

Université François-Rabelais de Tours

Institut Universitaire de Technologie de Tours

Département Génie Électrique et Informatique Industrielle



Borne d'Arrivée pour le projet 50m départ arrêté

ROUZIES Iwan
GRANGER Geoffrey
2^{ème} Année – Groupe Q2

Enseignant:
LEQUEU Thierry
GLIKSOHN Bernard


INTRODUCTION



Borne d'Arrivée 50m Départ-Arrêté


2

INTRODUCTION



Borne de Départ

Borne d'Arrivée



50 mètres

Mesure de la vitesse

Borne d'Arrivée 50m Départ-Arrêté

3

SOMMAIRE

Introduction

- 1. Cahier des Charges
- 2. Analyse Technique
- 3. Test du projet





Conclusion

Borne d'Arrivée 50m Départ-Arrêté

4

CAHIER DES CHARGES

1. Analyse des systèmes présents sur le projet :

- l'alimentation +15V et +5V 
- les afficheurs 7 segments 
- la carte microcontrôleur ATmega 8535 
- les capteurs par faisceaux lumineux 

2. Changement à apporter sur le projet:

- L'alimentation des lampes Maglite®
- La programmation de la carte microcontrôleur

Borne d'Arrivée 50m Départ-Arrêté 5

CAHIER DES CHARGES

- La finalité du produit
- Les contraintes entourant le projet :
 - Composants utilisés
 - La rapidité du microcontrôleur
 - Limites de fonctionnement
 - Performances Attendues

Borne d'Arrivée 50m Départ-Arrêté 6

SOMMAIRE

Introduction

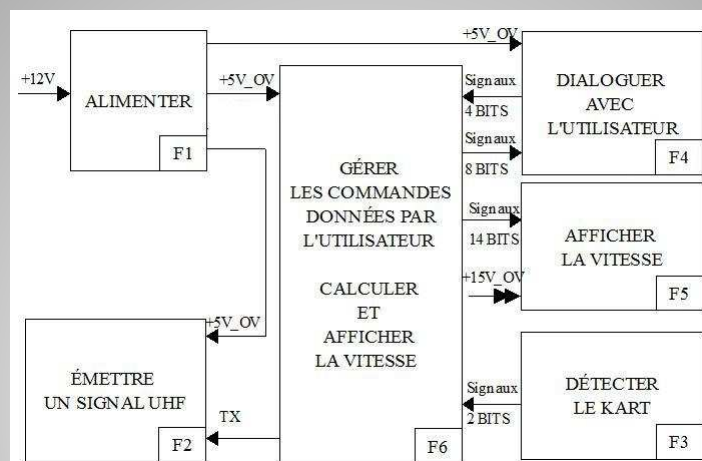
1. Cahier des Charges
- 2. Analyse Technique
3. Test du projet

Conclusion

Borne d'Arrivée 50m Départ-Arrêté

7



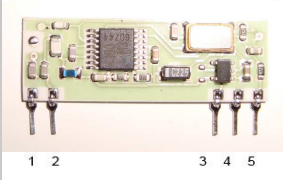


ANALYSE TECHNIQUE



Borne d'Arrivée 50m Départ-Arrêté

8

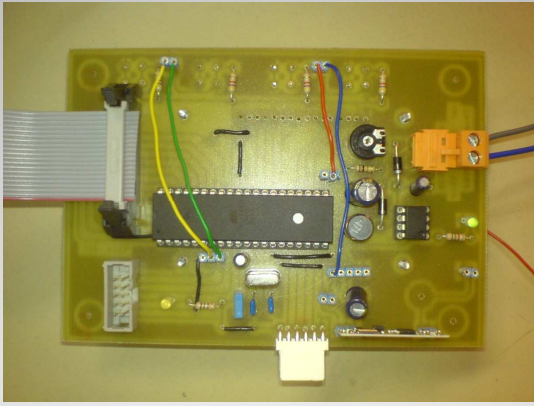
ANALYSE TECHNIQUE

F1 :ALIMENTER	F3 :DÉTECTER LE KART
	
F2 :ÉMETTRE UN SIGNAL UHF	F4 :DIALOGUER AVEC L'UTILISATEUR
	
	F5 :AFFICHER LA VITESSE
	

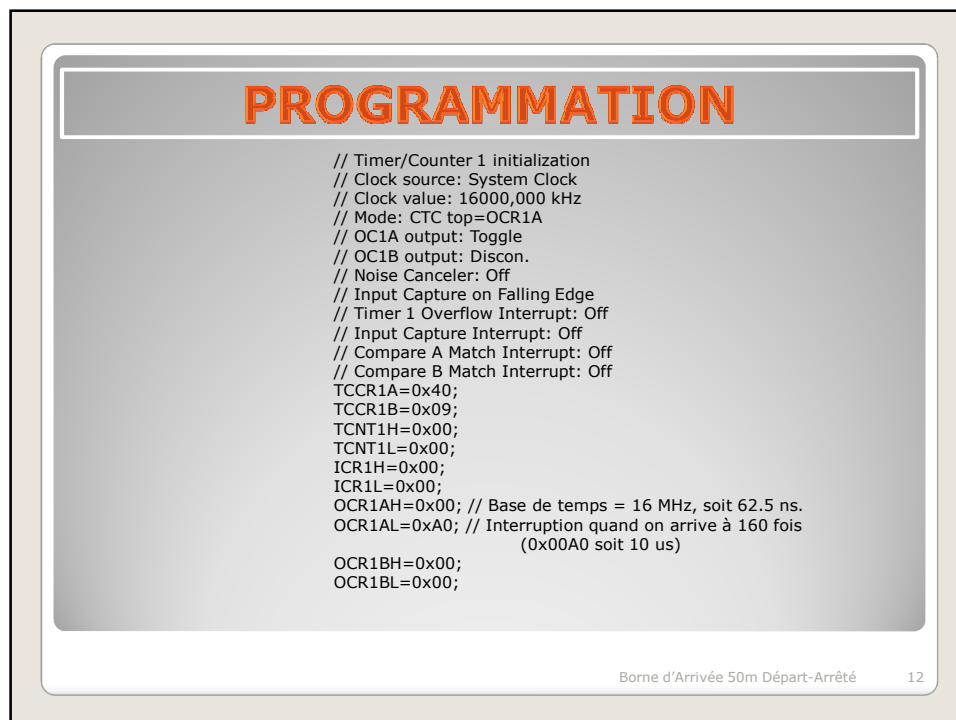
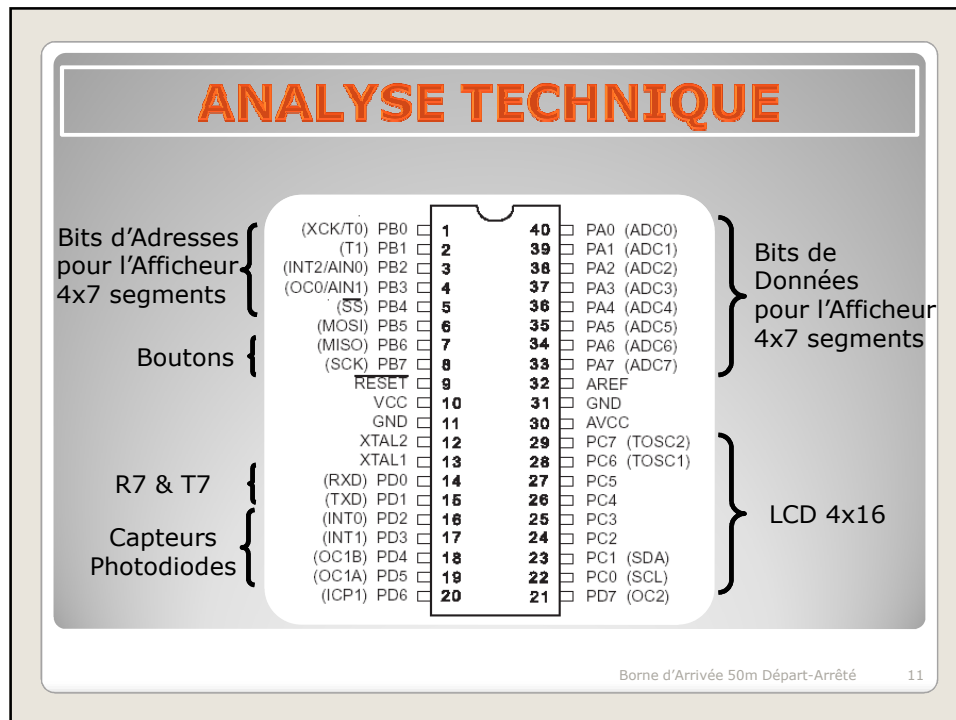
Borne d'Arrivée 50m Départ-Arrêté 9

ANALYSE TECHNIQUE

F6: CALCULER ET AFFICHER LA VITESSE



Borne d'Arrivée 50m Départ-Arrêté 10



PROGRAMMATION

```
// Timer 1 output compare A interrupt service routine
interrupt [TIM1_COMPA] void timer1_compa_isr(void)
{
  cms++;
  if(cms>9)
  {
    cms=0;
    dms++;
    if(dms>9)
    {
      dms=0;
      ms++;
      if(ms>9)
      {
        ms=0;
        dixms++;
        if (dixms>9)
        {
          dixms=0;
          centms++;
          if (centms>9)
          {
            centms=0;
            seconde++;
            cap1=0;
            cap2=0;
            tab[0]=71;
            tab[1]=71;
            if(var2==1)
            {
              secondep=0;
              var2=2;
            }
            if(var2==2)
            {
              secondep++;
            }
            if(secondep>3)
            {
              var2=0;
            }
            if (seconde>59)
            {
              seconde=0;
            }
          }
        }
      }
    }
  }
}
```

Borne d'Arrivée 50m Départ-Arrêté 13

PROGRAMMATION

```
unsigned char tab[3]={0,0,0x90};
```

```
interrupt [13] void usart_txcomplete(void) //fonction interruption avec usart
{
  UDR=tab[k]; //UDR est la donnée envoyée
  k++; //On incrémente k
  if (k==3) k=0; //retour à 0 après avoir envoyer toutes les données
}
```

Borne d'Arrivée 50m Départ-Arrêté 14

PROGRAMMATION

```
// External Interrupt 0 service routine
interrupt [EXT_INT0] void ext_int0_isr(void)
{
    if((var1==1)&&(var2==0))
    {
        T1=(centms*1000+dixms*100+ms*10+dms);
        secondex=seconde;

        var1=0;
    }
    cap1=1;
    tab[0]=83;
}
```

Borne d'Arrivée 50m Départ-Arrêté

15

PROGRAMMATION

```
// External Interrupt 1 service routine
interrupt [EXT_INT1] void ext_int1_isr(void)
{
    if((var1==0)&&(var2==0))
    {
        T2=(centms*1000+dixms*100+ms*10+dms);

        if(secondex>seconde)
        {
            T2=((seconde-secondex)*10000+centms*1000+dixms*100+ms*10+dms);
        }

        cal_aff_vitesse();

        var1=1;
        var2=1;
    }
    cap2=1;
    tab[1]=83;
}
```

Borne d'Arrivée 50m Départ-Arrêté

16

PROGRAMMATION

```
//Calcul et affiche la vitesse
void cal_aff_vitesse(void)
{
    T=T2-T1;

    Vitesse1=18000/(T/(6));
    Vitesse=(int)Vitesse1;

    centainev = Vitesse/1000;
    dizainev = (Vitesse-centainev*1000)/100;
    unitev = (Vitesse-(centainev*1000+dizainev*100))/10;
    centiemev = Vitesse-(centainev*1000+dizainev*100+unitev*10);

    afficheur(0,centainev,0);
    afficheur(1,dizainev,0);
    afficheur(2,unitev,1);
    afficheur(3,centiemev,0);
}
```

Borne d'Arrivée 50m Départ-Arrêté

17

PROGRAMMATION

```
// Affiche une valeur sur 8 bits à l'adresse de l'afficheur :
void afficheur(unsigned char adresse,unsigned char caractere, unsigned char point)
{
    if (point == 1)
    {
        PORTA=(valeur_constant[caractere] | 0x01 ); // Un caractère avec le point !
    }

    else
    {
        PORTA=valeur_constant[caractere]; // Un caractère
    }

    PORTB=(0b00010000 | adresse_constant[adresse & 0x0F]); // PB7 PB6 PB5 CS
    A0 A1 A2 A3
    PORTB.4=1;
    PORTB.4=0; // CS = 0
    PORTB.4=1;
    PORTA=0x00;
}
```

Borne d'Arrivée 50m Départ-Arrêté

18

PROGRAMMATION

```
//tableau adressage //tableau de croisement des bits d'adresse
flash const unsigned char adresse_constant[16] =
{0,8,4,12,2,10,6,14,1,9,5,13,3,11,7,15};
```

```
//tableau lettrage
flash const unsigned char valeur_constant[] =
{
0xEE,0x28,0x76,0x7C,0xB8,0xDC,0xDE,0x68,0xFE,0xFC, //0123456789
0xFA,0x9E,0x16,0x3E,0xD6,0xD2,0xCE,0xBA,0x82,0x2E, //abcdefghij
0x9A,0x86,0x1A,0x1A,0xEE,0xF2,0xF8,0x12,0xDC,0x96, //klmnopqrst
0xAE,0x0E,0xB6,0xBA,0xB2,0x76,0x04, //uvwxyz_
0,1,2,4,8,16,32,64,128,256}; //37->46
```

Borne d'Arrivée 50m Départ-Arrêté

19

PROGRAMMATION

```
//Test du grand afficheur
void test_aff(void)
{
for(k=0;k<3;k++)
{
for(j=37;j<47;j++)
{
for(i=0;i<4;i++)
{
afficheur(i,j,0);
delay_ms(50);
}
}
}
}
}
```

Borne d'Arrivée 50m Départ-Arrêté

20

PROGRAMMATION

```
//Affiche les informations sur l'écran LCD
void affichage_lcd(void)
{
  if(b1==1)
  {
    sprintf(tampon," Tps= %2d:%d%d ",seconde,centms,dixms);
    lcd_gotoxy(0,0);
    lcd_puts(tampon);

    sprintf(tampon," T= %4d ms ",T/10);
    lcd_gotoxy(0,1);
    lcd_puts(tampon);

    sprintf(tampon," Vit= %4d.%dkm/h",Vitesse/10,Vitesse%10);
    lcd_gotoxy(0,2);
    lcd_puts(tampon);

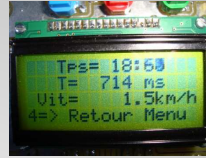
    lcd_gotoxy(0,3);
    lcd_puts("4=> Retour Menu ");
  }

  if(b2==1)
  {
    sprintf(tampon,"1=%4d|2=%4d|%d",T1/10,T2/10,cap1,cap2);
    lcd_gotoxy(0,0);
    lcd_puts(tampon);

    sprintf(tampon,"T=%4d|Tps=%2d:%d%d",T/10,seconde,centms,dixms);
    lcd_gotoxy(0,1);
    lcd_puts(tampon);

    sprintf(tampon," Vit= %4d.%dkm/h",Vitesse/10,Vitesse%10);
    lcd_gotoxy(0,2);
    lcd_puts(tampon);

    lcd_gotoxy(0,3);
    lcd_puts("4=> Retour Menu ");
  }
}
```



Borne d'Arrivée 50m Départ-Arrêté

21

PROGRAMMATION

```
if(b3==1)
{
  lcd_gotoxy(0,0);
  lcd_puts(">Test Grand Aff<");
  lcd_gotoxy(0,1);

  if(k==0)
  {
    lcd_putsf(" EN COURS ");
    lcd_gotoxy(0,2);
  }
  if((k==3)&&(j==47))
  {
    lcd_putsf(" FINI ");
    lcd_gotoxy(0,2);
  }

  lcd_putsf("");
  lcd_gotoxy(0,3);
  lcd_puts("4=> Retour Menu ");

  test_aff();
}

if(b4==1)
{
  lcd_gotoxy(0,0);
  lcd_putsf("====>Menu<====");
  lcd_gotoxy(0,1);
  lcd_putsf("1=> Aff. normal ");
  lcd_gotoxy(0,2);
  lcd_putsf("2=> Aff. detail ");
  lcd_gotoxy(0,3);
  lcd_putsf("3=> Test Grd Aff");
}
}
```



Borne d'Arrivée 50m Départ-Arrêté

22

SOMMAIRE

Introduction

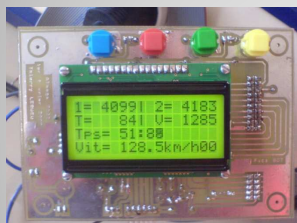
1. Cahier des Charges
2. Analyse Technique
- 3. Test du projet

Conclusion

Borne d'Arrivée 50m Départ-Arrêté

23

TEST DU PROJET



Borne d'Arrivée 50m Départ-Arrêté

24

SOMMAIRE

Introduction

1. Cahier des Charges
2. Analyse Technique
3. Test du projet

➤

Conclusion

CONCLUSION

