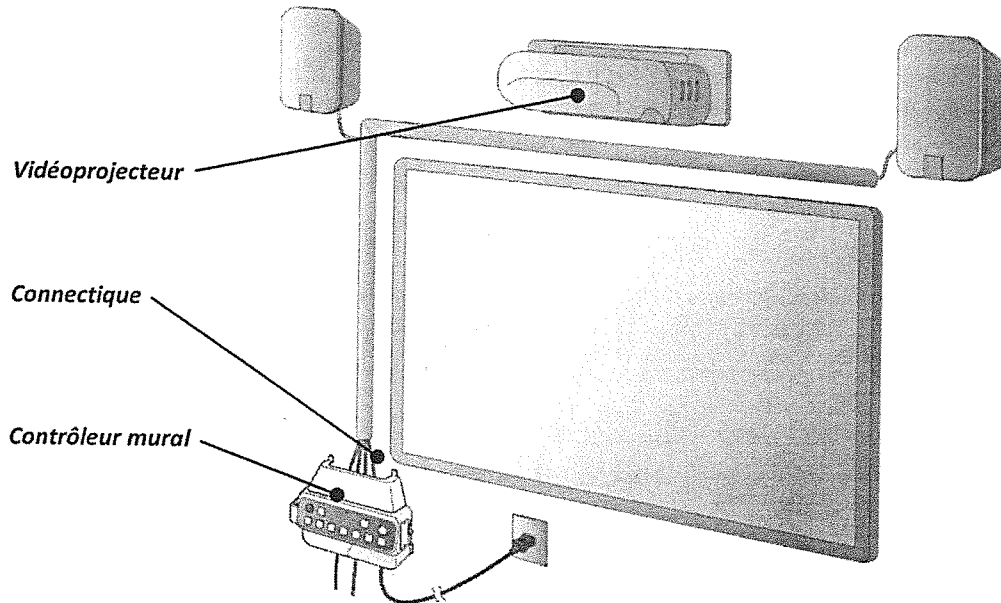


Cours – Transmission numérique de l'information

Cas de la commande d'un vidéoprojecteur par liaison série RS-232

I. Mise en situation

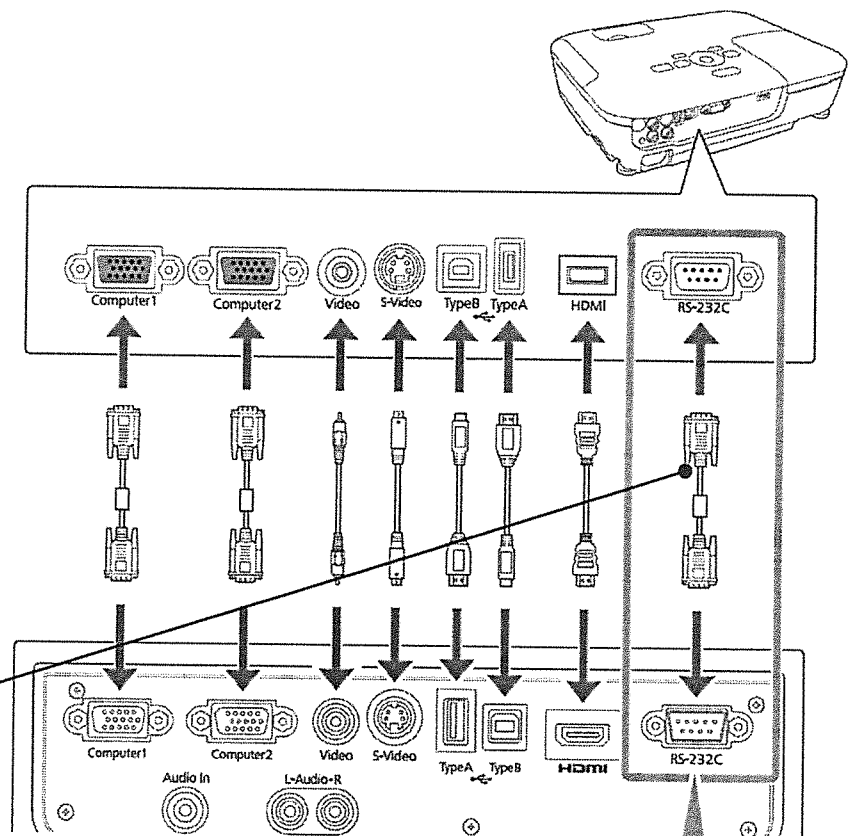
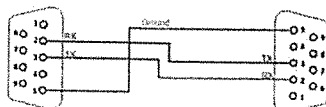
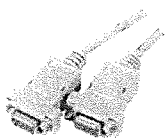


On s'intéresse ici à la commande d'un vidéoprojecteur (fixé au plafond, donc non accessible) depuis un contrôleur (fixé au mur, donc accessible).

Des informations numériques sont émises par le contrôleur à destination du vidéoprojecteur afin, par exemple, de :

- l'éteindre
- l'allumer
- sélectionner la source vidéo
- etc...

Ces informations numériques sont transmises selon la **norme RS-232** via une connectique se présentant sous la forme de connecteurs DE-9 (généralement appelés **DB9**) :



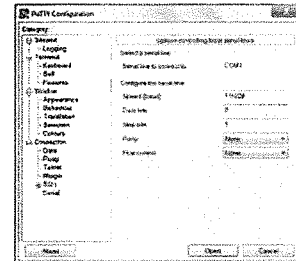
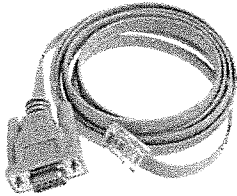
La norme **RS-232** est une norme standardisant (d'un point de vue électrique, mécanique et protocole) un bus de communication de type **série**.

Disponible sur presque tous les PC jusqu'au milieu des années 2000, il est communément appelé le « **port série** ».

Cette norme RS-232 reste encore très utilisée dans l'industrie, notamment grâce à **sa robustesse et à sa simplicité**. Cependant, elle est de plus en plus remplacée par la norme USB.

La norme RS-232 couvre plusieurs recommandations liées aux **caractéristiques électriques** (niveaux de tension, débits possibles, ...), au **protocole** utilisé ainsi qu'au type de **connectique**.

Fréquemment, les périphériques réseau (routeurs, commutateurs, etc...) sont équipés d'un port RS-232 à partir duquel il est possible de les configurer via un **câble console** et un logiciel terminal comme **Putty** :



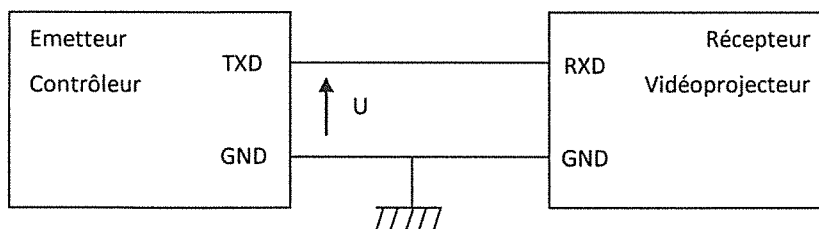
La norme **RS-232** permet une communication **série**.

En transmission **série**, les **bits**, des données numériques à transmettre, sont envoyés les uns à la suite des autres.

1. Caractéristiques électriques

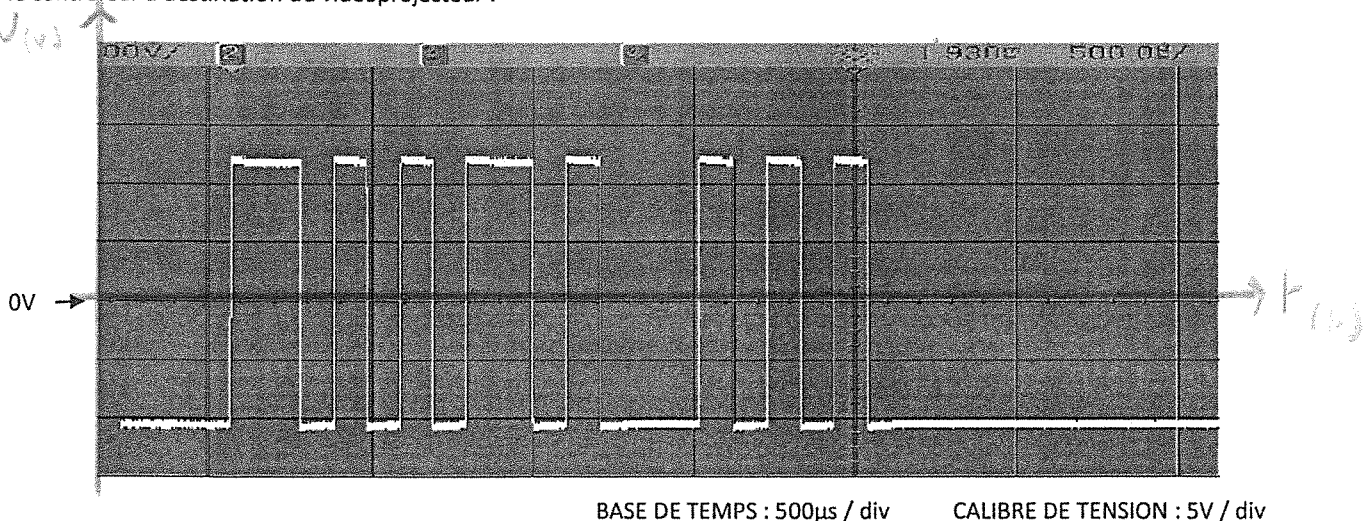
Câblage :

Dans le cas de la commande du vidéoprojecteur depuis le contrôleur, on peut considérer le schéma simplifié suivant :



Le contrôleur émet les données via son fil TXD et le vidéoprojecteur les reçoit via son fil RXD.

A l'aide d'un oscilloscope on a relevé l'évolution de la tension U dans le temps à l'instant où une commande d'allumage a été émise par le contrôleur à destination du vidéoprojecteur :

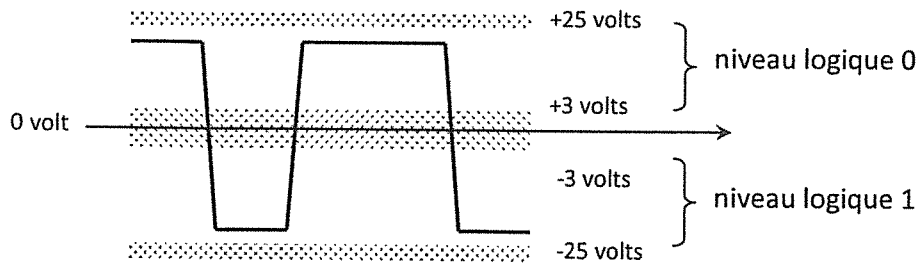


BASE DE TEMPS : 500 μ s / div

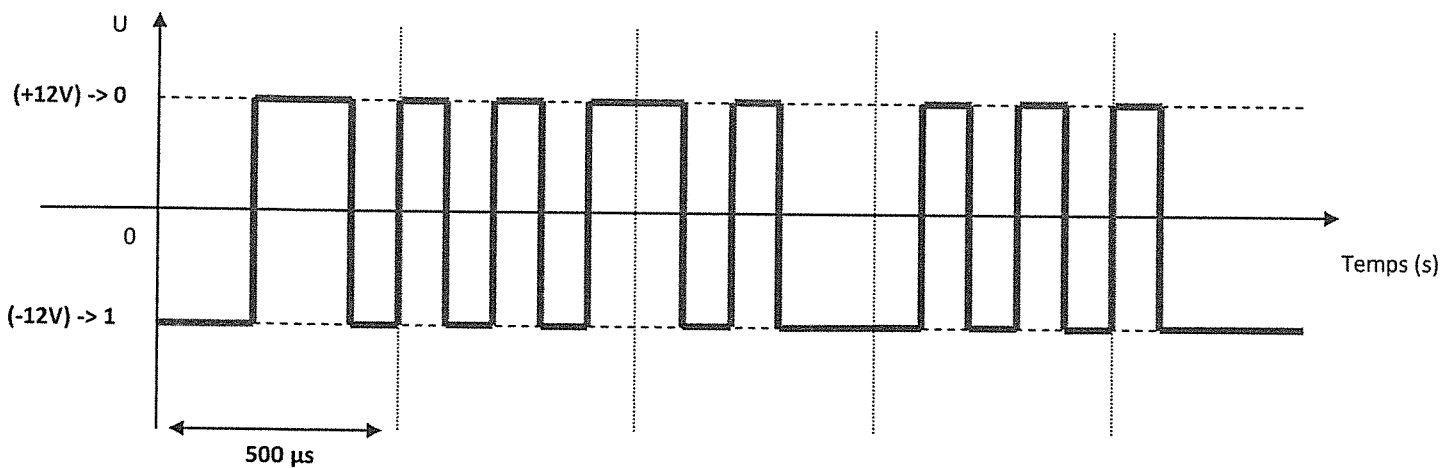
CALIBRE DE TENSION : 5V / div

Codage des bits et niveaux de tensions :

Une tension U comprise entre **+3 et +25V** correspond à un **0 logique** et une tension U comprises entre **-3V et -25V** correspondent à un **1 logique**. D'ordinaire, des niveaux de +12V et -12V sont utilisés.



Voici une représentation encore plus lisible de l'évolution de la tension U générée par le contrôleur dans le temps faisant apparaître les états logiques transmis :

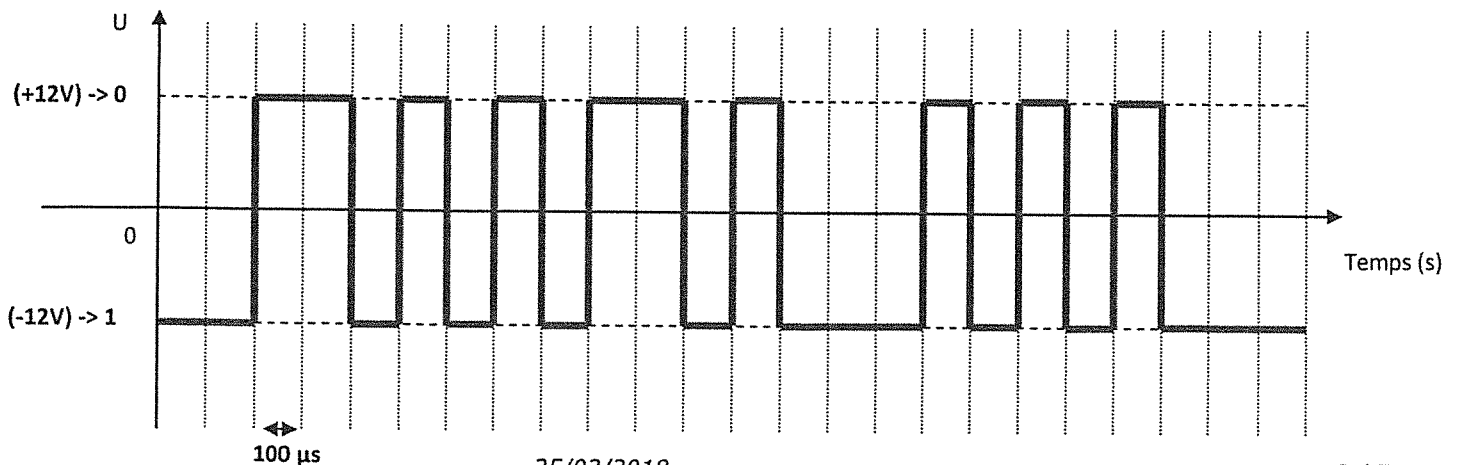
Débits

La spécification RS-232 prescrit des **débits** inférieurs à 20 000 bit/s : 19200 bits /s ; 9600 bits / s

Les documentations du contrôleur et du vidéoprojecteur spécifient que la liaison série RS-232 est configurée avec un débit de 9600bits/s.

La durée de transmission d'un bit est donc de : $1 / 9600 = 104\mu\text{s} \approx 100\mu\text{s}$

Voici une représentation faisant apparaître la durée de transmission d'un bit sur le signal électrique U généré par le contrôleur :



2. Protocole

La norme RS232 impose également le respect d'un **protocole de communication**.

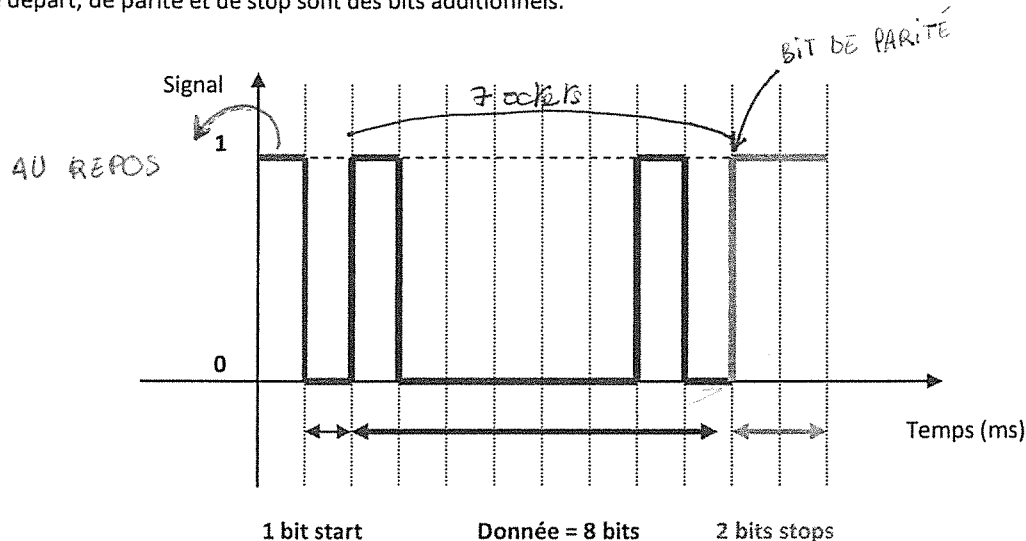
Un protocole étant la règle à respecter lors de la transmission de donnée.

La norme RS232 impose qu'une trame émise soit constituée de :

- 1 bit de départ ;
- 7 à 8 bits de données ;
- 1 bit de parité optionnel ;
- 1 ou 2 bits de stop.

Les bits de départ, de parité et de stop sont des bits additionnels.

Exemple :



Le bit de départ :

Le signal au repos (en absence de transmission de données) est au niveau logique 1.

Pour indiquer qu'une donnée va être transmise le signal passe au niveau logique 0 (bit de start).

Ce bit permet d'avertir le récepteur de la transmission d'une nouvelle donnée.

La longueur d'une donnée :

On peut configurer la transmission d'une donnée de 7 ou 8 bits. Le bit de donnée de poids faible est envoyé en premier suivi des autres.

↳ effet miroir

La parité (optionnel) :

On peut ajouter un bit de parité afin de détecter d'éventuelles erreurs de transmission. La parité peut être définie paire ou impaire.

- Parité paire : le bit ajouté à la donnée est positionné de telle façon que le nombre des états 1 soit pair sur l'ensemble donné + bit de parité.

ex : soit la donnée 11001011 contenant 5 états 1, le bit de parité paire est positionné à 1, ramenant ainsi le nombre de 1 à 6.

- Parité impaire : le bit ajouté à la donnée est positionné de telle façon que le nombre des états 1 soit impair sur l'ensemble donné + bit de parité.

ex : soit la donnée 01011011 contenant 5 états 1, le bit de parité est positionné à 0, laissant ainsi un nombre impair de 1.

Le bit de stop :

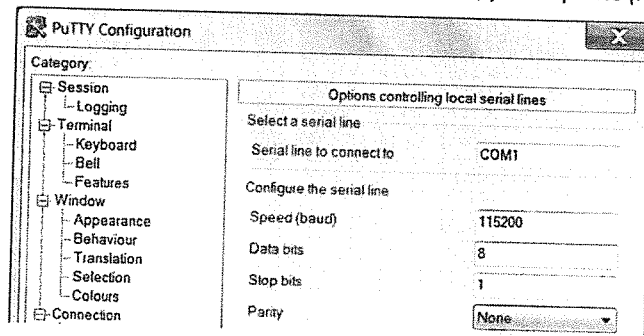
On configure un ou deux bits de stop de niveau logique 1 qui indiquent la fin de la transmission.

3. Pour pouvoir communiquer, les deux appareils connectés doivent partager les mêmes paramètres de transmission
Les paramètres sont parfois décrits sous forme condensée :

« 9600 712 » : 9600 bauds, 7 bits de données, parité impaire (I), 2 bits de stop.

« 115200 8N1 » : 115200 bauds, 8 bits de données, pas de parité (N), 1 bit de stop.

Exemple de configuration sur le terminal Putty (115200 bauds, 8 bits de données, pas de parité (N), 1 bit de stop) :

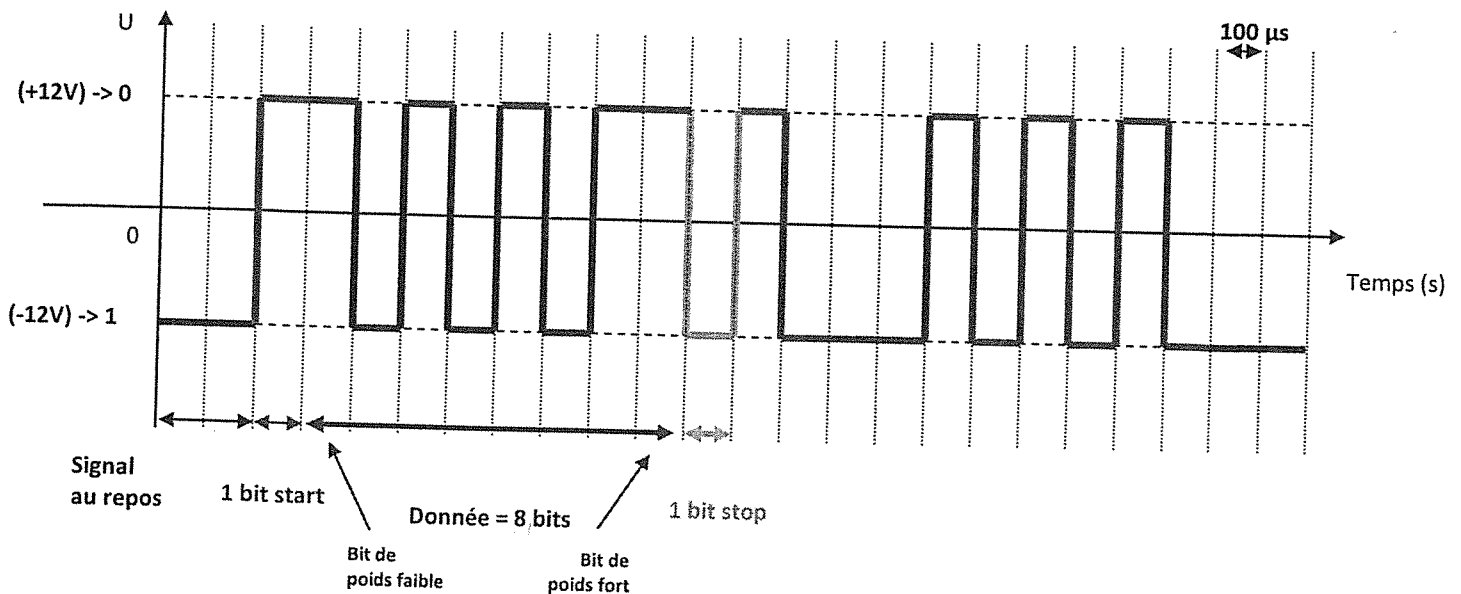


Les documentations du contrôleur et du projecteur spécifient que les paramètres de la liaison série sont les suivants : 9600 8 N 1
Soit :

- un débit de 9600bits/s
- 8 bits de données par trame
- Pas de bit de parité
- 1 bit de stop

Voici une représentation faisant apparaître les différents bits d'une des trames transmises via le signal électrique U généré par le contrôleur :

TRAME : succession de 0 et de 1



Soit la séquence :

START 01010100 STOP START 11101010 STOP

On "retourne" les octets ([b0 à b7] devient [b7 à b0]) :

00101010

01010111

4. Codage et transmission d'une information numérique : Caractère codé en ASCII

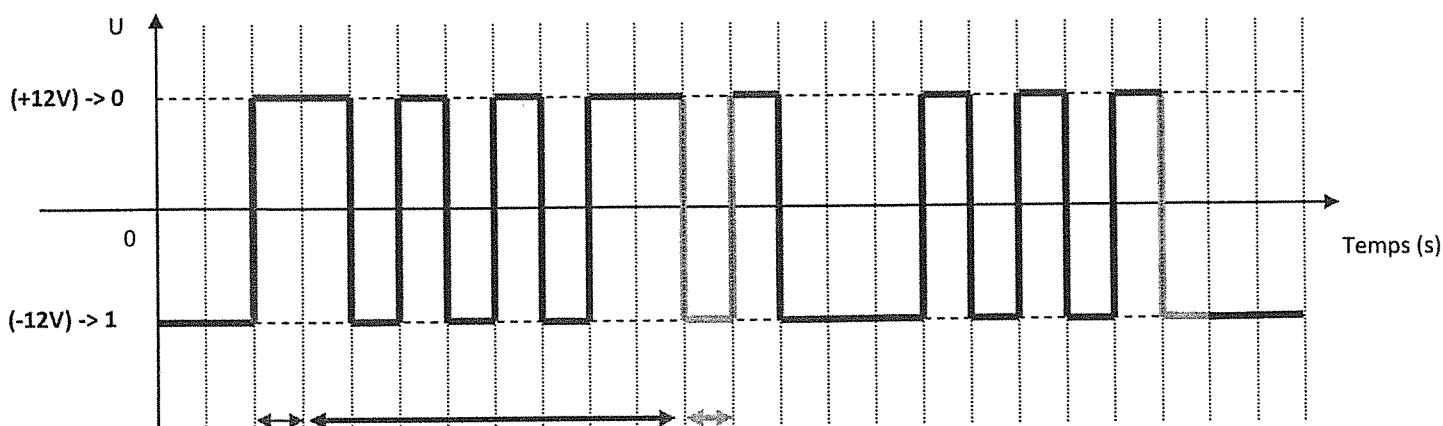
Un signal binaire ne peut prendre que deux états possibles 0 ou 1.

Alors comment représenter en binaire un caractère en vue de le transmettre ? Tous simplement en utilisant plusieurs '0' et plusieurs '1' qui assemblés constituent un code. Un code constitué de n bits peut représenter 2^n caractères différents.

Il existe de nombreux codes binaires. L'un d'entre eux, appelé *code ASCII*, est universellement utilisé pour représenter les caractères alphanumériques.

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`	128	80	Ç	160	A0	á	192	C0	É	224	E0	α						
1	01	Start of heading	33	21	!	65	41	A	97	61	a	129	81	ü	161	A1	í	193	C1	Ê	225	E1	β						
2	02	Start of text	34	22	"	66	42	B	98	62	b	130	82	é	162	A2	ó	194	C2	Ë	226	E2	Γ						
3	03	End of text	35	23	#	67	43	C	99	63	c	131	83	á	163	A3	ú	195	C3	Ï	227	E3	π						
4	04	End of transmit	36	24	\$	68	44	D	100	64	d	132	84	ä	164	A4	ñ	196	C4	Ï	228	E4	Σ						
5	05	Enquiry	37	25	%	69	45	E	101	65	e	133	85	å	165	A5	Ñ	197	C5	Ï	229	E5	σ						
6	06	Acknowledge	38	26	&	70	46	F	102	66	f	134	86	ä	166	A6	ª	198	C6	Ï	230	E6	μ						
7	07	Audible bell	39	27	'	71	47	G	103	67	g	135	87	ç	167	A7	º	199	C7	Ï	231	E7	τ						
8	08	Backspace	40	28	(	72	48	H	104	68	h	136	88	è	168	A8	¿	200	C8	Ï	232	E8	φ						
9	09	Horizontal tab	41	29	)	73	49	I	105	69	i	137	89	é	169	A9	¸	201	C9	Ï	233	E9	θ						
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j	138	8A	ê	170	AA	¸	202	CA	Ï	234	EA	Ω						
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k	139	8B	ÿ	171	AB	¸	203	CB	Ï	235	EB	δ						
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l	140	8C	ï	172	AC	¸	204	CC	Ï	236	EC	∞						
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m	141	8D	ì	173	AD	¸	205	CD	Ï	237	ED	∞						
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n	142	8E	ä	174	AE	¸	206	CE	Ï	238	EE	τ						
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o	143	8F	å	175	AF	¸	207	CF	Ï	239	EF	Π						
16	10	Data link escape	48	30	0	80	50	P	112	70	p	144	90	ê	176	BO	¸	208	DO	Ï	240	FO	≡						
17	11	Device control 1	49	31	1	81	51	Q	113	71	q	145	91	æ	177	B1	¸	209	D1	Ï	241	F1	±						
18	12	Device control 2	50	32	2	82	52	R	114	72	r	146	92	æ	178	B2	¸	210	D2	Ï	242	F2	≥						
19	13	Device control 3	51	33	3	83	53	S	115	73	s	147	93	ô	179	B3	¸	211	D3	Ï	243	F3	≤						
20	14	Device control 4	52	34	4	84	54	T	116	74	t	148	94	ö	180	B4	¸	212	D4	Ï	244	F4	∫						
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u	149	95	ö	181	B5	¸	213	D5	Ï	245	F5	∫						
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v	150	96	ù	182	B6	¸	214	D6	Ï	246	F6	÷						
23	17	End trans. block	55	37	7	87	57	W	119	77	w	151	97	ù	183	B7	¸	215	D7	Ï	247	F7	∞						
24	18	Cancel	56	38	8	88	58	X	120	78	x	152	98	ý	184	B8	¸	216	D8	Ï	248	F8	*						
25	19	End of medium	57	39	9	89	59	Y	121	79	y	153	99	ö	185	B9	¸	217	D9	Ï	249	F9	*						
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z	154	9A	Û	186	BA	¸	218	DA	Ï	250	FA	.						
27	1B	Escape	59	3B	;	91	5B	[	123	7B	{	155	9B	¸	187	BB	¸	219	DB	Ï	251	FB	√						
28	1C	File separator	60	3C	<	92	5C	\	124	7C		156	9C	¸	188	BC	¸	220	DC	Ï	252	FC	∞						
29	1D	Group separator	61	3D	=	93	5D	]	125	7D	}	157	9D	¸	189	BD	¸	221	DD	Ï	253	FD	*						
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~	158	9E	¸	190	BE	¸	222	DE	Ï	254	FE	■						
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□	159	9F	¸	191	BF	¸	223	DF	Ï	255	FF	□						

Voici une représentation faisant apparaître l'état des bits de la trame transmise via le signal électrique U généré par le contrôleur :



Soit la séquence :

START 01010100 STOP START 11101010 STOP

On "retourne" les octets ([b0 à b7] devient [b7 à b0]) :

00101010

01010111

Soit en hexadécimal :

2A

57

Correspondance avec le code ASCII :

2A code de *

57 code du caractère W

La lecture de la documentation du vidéoprojecteur confirme le fait que *W corresponde à une commande d'allumage !