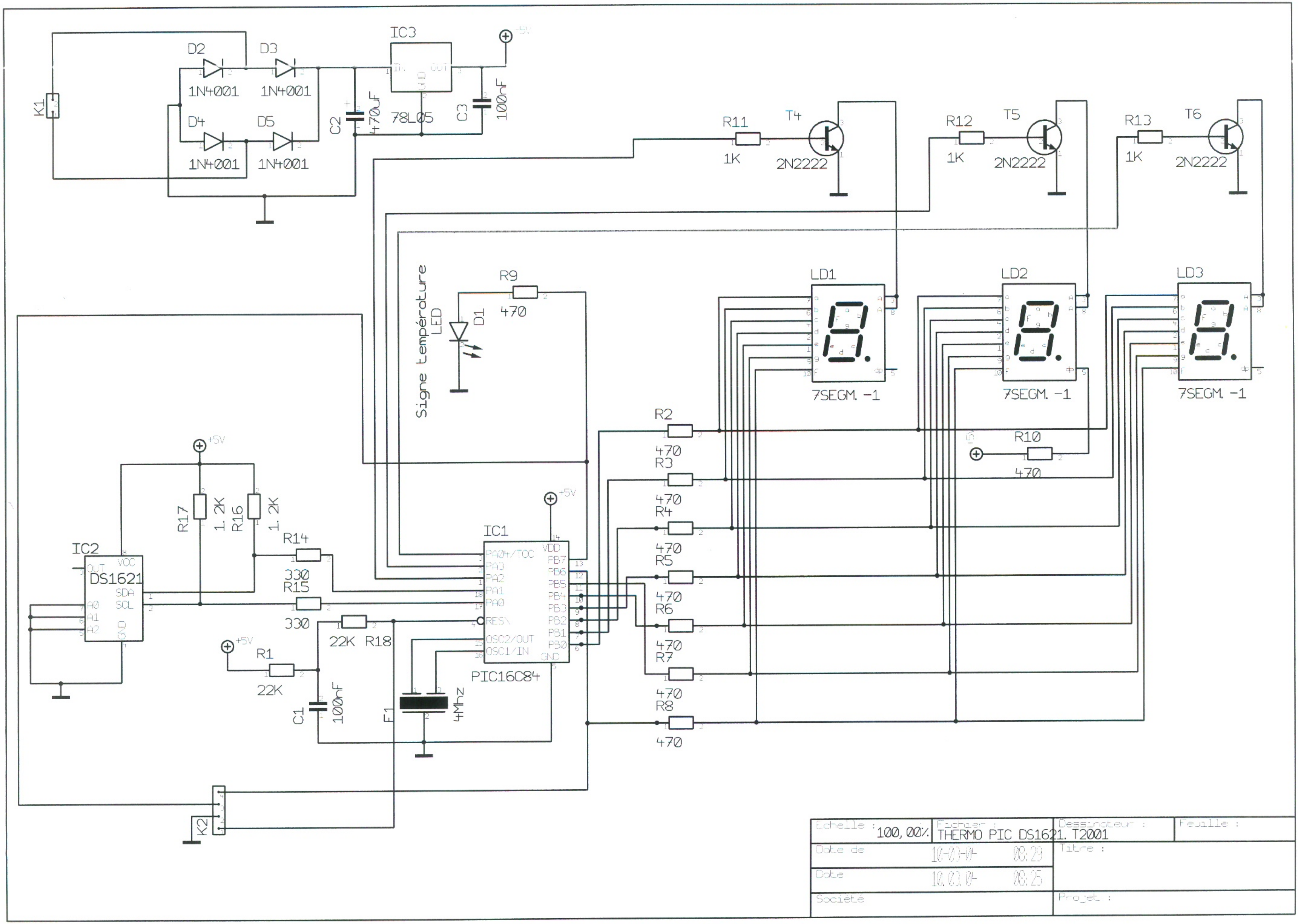


Echelle : 100,00%

Fichier : THERMO PIC DS1621. T2001

Dessinateur :

Feuille :

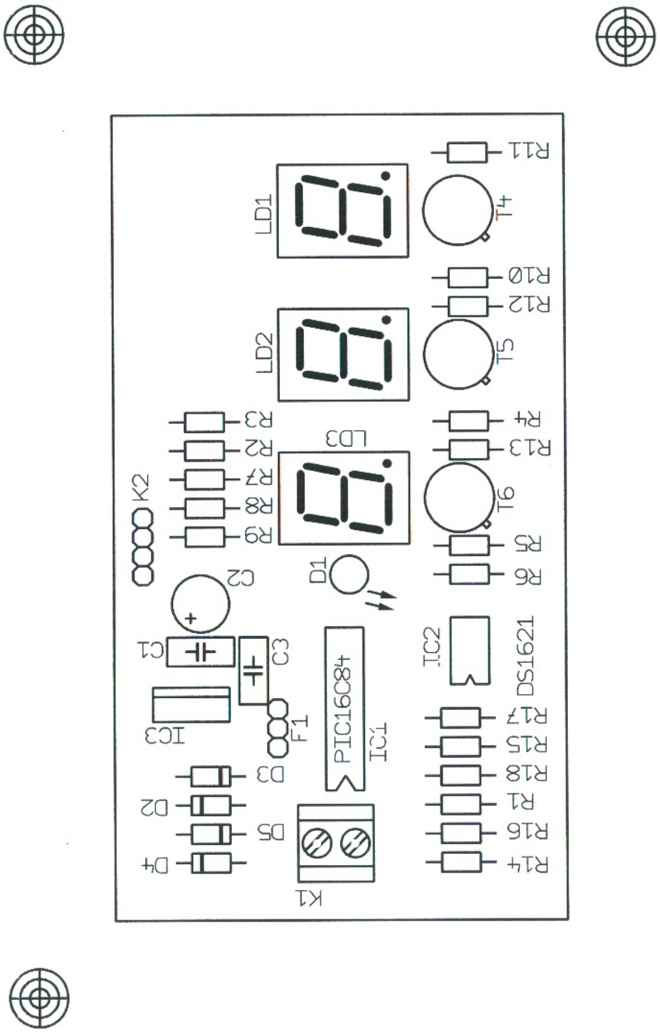
Date de 10-03-04 08:29

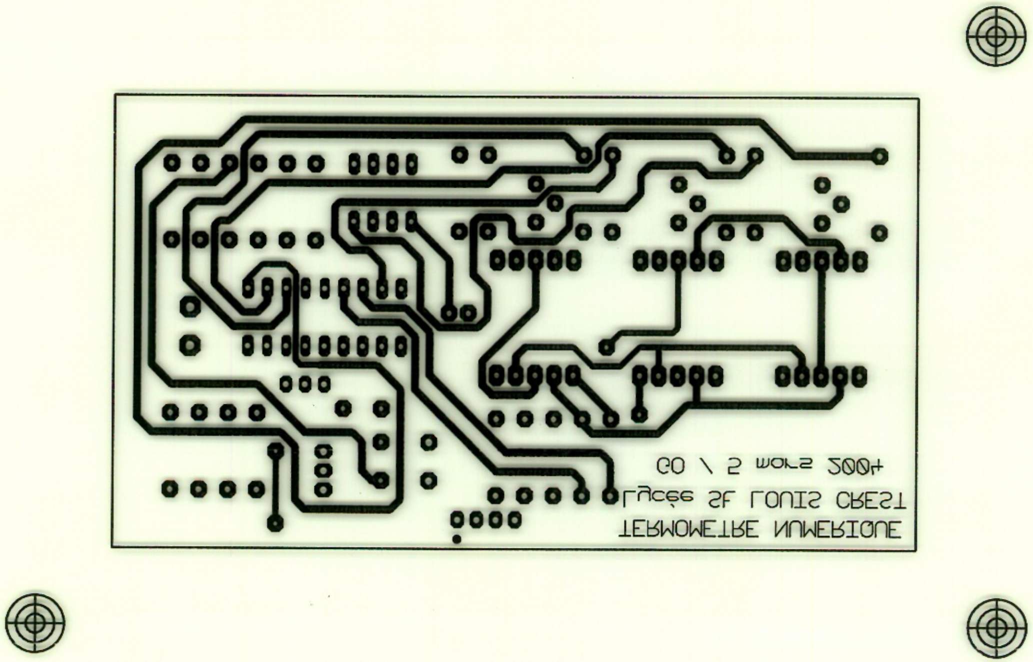
Date 10.03.04 08:25

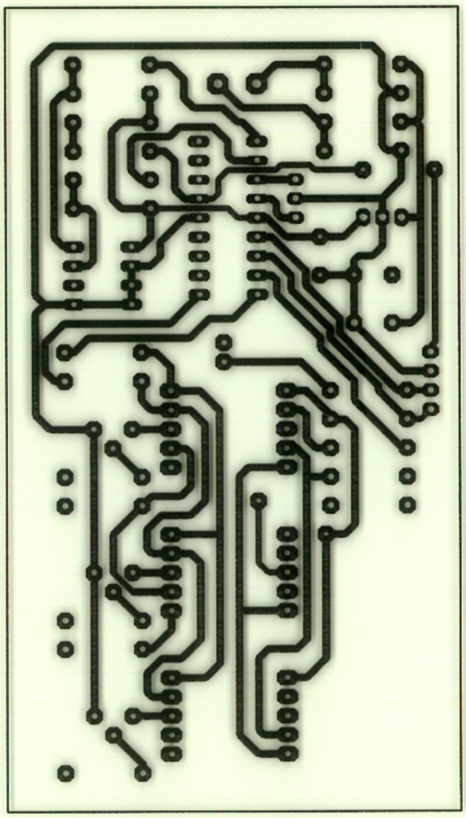
Titre :

Société

Projet :


R11
LD1
T4
R10
R12
LD2
T5
R3
R2
R7
R8
R9
K2
LD3
R4
R13
T6
R5
R6
D1
C2
C1
C3
IC2
DS1621
PIC16C84
IC1
IC3
F1
R17
R15
R18
R1
R16
R14
D3
D2
D5
D4
K1


TERMOMETRE NUMERIQUE
Lycée St LOUIS CREST
GO / 5 mars 2004



```
// Objet du programme :
// Commencé le 07/02/03
#include    std84.h
#include    bit84.h
char octet, adrds1621r, adrds1621w,cdeliretemp, i,j,tempera_h,tempera_l,a;
char signe,affu, affd, affdecimal;
char  unite [10] = { 0x3F, 0x06, 0x5B, 0x4F, 0x66, 0x6D, 0x7D, 0x07, 0x7F, 0x6F } ;

void main()
{
      adrds1621r = 0x91;        //10010001
      adrds1621w = 0x90;        //10010000
      cdeliretemp = 0xAA;       // commande pour lire la température
      trisa = 0x00;
      trisb = 0x00;
      portb = 0x00;             // extinction des afficheurs
      porta = 0xFF;
      stop();                   // mettre les lignes PA1 et PA0 à 1 ( repos du bus )
      lancerds1621();
      microdelay(100);
      for(;;)
      {
            signe = 0x00;
            lectureds1621();
            convertir();
            afficher();
      }
}
//*******************************************************************************
void lectureds1621()
{
      start();                  // START
      microdelay(10);
      octet = adrds1621w;       // adresse du ds1621 en ecriture pr dire qu'on va
      serial();                                   // envoyer une commande de lecture
      acknol();
      microdelay(10);
      octet = cdeliretemp;
      serial();
      acknol();
      microdelay(10);
      porta.1 =1;
      microdelay(10);
      porta.0 = 1;
      microdelay(10);
      start();
      microdelay(10);
      octet = adrds1621r;       // adresse du ds1621 en lecture
      serial();
      acknol();
      microdelay(10);
      serialin();
      tempera_h = octet;
      acknolmaster();
      microdelay(10);
      serialin();
      tempera_l = octet;
      stop();
      microdelay(100);
}
//*******************************************************************************
void lancerds1621()
{
      start();
      octet = adrds1621w;
      serial();
      acknol();
      microdelay(10);
      octet = 0xEE;
      serial();
      acknol();
      microdelay(10);
      stop();
      microdelay(100);
}
//*******************************************************************************v
```

```
void serial()
{
      for ( i=1;i<8;i++)
      {
            if ((octet & 0x80)==0x80)
            porta.1 = 1;      // sda à 1
            else
            porta.1 = 0;      // sda à 0
            octet = octet << 1;
            pulse();
      }
}
//******************************************************************************
void serialin()
{
      trisa.1 = 1;      //pa0 en entrée
      for ( i=1;i<8;i++)
      {
            octet = octet << 1;
            if ((porta & 0x02 ) ==0x02)
                  octet = octet | 0x01;
            pulse();
      }
      trisa.1 = 0;
}

//******************************************************************************
void pulse()
{
      porta.0 = 1;
      microdelay(2);
      porta.0 = 0;
}
//******************************************************************************
void stop()
{
      porta.0 = 1;
      microdelay(50);
      porta.1 = 1;
}
//******************************************************************************
void start()
{
      porta.1 = 0;      // sda à 0
      microdelay(50);
      porta.0 = 0;      // scl à 0
      microdelay(20);
}
//******************************************************************************
void acknol()
{
      //trisa.1 = 1;
      pulse();
      //trisa.1 = 0;
}
//******************************************************************************
void acknolmaster()
{
      porta.1 = 0;
      pulse();
}
//******************************************************************************
void convertir()
{
      if ((tempera_h & 0x80 ) == 0x00)     // test du signe de la température
      {
            signe = 0x00;     // température positive ( led eteinte )
      }
      else
      {
            signe = 0x01;     // température négative ( led allumée )
            tempera_h --;
            a = ~tempera_h;
            tempera_h = a;
      }
```

```
        if ( tempera_h < 10 )
        {
                affu = unite[tempera_h];
                affd = 0x00;

        }
        else
        {
                j = 0;
                do
                {
                        tempera_h = tempera_h - 10;
                        j ++;
                }
                while ( tempera_h >= 10 );
                affu = unite[tempera_h];
                affd = unite[j];
        }

        if ((tempera_l & 0x80 ) == 0x80)
        {
                affdecimal = 0x6D;
        }
        else
        {
                affdecimal = 0x3F;
        }

}
//******************************************************************************
void afficher()
{
        for (i=0;i<100;i++)
        {
                porta.2 = 1;
                portb = affdecimal;
                delay(1);
                porta.2 = 0;

                porta.3 = 1;
                portb = affu;
                delay(1);
                porta.3 = 0;

                porta.4 = 1;
                portb = affd;
                delay(1);
                porta.4 = 0;

                if (signe == 0x01)
                {
                        portb.7 = 1;
                        delay(1);
                }
                else
                {
                        portb.7 = 0;
                }
        }
}
//******************************************************************************
// fin du texte
```