un agrandissement de l'impulsion de 1 ms, on remarque qu'en réalité, chaque impulsion de 1ms est constituée de 100 impulsions de 5 µs espacées de 5 µs : l'impulsion de 1ms module en fait une porteuse de fréquence 100 kHz.

2.2 Schéma électrique

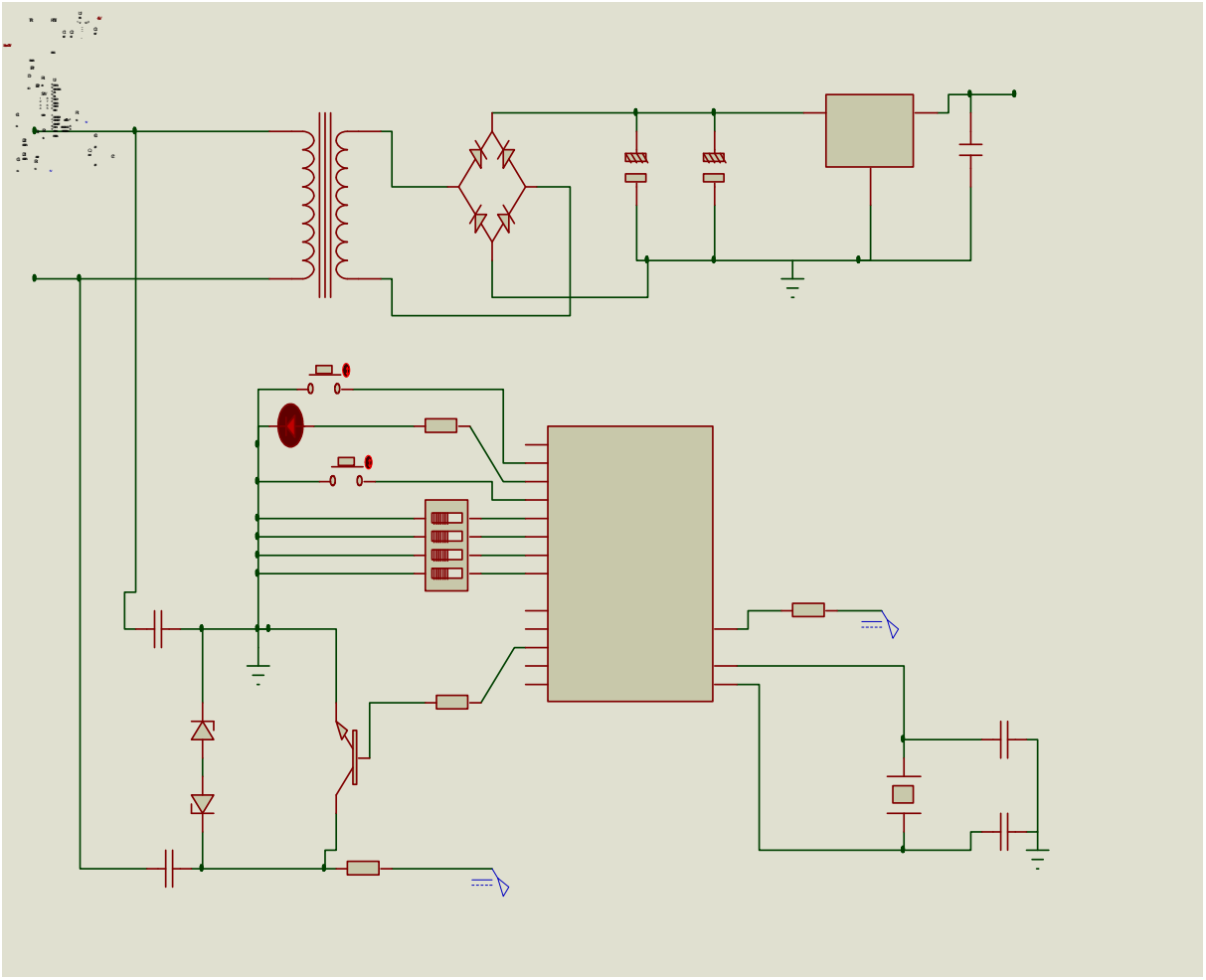


Figure 3 :

- Le circuit d'horloge nécessaire au fonctionnement du PIC 16F628 est constitué du quartz à 4MHz et de ses deux condensateurs associés C6 et C7.
- Les broches utilisées en entrée sont facilement identifiables : RB0 à RB3 sur les - quelles sont branchés les quatre mini-interrupteurs, RB4 et RB6 sur lesquelles sont branchés les deux boutons poussoirs MARCHE et ARRET. Des résistances de pull – up internes au PIC « tirent » ces broches à l'état haut en l'absence d'appui ou quand elles ne sont pas reliées à la masse.
- Les broches RB5 et RA2 sont utilisées en sortie : RB5 est reliée à une LED témoin d'émission, et par RA2 sort la trame à envoyer sur les fils du secteur. Le signal à

transmettre parvient à la base du transistor T par l'intermédiaire de R2. Ce transistor de type NPN est rendu passant quand un niveau haut est présent sur sa base.

- Le courant étant amplifié, il est envoyé sur les lignes du secteur à travers les deux condensateurs d'isolement C4 et C5.

2.3 La programmation des pics

Le programme chargé dans la mémoire du PIC utilisé contient les instructions suivantes (en langage BASIC) :

```
config  _XT_OSC&_WDT_OFF&_LVP_OFF\lang1036
REGISTRES REG_16F628 ; pour le PIC16F628
;(1)Définition des variables et tableaux
VAR  APPUI
VAR  DONNEE
VAR ADRESSE
VAR NB_IMPUL
VAR VB1
VAR VB2
VAR I

; Initialisation
ORG 0
CMCON=7
BSF STATUS,RP0              //  Accès à la page 1 de la RAM
TRISA= 0          //  Toutes les broches du port A sont configurées comme sorties
TRISB=% 0 1 0 1 1111   //  Toutes les broches du port B en entrée sauf RB5 et
RB7
BCF \f1 OPTION_REG ,7     // Activer les résistances de PULL UP  avec le bit 7
BCF STATUS,RP0           //  Accès à la page 0 de la RAM
BCF PORTA, 2     //  Bloquer le transistor

; Le programme principal
PRIN      INTCON=%00001000  // Configurer le pic pour qu'il se réveille si on appuie sur
un bouton
            CLRF PORTB
            BCF PORTA, 2
            SLEEP
            BSF PORTB,5       // La LED s'allume comme un voyant d'émission
LA_ICI      GOSUB BOUTON
            IF APPUI<>0 THEN
             GOSUB ENV_CODE
             ENDIF
            IF APPUI=1 THEN
```

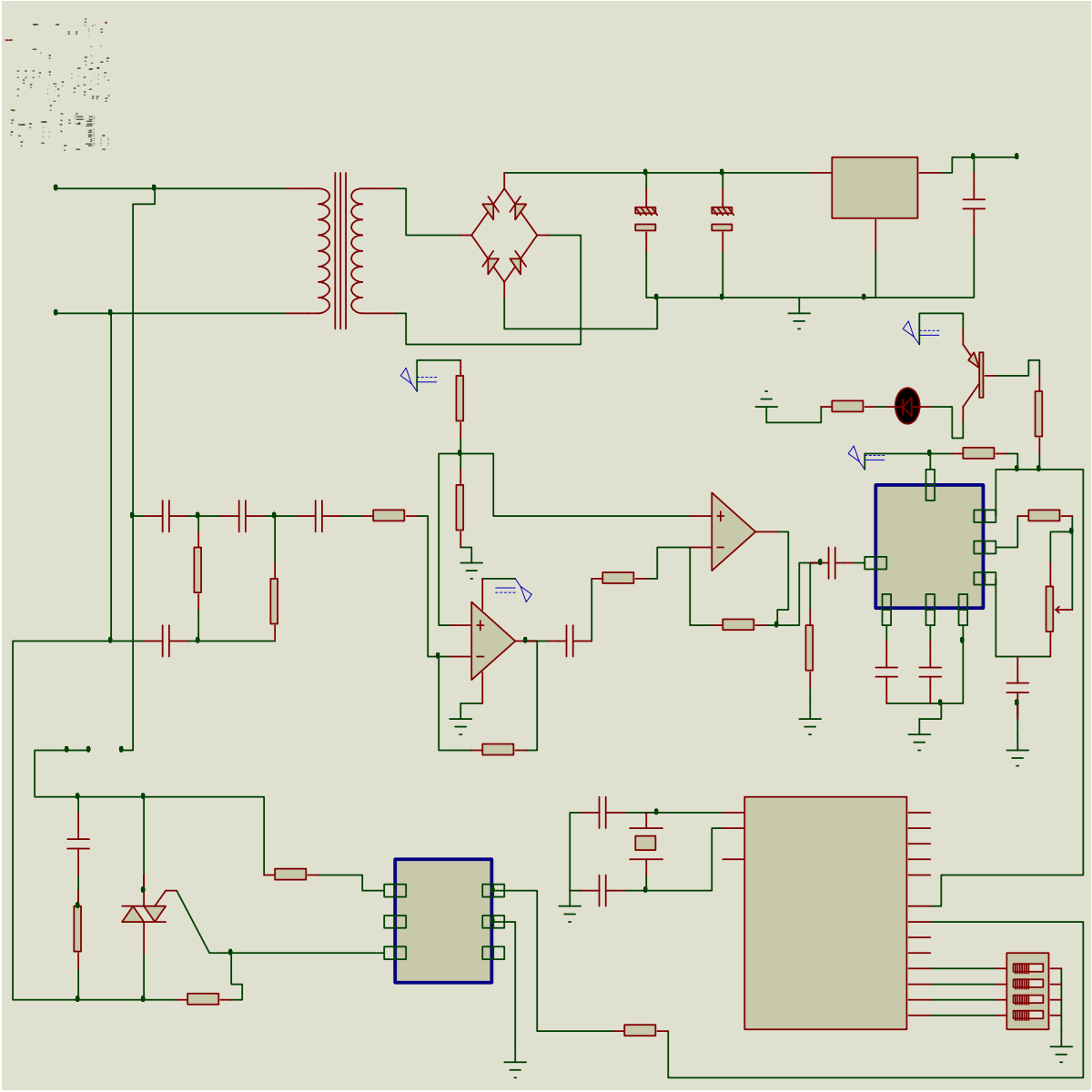
```
            GOTO LA_ICI
            ENDIF
            GOTO PRIN

        ; Gestion des boutons
BOUTON      APPUI=1
            BTFSS PORTB,4
            GOTO TOUCHE_ON
            BTFSS PORTB,6
            GOTO TOUCHE_OFF
            APPUI=0
            GOTO FIN_B
TOUCHE_ON   DONNEE=0
            GOTO FIN_B
TOUCHE_OFF  DONNEE=3
            FIN_B
            RETURN

      ; Envoi des 7 bits de  la trame
      ;  Envoi du start
ENV_CODE    GOSUB ENVOI_1

      ; Envoi de l'adresse
            ADRESSE=PORTB &15
            FOR I=1 TO 4
            RRF ADRESSE,1
            BTFSC STATUS,C
            GOSUB ENVOI_1
            BTFSS STATUS,C
            GOSUB ENVOI_0
            NEXT I

      ; Envoi de la donnée
            FOR I=1 TO  2
            RRF  DONNEE ,1
            BTFSC STATUS,C
            GOSUB  ENVOI_1
            BTFSS STATUS,C
```

```
                GOSUB ENVOI_0
                NEXT I


        ; Envoi du stop
                GOSUB ENVOI_0
                FOR I=1 to  9
                GOSUB CYCLE 4m
                NEXT I
                RETURN

         ; Sous-programme d'envoi d'un 1
ENVOI_1     NB_IMPUL= 100
BCL_1       BSF PORTA, 2
            NOP
            NOP
            NOP
            NOP
            BCF PORTA, 2
            NOP
            DECFSZ NB_IMPUL,1
            GOTO BCL_1
            GOSUB CYCLE8m
            RETURN


        ; Sous-programme d'envoi d'un 0
ENVOI_0     NB_IMPUL= 100
            BCL_2  BSF PORTA, 2
            NOP
            NOP
            NOP
            NOP
            BCF PORTA,2
            NOP
            DECFSZ     NB_IMPUL,1
            GOTO    BCL_2
            GOSUB    CYCLE 4m
```

```
            RETURN

                ; Durée de 4000 cycles
CYCLE4m     VB1= 40
                BOUC1 VB2=32
                BOUC2  DECFSZ  VB2,1
                GOTO BOUC2
                DECFSZ VB1,1
                GOTO BOUC1
                RETURN
        ; Durée de 8000 cycles
        CYCLE8m  VB1= 80
        BOUC3    VB2=32
        BOUC4     DECFSZ  VB2,1
                  GOTO BOUC4
                  DECFSZ VB1,1
                  GOTO BOUC3
                  RETURN
```

3. Le récepteur

3.1 Schéma électrique



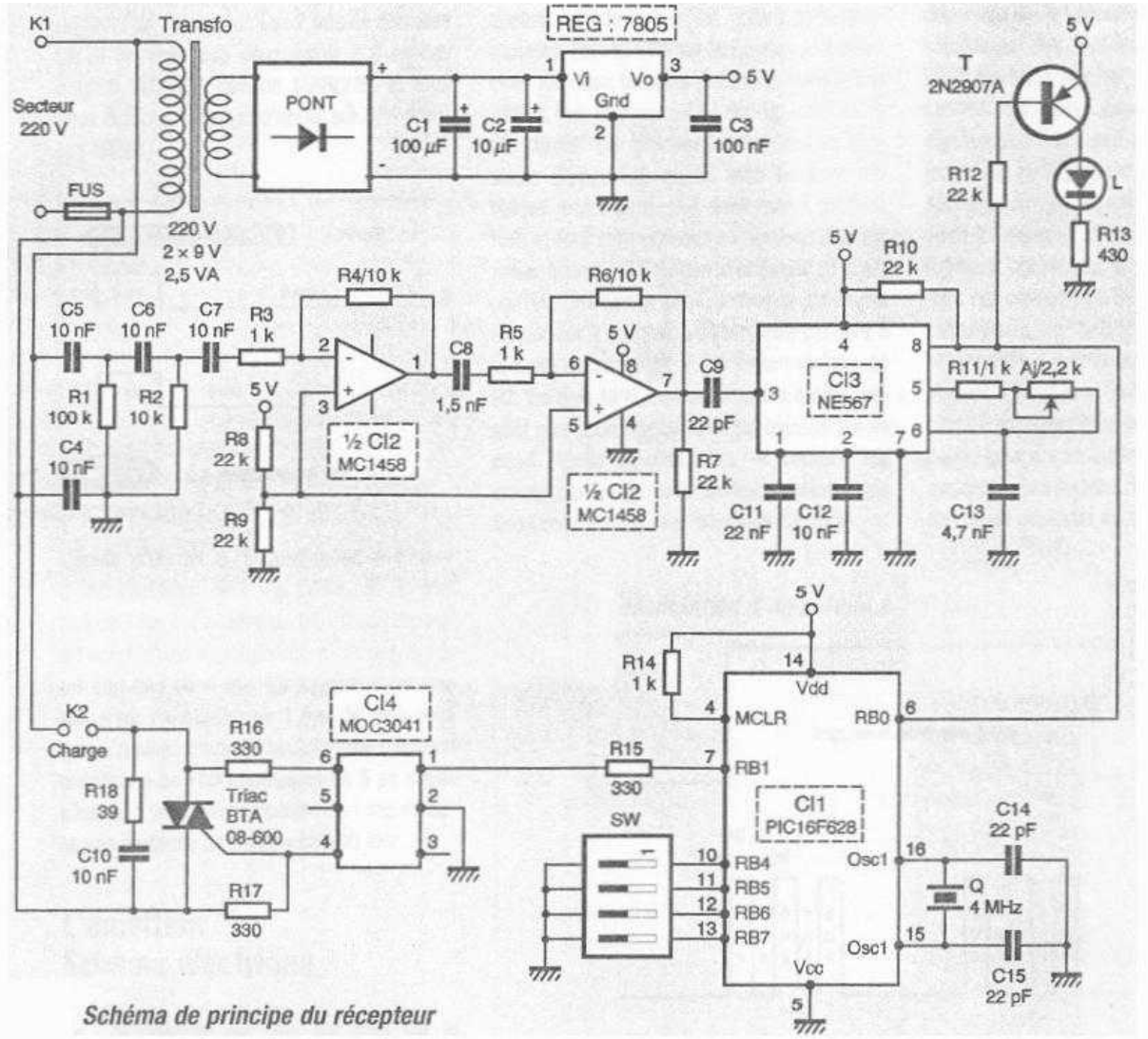


Schéma de principe du récepteur

- Le récepteur a trois missions à accomplir : détecter les signaux de fréquence 100 kHz provenant de l'émetteur, décoder ces signaux puis exécuter l'ordre commandé.
- Tous les signaux provenant du secteur sont d'abord filtrés par les condensateurs C4 à C7 et les résistances R1 et R2. Les signaux HF qui ont réussi le passage de ces filtres sont ensuite amplifiés par 100 à l'aide de deux amplificateurs opérationnels montés en série. Ces ampli-op de type 1458 ne peuvent amplifier chacun que d'un facteur 10, un signal à une fréquence de 100 kHz.
- En sortie du second ampli-op, le signal est envoyé au NE567, décodeur de fréquence à PLL. Les composants C13, R11, et AJ sont choisis et réglés pour qu'en présence du 100 kHz sur la broche3, la broche 8 normalement à l'état haut, passe à l'état bas. Dans notre cas, lors de la réception d'une trame émise par notre

émetteur, on observera sur la sortie 8 des états bas de 1 ms séparés par des états hauts de 4 ou 8 ms.

- La sortie 8 est ensuite connectée d'une part au pic qui sera chargé de décoder ces signaux et d'autre part à une LED qui servira de témoin de réglage et de réception.
- Apres décodage, si l'adresse envoyée correspond à l'adresse affichée sur les minis interrupteurs connectés de RB4 à RB7, l'ordre reçu est exécuté. Le triac est activé par l'intermédiaire du MOC3041 en portant la broche RB1 à l'état haut ou désactivée en la plaçant à l'état bas.
- Le MOC3041 est un petit circuit intégré d'interface, spécialement conçu pour commander des triacs à partir de systèmes fournissant des signaux logiques. Ce circuit construit autour d'un optocoupleur et d'un détecteur de passage a zéro de la tension secteur offre une isolation galvanique de 7500 V et ne génère pas de parasite. Le triac utilisé est de type BTA 08-600 qui peut fonctionner sous une tension de 600 V et débiter 8A.

3.2 Programmation du PIC

Le programme chargé

```
__config  _XT_OSC&_WDT_OFF&_LVP_OFF\lang1036\tab

REGISTRES REG_16F628 ; pour le PIC16F84
        ; Définition des variables et tableaux
                VAR I
                VAR DONNEE
                VAR DUREE
                VAR ADRESSE
                VAR AD_REC
                VAR VB1
                VAR VB2

        ; Initialisation
                ORG 0
                CMCON=7
                BSF STATUS , RP0
                TRISA=0
                 TRISB= % 1110001
                 BCF  OPTION_REG ,7
                 BCF  OPTION_REG ,6
                 BCF STATUS,RP0
                 AD_REC=PORTB
                 SWAPF AD_REC,1
                AD_REC=AD_REC & 15


          ;  Le programme principal
PRIN            INTCON=%00010000
                SLEEP

        ; Décodage des 5 premiers bits
                ADRESSE=0
                 FOR I= 4  DOWNTO 0
                DUREE=0
```

```
TEST_A      BTFSS PORTB,0
            GOTO TEST_A
 LONG_1     GOSUB DELAI100
            INCF DUREE,1
            IF DUREE>200 THEN
            GOTO PRIN
            ENDIF
            BTFSC PORTB,0
            GOTO LONG_1
            IF DUREE> 60 THEN
            BSF STATUS,C
            ELSE
            BCF STATUS,C
            ENDIF
            RRF ADRESSE,1
            NEXT I

          ; Décodage des 2 derniers bits
            DONNEE=0
            FOR I= 1  DOWNTO 0
            DUREE=0
 TEST_D     BTFSS PORTB,0
            GOTO TEST_D
LONG_2      GOSUB DELAI100
            INCF DUREE,1
            IF DUREE>200 THEN
            GOTO PRIN
            ENDIF
            BTFSC PORTB,0
            GOTO LONG_2
            IF DUREE> 60 THEN
            BSF STATUS,C
            ELSE
            BCF STATUS,C
            ENDIF
            RLF DONNEE,1
            NEXT I
```

```
        ; Extraction des 4 bits d'adresse
            SWAPF ADRESSE,1
            ADRESSE=ADRESSE & 15

        ;  Vérifier si l'adresse est bonne
             IF ADRESSE<> AD_REC  THEN
            GOTO PRIN
            ENDIF

       ; Bonne adresse, on traite les données
            IF DONNEE=0 THEN
            BSF PORTB,1
            ENDIF
            IF DONNEE= 3  THEN
            BCF PORTB,1
            ENDIF

         ; Données traitées, on repart au début
            GOTO PRIN
        ; Pause de 100  microsecondes
DELAI100  VB1=14
 BOUC1    VB2=1
 BOUC2    DECFSZ  VB2,1
          GOTO BOUC2
          DECFSZ VB1,1
          GOTO BOUC1
          RETURN
```