6. TRANSMISSIONS. LIAISON SERIE

Sens de transmission

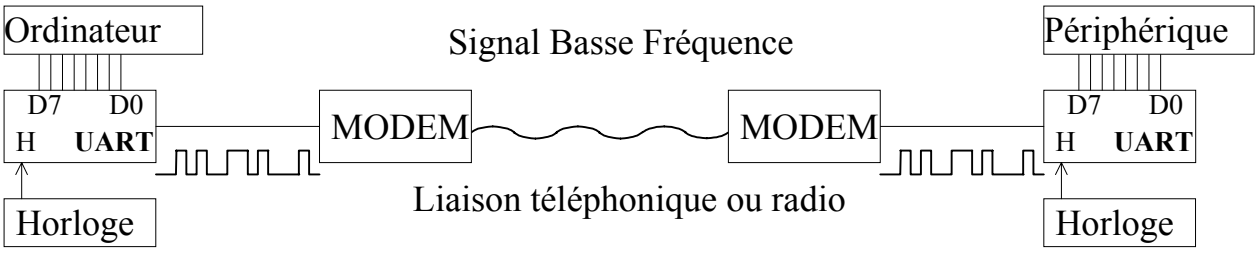
Simplex Emetteur ⟶ Récepteur

Half-duplex Emetteur ⟶ Récepteur
ou Récepteur ⟵ Emetteur

Full-duplex Emetteur ⟶ Récepteur
et Récepteur ⟵ Emetteur

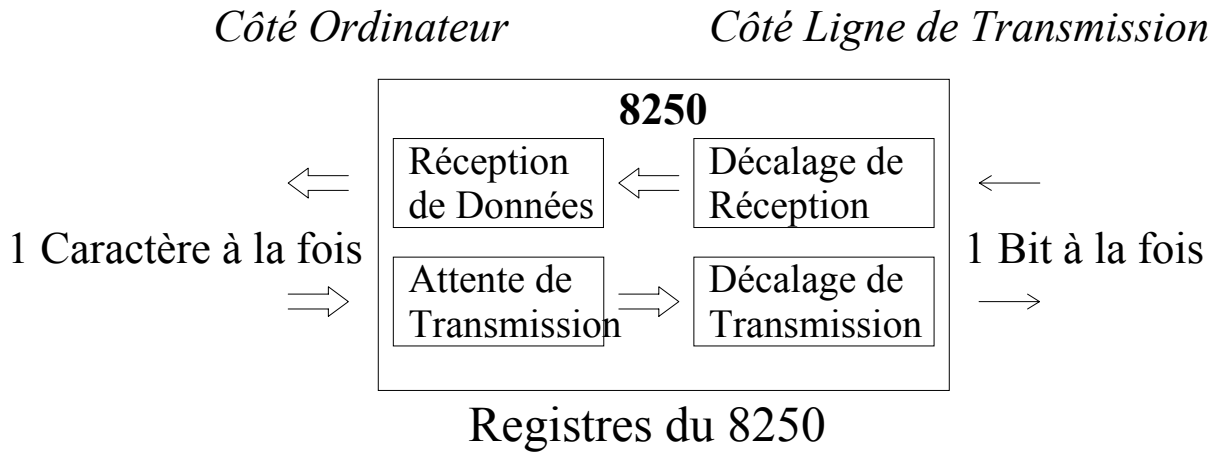
Modems

Liaison > 100 mètres → modulation

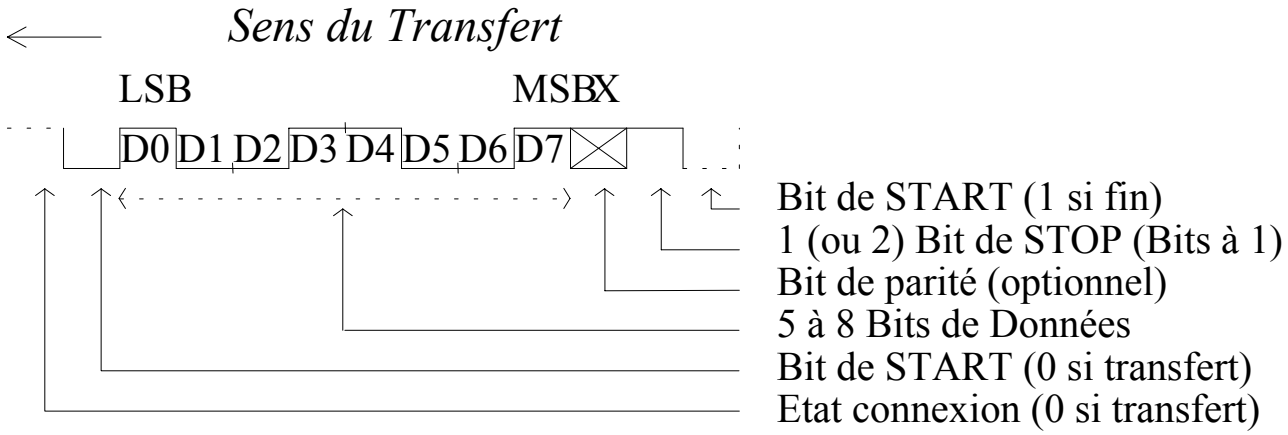


L'interface série d'entrées/sorties

Registres émission/réception du composant 8250 INTEL



Format de transmission série asynchrone



PROGRAMMATION DE LA LIAISON SERIE

Programmation par programmation directe du contrôleur série 8250

Adresses des portes COM1 du composant contrôleur d'E/S série (8250)

Adresse de port d'E/S	*Signification*	*Lecture - Ecriture*
3F8h	**Registre de données du 8250 / Registre** (8 bits L) **des vitesses de transmission**	L - E / E
3F9h	Registre de masques d'interruption du 8250 / **Registre** (8 bits L) **des vitesses de transmission**	E / E
3FAh	Registre d'interruptions du 8250	L
3FBh	**Registre de contrôle du 8250**	E
3FCh	**Registre de contrôle du modem du 8250**	E
3FDh	**Registre d'état de la ligne série du 8250**	L
3FEh	Registre d'état du modem du 8250	L

Registre de contrôle du 8250 (adresse 3FBh)

Bit	*Signification*	*Codage*
7 (MSB)	**Sélection des registres**	0 = Registres données - masques 1 = Registre vitesses de transmission
6	**Interruption Break**	0 = Active 1 = Inactive
5	**Parité fixe**	0 = Pas de parité fixe 1 = Parité fixe
4	**Parité**	0 = Paire 1 = Impaire
3	**Parité**	0 = Pas de parité 1 = Parité
2	**Nombre de bits de stop**	0 = 1 bit d'arrêt 1 = 2 bits d'arrêt
1-0	**Nombre de bits de données**	00 = 5 bits de données 01 = 6 bits de données 10 = 7 bits de données 11 = 8 bits de données

Registres des vitesses de transmission (adresses 3F8h et 3F9h)

Vitesse de transmission	*Valeur de l'octet à l'adresse 3F8h*	*Valeur de l'octet à l'adresse 3F9h*
110 bauds (≡ bits/s)	17h	04h
150 bauds	80h	01h
600 bauds	C0h	00h
1 200 bauds	60h	00h
1 800 bauds	40h	00h
2 400 bauds	30h	00h
3 600 bauds	20h	00h
4 800 bauds	18h	00h
9 600 bauds	0Ch	00h

Registre de contrôle du modem du 8250 (adresse 3FCh)

Bit	*Signification*
7 (MSB)	**0**
6	**0**
5	**0**
4	**0**
3	**Out 2**
2	**Out 1**
1	Request To Send (RTS)
0 (LSB)	Data Terminal Ready (DTR)

Registre d'état de la ligne série du 8250 (adresse 3FDh)

Bit	*Signification*
7 (MSB)	0
6	**(EMITTER) = 1 si donnée précédente envoyée**
5	**(EMITTER) = 1 si donnée précédente envoyée**
4	Break Interrupt (BI)
3	Framing Error (FE)
2	Parity Error (PE)
1	(RECEIVER) = 1 si donnée non lue avant nouvelle donnée reçue
0 (LSB)	**(RECEIVER) = 1 si donnée complète reçue**

Organigramme de programmation des portes d'E/S liaison série

I. Initialisation de la communication série RS232

a) Initialisation du registre de contrôle du modem

. Bits 2 à 7 du registre R de contrôle du modem (3FCh) ← 0

. Bits 0 à 1 inchangés

(R) AND (0000 0011 B) → 3FCh

b) Initialisation du port série

b1) Sélection du registre des vitesses de transmission

. Bit 7 du registre R de contrôle (3FBh) ← 1

(R) OR (1000 0000 B) → 3FBh

b2) Sélection de la vitesse de transmission

ex. : 110 bauds : 17h →3F8h et 04h →3F9h

17h → 3F8h
04h → 3F9h

b3) Sélection du registre des données

. Bit 7 du registre R de contrôle (3FBh) ← 0

. Données 8 bits, 1 bit STOP, pas de parité, Break active :

0000 0011 B = 03h → 3FBh

II. Communication série RS232

c)Envoi ou réception de caractères :

caractère à émettre → 3F8h ou : caractère reçu ← 3F8h

Programme d'exemple C sous LabWindows/CVI (émission/réception d'un caractère)

/* Programmation de la liaison série par programmation directe du 8250 (BAS-NIVEAU) */

// Fonctions RS232 developpees :

void **OpenComConf**(void); *// Initialisation de la ligne série*

void **ComWrtByt**(char carE);
// Emission d'1 caractère si caractère précédent fini d'être émis

char **ComRdByt**(void);
// Réception d'1 caractère (non bloquant si pas de caractère à
// recevoir, mais alors caractère précédemment reçu de nouveau
// reçu)

int **GetIn**(void); *// Y-a-t-il un caractère à recevoir ?*

```
_
void OpenComConf ()
{
char octet;
// Initialisation Registre Controle du Modem
   octet = inp(0x3FC);
   octet = octet & 0x03;
   outp(0x3FC, octet);
// Initialisation Port Serie (COM1)
   // Selection du Registre des Vitesses de Transmission
   octet = inp(0x3FB);
   octet = octet | 0x80;
   outp(0x3FB, octet);
   // Selection de la Vitesse de Transmission : 110 bits/s
   outp(0x3F8, 0x17);
   outp(0x3F9, 0x04);
   // Selection du Registre des Donnees :
   // 8 bits de donnees, 1 bit d'arret, pas de parite, interruption par Break active
   outp(0x3FB, 0x03);
}
```

—

```
_
void ComWrtByt (char carE)
{
int bit5, bit6;
// Attente transmission caractere precedent ?
   do
   {
      bit5 = inp(0x3FD) & 0x20;
      bit6 = inp(0x3FD) & 0x40;
   }
   while(bit5 == 0 || bit6 == 0);
// Envoi d'un caractere
   outp(0x3F8, carE);    // Caractere a emettre
}
```

_

```
_
char ComRdByt ()
{
char carR;
   carR = inp(0x3F8);       // Caractere a recevoir
   return (carR);
}
```

_

```
_
int GetIn(void)
{
int bit0;
char octet;
   octet = inp(0x3FD);       // Donnee prete ? (Registre d'Etat ligne serie)
   bit0 =  octet & 0x01;  // Bit0 = 1 si Donnee (complète) recue
   return (bit0);
}
```

—

Programme d'exemple C sous LabWindows/CVI (émission/réception d'un caractère)

/* Programmation de la liaison série par librairie RS232 CVI (HAUT-NIVEAU) */

// Fonctions de la librairie RS232 de CVI utilisées :

OpenComConfig(); *// Initialisation de la ligne série*

ComWrtByte(); *// Emission d'1 caractère*

ComRdByte(); *// Réception d'1 caractère*
// (bloquant si pas de caractère à recevoir)

GetInQLen(); *// Nombre de caractères en attente de réception*

CloseCom(); *// Fin de communication. Clôture de la ligne série*

TP 6. TRANSMISSIONS. LIAISON SERIE

1. Programmation de la ligne série (Haut-niveau CVI)

. Emission/Réception d'un seul caractère (Numeric) en duplex

2. Programmation de la ligne série (Bas-niveau 8250)

. Emission/Réception d'un seul caractère (Numeric) en duplex

Facultatif :

3. Programmation de la ligne série (Haut-niveau CVI)

. Emission/Réception d'une séquence de caractères en duplex

4. Programmation de la ligne série (Bas-niveau 8250)

. Emission/Réception d'une séquence de caractères en duplex