

Transmission de l'information

Objectif(s) :

- Mettre en œuvre la communication série d'une carte ARDUINO avec le moniteur série.
- Identifier les paramètres d'une liaison série par la mesure.

La carte ARDUINO utilise le protocole de liaison série de type RS232 pour communiquer avec le Serial Monitor. Vous allez mettre en œuvre cette liaison et valider ses paramètres de fonctionnement par la mesure à l'aide d'un oscilloscope.

Activité : Mesurer la vitesse de transmission

```
void setup() {
    Serial.begin(19200);
}

void loop() {

    Serial.print("B");

    delay (2);

}
```

1) Rappeler (en 1 ligne) le rôle des instructions (fléchées)

→ définir le serial moniteur sur 19200 bauds

→ écrire sur le serial moniteur la lettre "B"

→ faire une pause de 0,002 secondes

- 2) Créer un nouveau programme, appelé 'TransmissionB', en recopiant le code ci-dessus.
 3) Quel paramètre doit être réglé sur le Serial Monitor pour recevoir correctement les données

Le serial.begin(19200) doit être de (9600) (regler le serial moniteur a 19200)

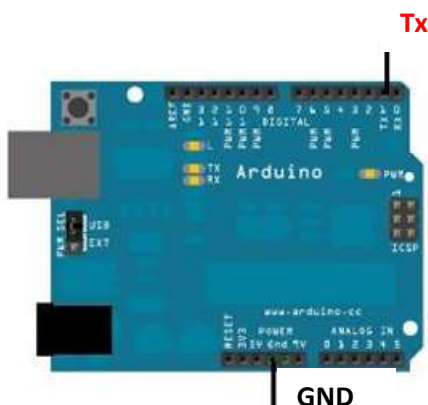
4) Valider la communication avec le Serial Monitor.

Vous observez :....

BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB.....

5) Visualisation de la transmission

Vous allez visualiser à l'oscilloscope l'information transmise (Tx) par la carte ARDUINO au Serial Monitor.



Connecter une sonde de mesures sur la broche **Tx** de la carte.

Relier le **GND** de la sonde au GND de la carte



Rappel : la vitesse de transmission d'une liaison série est définie par son nombre de bauds ou bits/seconde.

1) Etude théorique

a) Quelle est la vitesse de transmission utilisée dans notre programme, justifier votre réponse :

19200 bits/secondes

Serial.begin(19200);

b) Calculer la durée de transmission d'un bit

Sachant que :

Vitesse de transmission = 19200 bits/seconde

1 bit est transmis en : 0.0000521 s

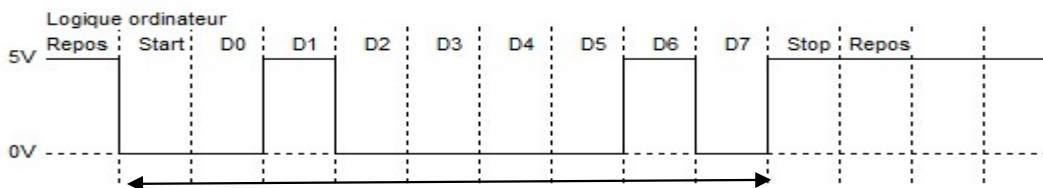
c) Calculer la durée de transmission d'un envoi

La transmission utilise 1 bit de START, 8 bits de données, 1 bit de STOP .

Calculer la durée théorique de transmission de l'envoi d'un caractère :

$$0.0000521 * 8 = 0.0004168 \text{ s}$$

Sachant que le caractère B correspond à cet envoi :



Calculer le temps de transmission, mesurable entre START et D7 :

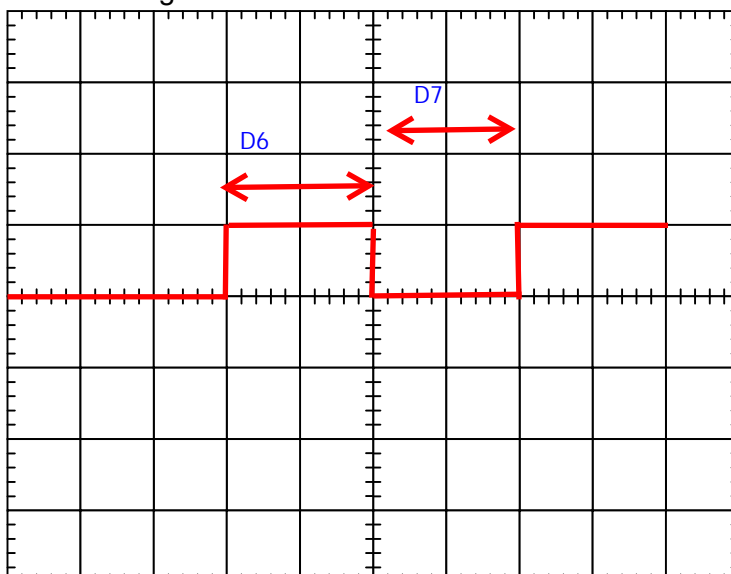
$$0.0000521 * 9 = 0.0004689$$

environ 50 us

2) Etude pratique, en utilisant le programme appelé 'TransmissionB'. (caractère 'B')

a) Régler l'oscilloscope pour visualiser le plus précisément la partie du signal correspondant à D6, D7 et STOP. (Valider les réglages par le professeur)

b) Relever ce signal :



Calibre tension CH1 :

5v

Calibre tension CH2 :

Calibre base de temps :

25us

c) Représenter sur le relevé, le temps à l'état 1 de D6 et mesurez ce temps

Le calibre de temps est sur 25us, d6 prends 2 bases de temps donc 50us

d) A partir de votre mesure, retrouver la valeur de la vitesse de transmission de la liaison série

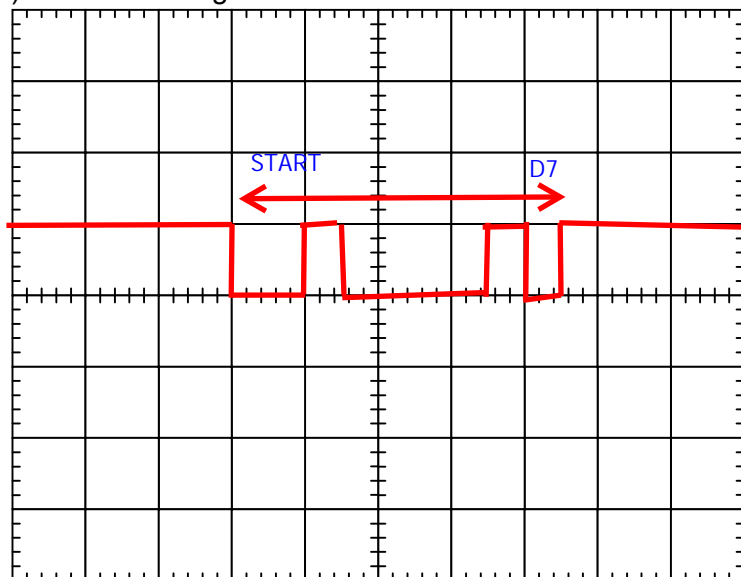
En bauds (bit/seconde), calcul :

$$1/0.00005 = 20000 \quad 1/19200 = 0.0000521$$

3) Etude pratique, en utilisant le programme appelé 'TransmissionB'. (caractère 'B')

a) Régler l'oscilloscope pour visualiser le plus précisément la totalité du signal correspondant à l'envoi du caractère (Valider les réglages par le professeur)

b) Relever ce signal :



Calibre tension CH1 :
5V

Calibre tension CH2 :

Calibre base de temps :
100 us

c) Représenter sur le relevé, le temps de transmission, mesurable entre START et D7 et mesurez ce temps

450us

d) A partir de votre mesure, retrouver la valeur de la vitesse de transmission de la liaison série

En bauds (bit/seconde), calcul :

$$9/0.000450 = 20000.0000$$

4) Conclusion : (compléter) Par la mesure, il est aisé de retrouver la vitesse de transmission, deux méthodes sont possibles :

on mesure le temps de transmission et le nombre de bits transmis puis on les divise (bits/seconde), ce qui nous permet de le retrouver en bauds

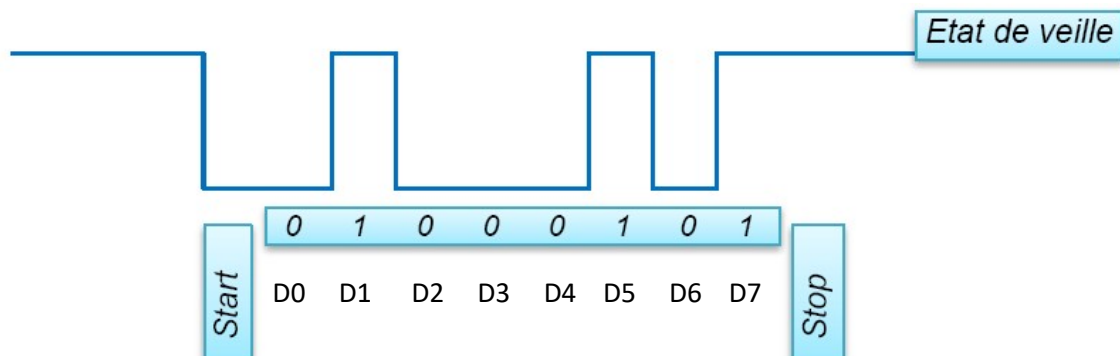
ou
on prend un bit et on le divise par son temps de transmission (bit/seconde), et on retrouve la vitesse de transmission en bauds

Annexe 1: protocole de transmission : START DONNEE STOP

La figure montre l'information en TX lors de l'envoi de l'octet de valeur hexadécimale : A2 = 1010 0010b

D7 D6 D5 D4 D3 D2 D1 D0
1 0 1 0 0 0 1 0

- Le bit de START, niveau bas, est transmis
 - La donnée, 8 bits, Le bit D0 de poids faible de l'octet transmis est envoyé en premier.
 - Le bit de STOP, niveau haut, est transmis
- Puis c'est l'état de veille : niveau haut pendant une durée indéterminée



Annexe 2: Table code ASCII

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL