

Programmes Arduino Minitel

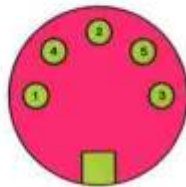
Librairies

Libraries Minitel1B_Soft

1.1 Prise mécanique

La prise péri-informatique est du type DIN 5 broches femelle sur laquelle sont disponibles les signaux suivants :

- **broche 1** : réception des données par le terminal (signal Rx) ;
- **broche 2** : masse ;
- **broche 3** : émission de données par le terminal (signal Tx) ;
- **broche 4** : périphérique en transmission (signal PT) ;
- **broche 5** : sortie alimentation disponible pour les périphériques. Cette fonction n'est pas disponible sur les versions dont l'identification porte les références Cu2 à Cu4 incluses.



Prise femelle vue de face

- prise Arduino D2 sur 3 minitel
- prise Arduino D3 sur 1 minitel
- prise Arduino GND sur 2 minitel

Arduino_Minitel.ino

```
#include <SoftwareSerial.h>
SoftwareSerial mySerial(2, 3);

byte gauche = 8;
byte droite = 9;
byte bas = 10;
byte haut = 11;
byte debutDeLigne = 13;
byte hautGauche = 30;
byte hautGaucheEfface = 12;
byte separateurDeSousArticle = 31;
byte remplissageEspace = 24; //Remplit le reste de la rangée avec des
espaces
byte CBleu = 68; // caractère niveau gris bleu
byte CBlanc = 71; // caractère couleur blanche
byte Clignote = 72 ; // caractère clignote
byte Fixe = 73 ; // caractère fixe
byte NormalH = 76 ; // taille caractère normal
byte DoubleH = 77 ; // double hauteur
byte Ligne = 90 ; // caractère souligné
byte SLigne = 89; // annule souligné
```

```
short incomming;
char inascii = » »;
short outcomming;
int TS = 0; // touche spéciale
String TSS = « »; // touche spéciale texte

void setup() {

Serial.begin(1200); // port serie vers le PC
mySerial.begin(1200); // port serie vers le minitel

mySerial.write(hautGaucheEfface); //efface l'écran
// serialprint7(0x0E); // passe en mode graphique
delay(500);
sendMessage(« BONJOUR »);
CR();
sendMessage(« BONSOIR »);
Gauche(3);
sendMessage(« REBONSOIR »);
Droite(3);
delay(1000);
sendMessage(« JOUR »);
CR();
ESC(Clignote);
sendMessage(« BONJOUR »);
ESC(Fixe);
CR();
CR();
ESC(DoubleH);
sendMessage(« BONJOUR »);
CR();
ESC(NormalH);
ESC(CBleu);
sendMessage(« BONJOUR »);
CR();
ESC(CBlanc);
sendMessage(« BONJOUR »);
CR();
ESC(Ligne);
sendMessage( » BONJOUR »);
CR();
ESC(SLigne);
sendMessage(« BONJOUR »);
CR();
Serial.println( » « );
}

char modifyParity(char c) {
char i = 1 << 6;
```

```
boolean p = false;
c &= B01111111;
while (i) {
  if (c & i) {
    p = !p;
  }
  i >>= 1;
}
c |= p << 7;
return c;
}

void sendMessage(char *msg) {
  int i = 0;
  while (msg[i]) {
    serialprint7(msg[i]);
    i++;
  }
  Serial.write(msg);
  Serial.flush();
}

void serialprint7(byte b) // permet d'ecrire en 7 bits + parité sur le
software serial
{
  boolean i = false;
  for (int j = 0; j < 8; j++)
  {
    if (bitRead(b, j) == 1) i = !i; //calcul de la parité
  }
  if (i) bitWrite(b, 7, 1); //écriture de la partié
  else bitWrite(b, 7, 0); //écriture de la partié
  mySerial.write(b); //écriture du byte sur le software serial
}

void Gauche(int g) {
  for (int i = 0; i <= g; i++) {
    serialprint7(9);
  }
}

void Droite(int g) {
  for (int i = 0; i <= g; i++) {
    serialprint7(8);
  }
}

void Haut(int g) {
  for (int i = 0; i <= g; i++) {
    serialprint7(11);
  }
}
```

```
}

void ESC(int c){
serialprint7(27);
serialprint7(c);
}

void CR() {
serialprint7(13);
serialprint7(10);
}

void loop() //tout ce que je recois sur le port serie, je le renvoi sur
le software serial
{

// Serial.println(« loop »);
if (Serial.available()) {
outcomming = Serial.read();
Serial.print(« saisie arduino: »);
Serial.println (outcomming);
// serialprint7(incomming);
serialprint7(outcomming);
}

if (mySerial.available()) {
incomming = mySerial.read() & B01111111; // ignore parity check //
ignore parity check
Serial.print(« saisie minitel : »);
inascii = char(incomming);
Serial.println (inascii);
if (TS == 1) {
touchespeciales();
TS = 0;
}
if (incomming == 19) {
TS = 1;
}

}

}

void touchespeciales() {
switch (incomming) {
case 70:
Serial.println (« Sommaire »);
TSS = « Sommaire »;
break;
case 69:
```

```
Serial.println (« Annulation »);  
TSS = « Annulation »;  
break;  
case 66:  
Serial.println (« Retour »);  
TSS = « Retour »;  
break;  
case 67:  
Serial.println (« Repetition »);  
TSS = « Repetition »;  
break;  
case 68:  
Serial.println (« Guide »);  
TSS = « Guide »;  
break;  
case 71:  
Serial.println (« Correction »);  
TSS = « Correction » ;  
break;  
case 72:  
Serial.println (« Suite »);  
TSS = « Suite »;  
break;  
case 65:  
Serial.println (« Envoi »);  
TSS = « Envoi »;  
break;  
case 89:  
Serial.println (« Connexion »);  
TSS = « Connexion »;  
break;  
  
}  
}
```

From:

<https://www.magenealogie.chanterie37.fr/www/fablab37110/> - Castel'Lab le Fablab MJC de Château-Renault

Permanent link:

<https://www.magenealogie.chanterie37.fr/www/fablab37110/doku.php?id=start:arduino:minitel:programme&rev=1651224524>

Last update: 2023/01/27 16:08

