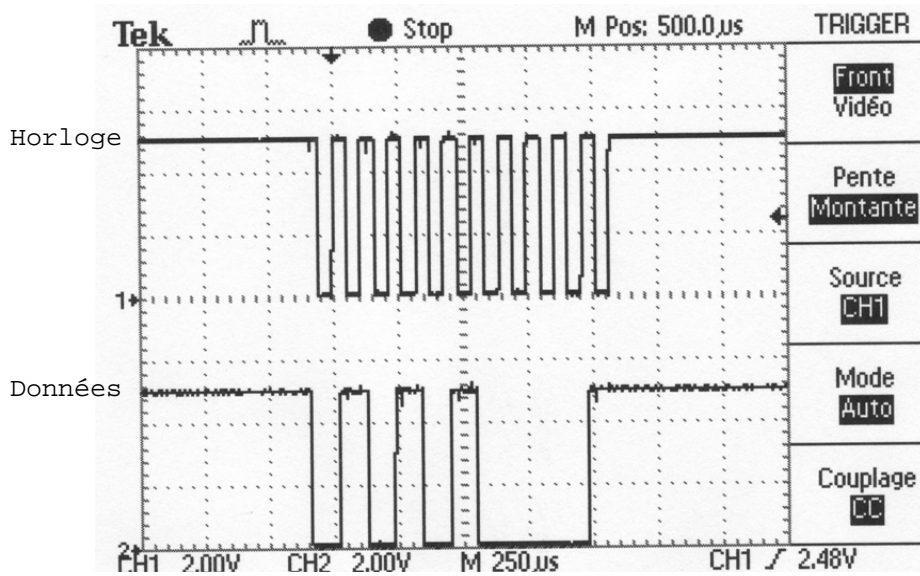
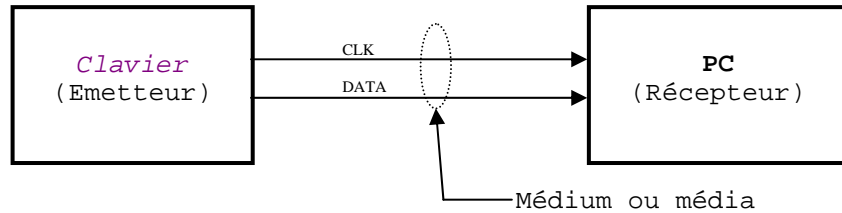


## Exercice 1 : Décodage d'une trame transmise par un clavier de PC

Lorsqu'un clavier émet un code vers un PC il se comporte en émetteur et le PC en récepteur.



Les différents bits de la trame des données sont acquis par le PC sur les fronts descendant du signal d'horloge.

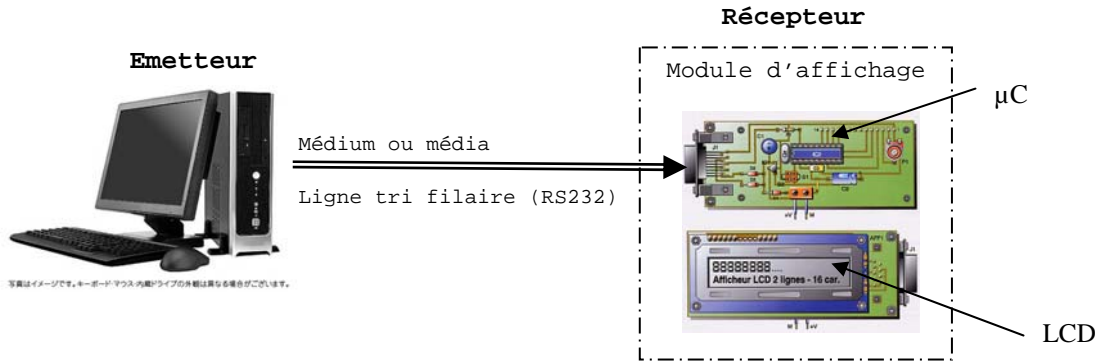
**Q1.** Décodez la trame des données et complétez le tableau ci-dessous.

Format général	<b>En-tête</b>	<b>Données applicatives</b>							<b>Termineur</b>	
Format d'une trame IBM	Start	Données 'Code IBM du caractère'							Parité	Stop
		b0	b1	b2	b3	b4	b5	b6	b7	
Bits de la trame IBM										

**Q2.** Reconstituez le code du caractère transmis. Retrouvez le caractère transmis dans la table des caractères IBM (scan code set 2) de l'annexe 1.

	b7	b6	b5	b4	b3	b2	b1	b0
Le code hexadécimal de la touche généré par le clavier est :								
Le caractère transmis par le clavier est :								

**Exercice 2 : Décodage d'une trame transmise sur une liaison RS232**



Il n'y a pas de signal de synchronisation dans une transmission asynchrone. L'émetteur et le récepteur possède chacun une horloge. Pour que le signal transmis soit correctement reçu, les horloges doivent avoir la même fréquence. Celle-ci est réglable aussi bien sur l'émetteur que sur le récepteur.

On ne parle pas de fréquence en Hertz mais de **débit binaire** en bit/s. Lorsque débit binaire est connu, il est possible de calculer le temps bit (durée d'un bit).

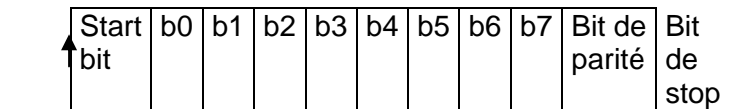
Exemple : à 300bits/s correspond un temps bit de  $1/300 = 3,33\text{ms}$ .

En connaissant le temps bit, la composition de la trame du message et le nombre de bits dans le message, il est possible de retrouver l'information transmise.

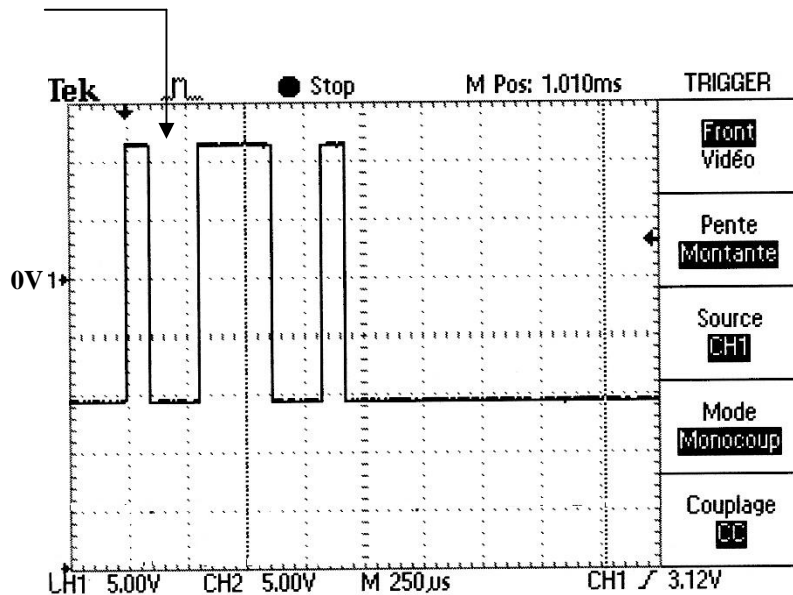
Exemple :

Le débit binaire du message contenu dans le signal ci-dessous est 9600bits/s.

La trame est organisée comme ci-dessous.

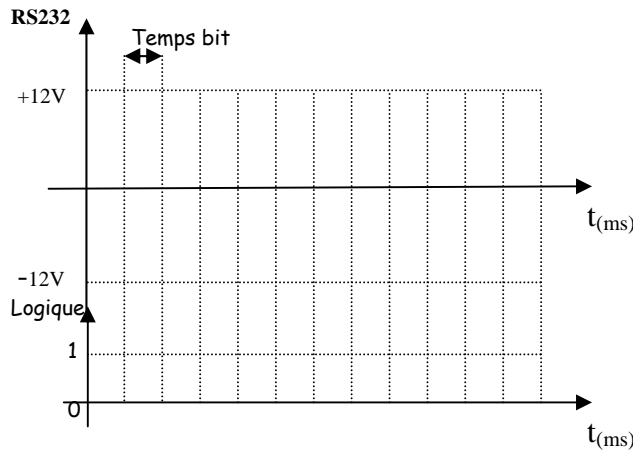


Début de la trame du message



**Q3.** Calculez le temps bit, puis calculer la durée du message.

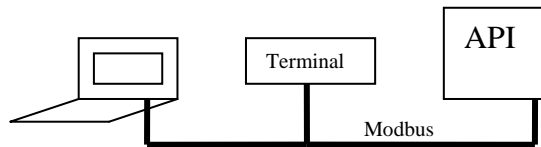
**Q4.** Dessinez la trame RS232 dans le repère ci-dessous et le signal logique correspondant sachant que +12V = « 0 » logique et -12V = « 1 » logique. Entourez l'en-tête, le terminateur et les données applicatives.



**Q5.** Décodez la trame du message et donnez le caractère ASCII transmis (Annexe 2).

**Exercice 3 : Application RS232 (transmission asynchrone)**

Un terminal de données est connecté sur un BUS de terrain type **Modbus**.



Le **protocole** de ligne est de type **ASCII asynchrone**.

Le **format** des trames est le suivant :

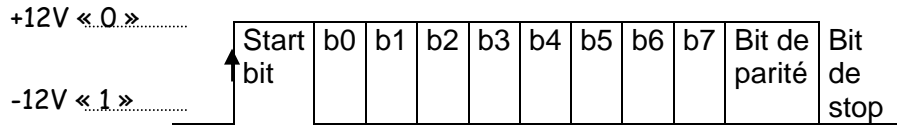
En tête	Adresse du destinataire	Code fonction	Données	LRC	Terminateur	
:	2 caractères	2 caractères	N*2 caractères	2 caractères	CR	LF

Le contenu du champ dépend du code fonction

**Q6.** Entourez l'en-tête, le terminateur et les données applicatives sur le schéma ci-dessus.

Dans une transmission asynchrone **type RS232**, le récepteur se synchronise à **chaque caractère** transmis lors du front montant du bit de start.

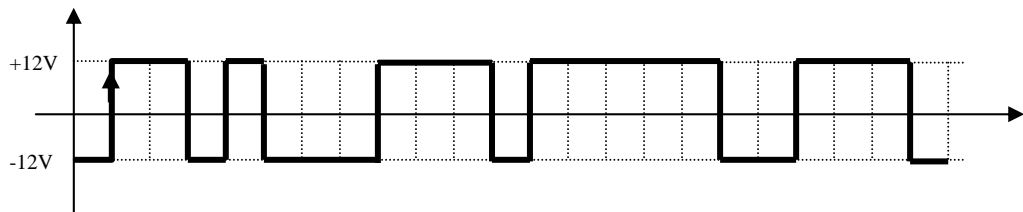
Un caractère est émis comme ci-dessous :



Ce type de transmission étant configurable, elle est paramétrée ici avec :

- 8 bits de donnée (bit0 à bit7) (Ex : si b7b6b5b4b3b2b1b0 = 01000111, le caractère transmis est G).
- parité paire (Ex : dans le cas précédent le bit de parité est à zéro).
- 1 bit de stop

**Q7.** Le signal ci-dessous contient deux caractères. Décodez-le et donnez la position des deux caractères dans la trame.



Trame du premier caractère

Start	Caractère								parité	stop	Hexa	ASCII
	b7	b6	b5	b4	b3	b2	b1	b0				

Trame du deuxième caractère

Start	Caractère								parité	stop	Hexa	ASCII
	b7	b6	b5	b4	b3	b2	b1	b0				

**Exercice 4 : Le bus CAN : un standard pour relier les équipements de confort d'un véhicule**

Dans sa version **basse vitesse** (125kbts/s), le **bus CAN (Controler Area Network)** est utilisé dans l'automobile pour relier les **équipements de confort** (éclairage, lève-vitre, rétroviseur etc.)

La figure 1 ci-dessous représente quatre équipements : deux moteurs de lève vitre, une console de commande et un tableau de bord.

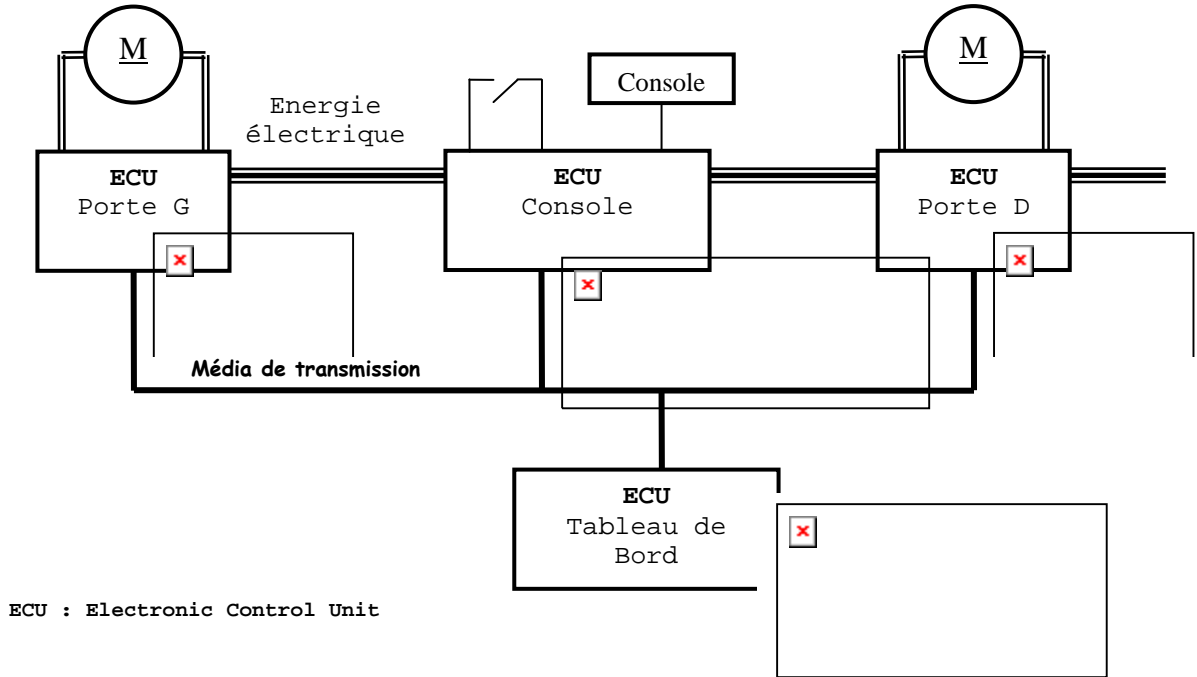
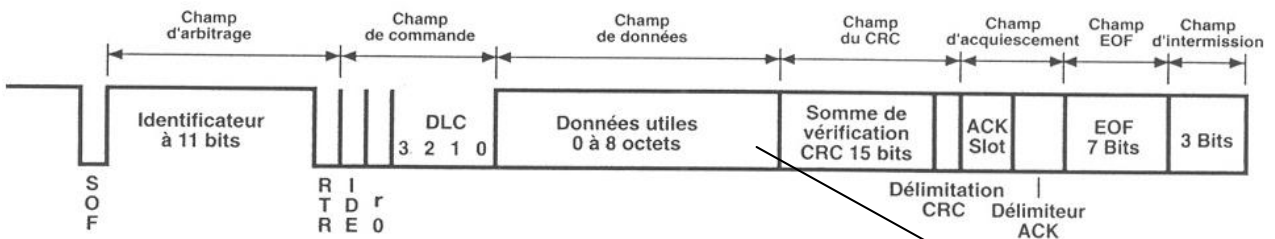


figure 1

Ces éléments communiquent par l'intermédiaire d'un bus CAN composé d'un **média de transmission** (fils électriques) et d'unités de contrôle électroniques (**ECU**).

Le format d'une trame CAN est représenté ci-dessous :



**Q8.** Entourez l'en-tête, le terminateur et les données applicatives sur le schéma ci-dessus.

On donne les messages ci-dessous

227.214972	1	111	Gateway_2	Tx	8	4c 13 01 20 14 06 f4 01
Voltage		50 V	[		1f4]	
PetrolLevel		6	[		6]	
EngSpeed		5152	[		1420]	
CarSpeed		137.5	[		113]	
EngineTemp		76	[		4c]	
116.930236	1	604	TP_ConsoleMessage	Tx	8	05 05 53 57 52 20 33 00
<OTP>					5	53 57 52 20 33
228.300132	1	1a1	Console_2	Tx	2	02 01

La tension de l'alternateur (Voltage) peut être déterminée à partir de la relation  $Voltage = k \cdot Tx$

**Q9.** Déterminez le coefficient k.

Annexe 1

Symbole	Scan-Code-Set 1		Scan-Code-Set-2		Scan-Code-Set-3 Code
	Make	Break	Make	Break	
1	02	82	16	F0-16	16
2	03	83	1E	F0-1E	1E
3	04	84	26	F0-26	26
4	05	85	25	F0-25	25
Q	10	90	15	F0-15	15
W	11	91	1D	F0-1D	1D
E	12	92	24	F0-24	24
R	13	93	2D	F0-2D	2D
T	14	94	2C	F0-2C	2C

Annexe 2

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	DLE	SP	0	@	P	'	p				°	À	Ð	à	ð
1	SOH	DC1	!	1	A	Q	a	q			ı	±	Á	Ñ	á	ñ
2	STX	REP	"	2	B	R	b	r	,		ı	²	Â	Ò	â	ò
3	ETX	SEP	#	3	C	S	c	s	f		ı	³	Ã	Ó	ã	ó
4	EOT	DC4	\$	4	D	T	d	t	"		ı	´	Ä	Ô	ä	ô
5	ENQ	NAK	%	5	E	U	e	u	...	•	ı	µ	Å	Õ	å	õ
6	ACK	SYN	&	6	F	V	f	v	†	—	ı	¶	Æ	Ö	æ	ö
7	BEL	ETB	'	7	G	W	g	w	‡	—	ı	·	Ç	×	ç	÷
8	BS	CAN	(	8	H	X	h	x	‰	™	ı	¸	È	Ø	è	ø
9	HT	SS2	)	9	I	Y	i	y	‰	™	ı	¹	É	Ù	é	ù
A	LF	SUB	*	:	J	Z	j	z	‰	™	ı	º	Ê	Ú	ê	ú
B	VT	ESC	+	;	K	[	k	{	‰	™	ı		Ë	Û	ë	û
C	FF	FS	,	<	L	\	l		‰	™	ı	¼	Ì	Ü	ì	ü
D	CR	GS	-	=	M	]	m	}	‰	™	ı	½	Í	Ý	í	ý
E	SO	RS	.	>	N	^	n	~	‰	™	ı	¾	Î	Þ	î	þ
F	SI	US	/	?	O	_	o	DEL	‰	™	ı	¿	Ï	Ë	ï	ÿ