

SYSTEME DE GESTION DES ENERGIES

EWTS

EMBEDDED WIRELESS TELEMETRY SYSTEM







Copyright **TECHNEXT**[®] 2012



Nom :
Prénom :
Classe :

 **Problématique:**
Comment le ClipFlow transmet-il physiquement les données au serveur ?

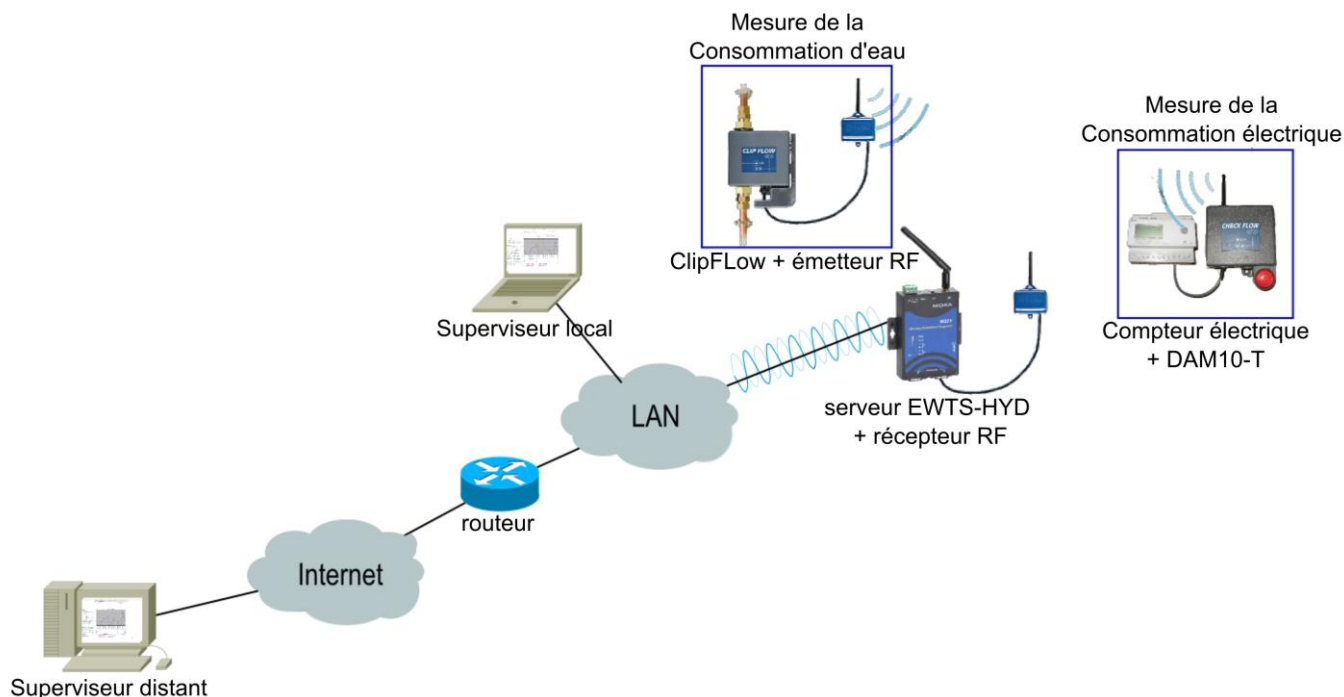
Activités du TP:

-  **1 Identification des constituants de la chaîne d'information**
-  **2 Analyse de la transmission d'une trame du ClipFlow**
-  **3 Analyse de la transmission d'un caractère**
-  **4 La communication sans fil**
-  **5 Adéquation avec le cahier des charges**
-  **6 Synthèse sur la liaison série RS232**

Activité 1 : Identification des constituants de la chaîne d'information

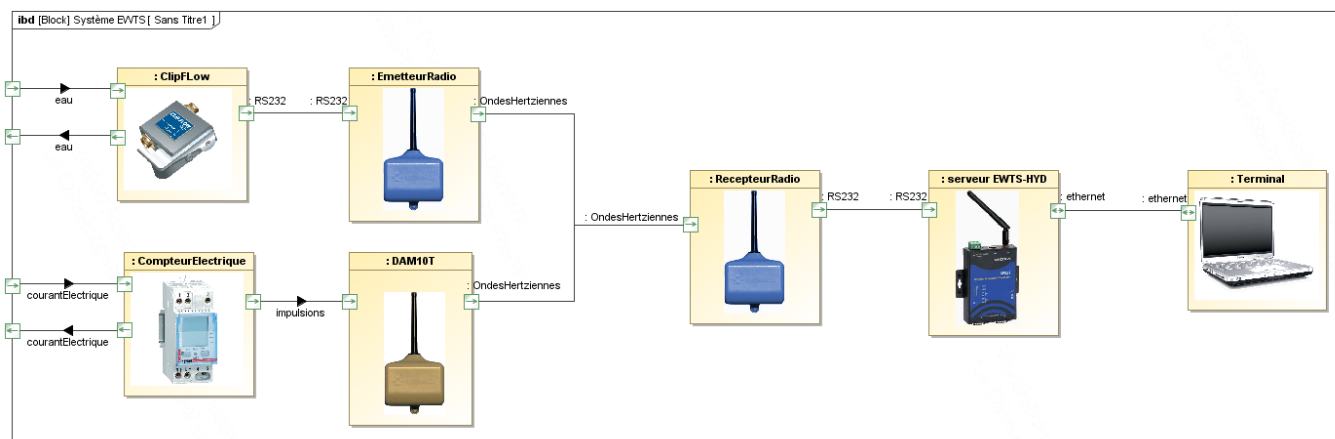
L'objectif de ce TP est d'analyser la communication entre les capteurs et le serveur du système EWTS d'un point de vue physique, c'est-à-dire la façon dont les bits composant la trame sont transmis physiquement sur le support.

Q1. Sur le synoptique suivant, entourer les éléments concernés par cette analyse.

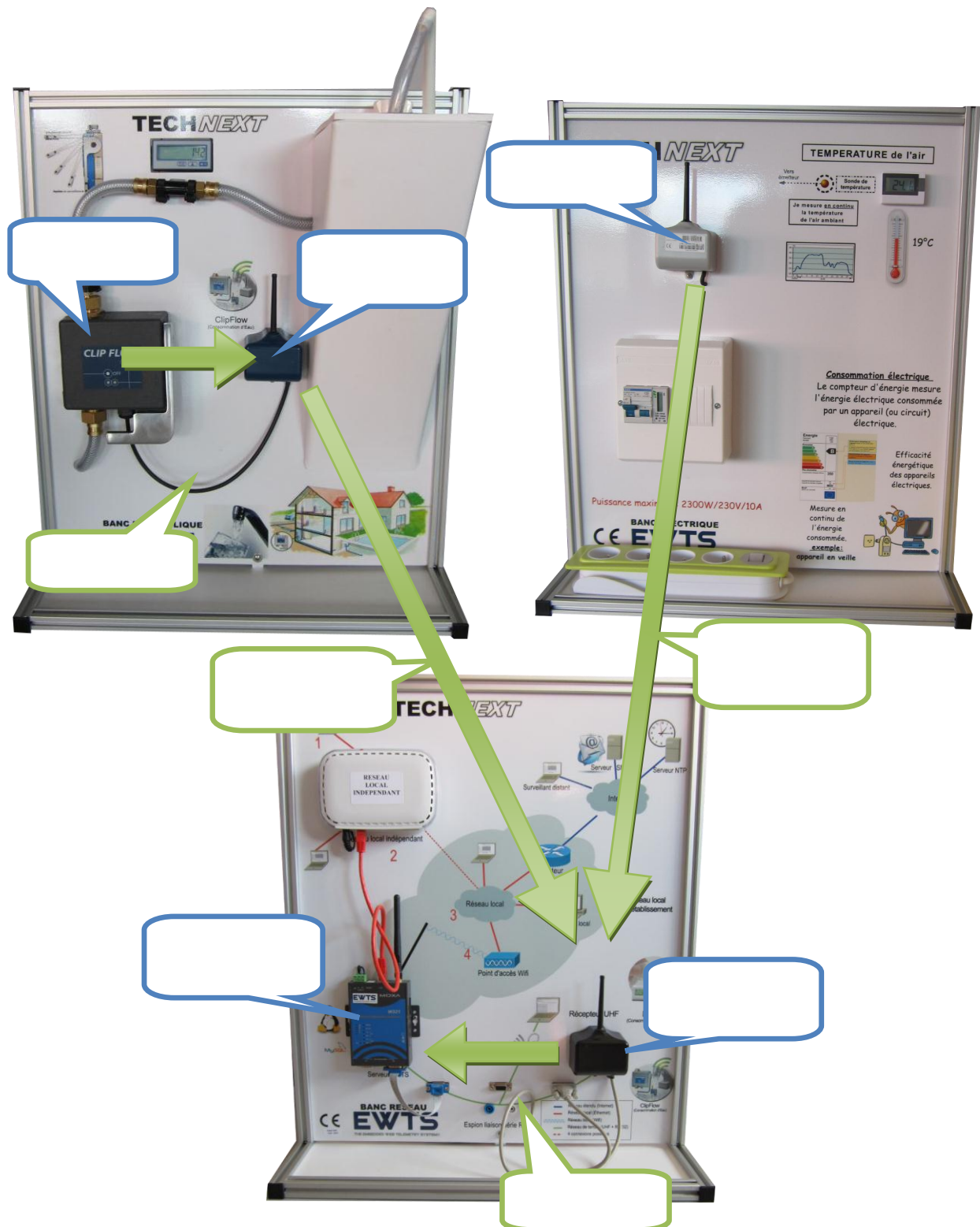


Q2. A quelle couche du modèle OSI cette analyse fait-elle référence ?

Le diagramme interne de bloc SysML montre les relations entre chacun des blocs composant le système. Les ports de chaque bloc permettent de préciser la nature et le sens de circulation des flux (matière, énergie ou information) qui transitent d'un bloc à l'autre.



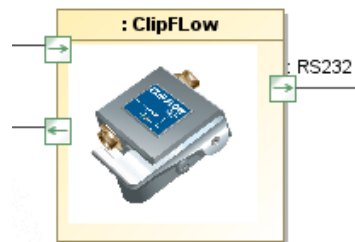
Q3. A partir du diagramme précédent et en observant les bancs didactiques, compléter la légende des photos suivantes en entourant et en nommant les liaisons entre la DAM10T, le ClipFlow et le serveur.



Remarque :

Dans le système réel, le cordon gris qui sort du récepteur radio est relié directement au port P2 du serveur EWTS-HYD. Sur le banc réseau, Une prise **espion** composée d'un connecteur DB9 (connexion RS232 vers un PC) et de deux fiches bananes (connexion vers un oscilloscope) permettent de capturer les échanges entre les capteurs et le serveur. Celle-ci est « intercalée » entre le récepteur et le serveur.

En comparant le modèle SysML (diagramme de bloc interne) aux éléments réels présents sur les maquettes, répondre aux questions suivantes :



Q4. A quel type de connecteur correspond le port RS232 du bloc ClipFlow ?

.

Q5. Combien de broches possède-t-il ?

.

Remarque :

Ce connecteur est également utilisé pour :

- délivrer des impulsions (une impulsion à chaque litre consommé),
- commander le déclenchement
- déclencher une alarme
- connaître la position du levier du ClipFlow (ouvert ou fermé)
- alimenter le ClipFlow

Q6. D'après le document « NoticeClipFlow.pdf » page 14, donner le numéro et le rôle des 3 broches utilisées pour la communication série.

.

Remarque :

Ces trois broches sont utilisées dans le cadre d'une communication bidirectionnelle, ce qui est le cas lorsqu'on connecte le ClipFlow à un PC à l'aide du cordon USB-RS232.



Q7. Quelles sont les 2 seules broches utilisées lorsque le ClipFlow est connecté à l'émetteur radio.

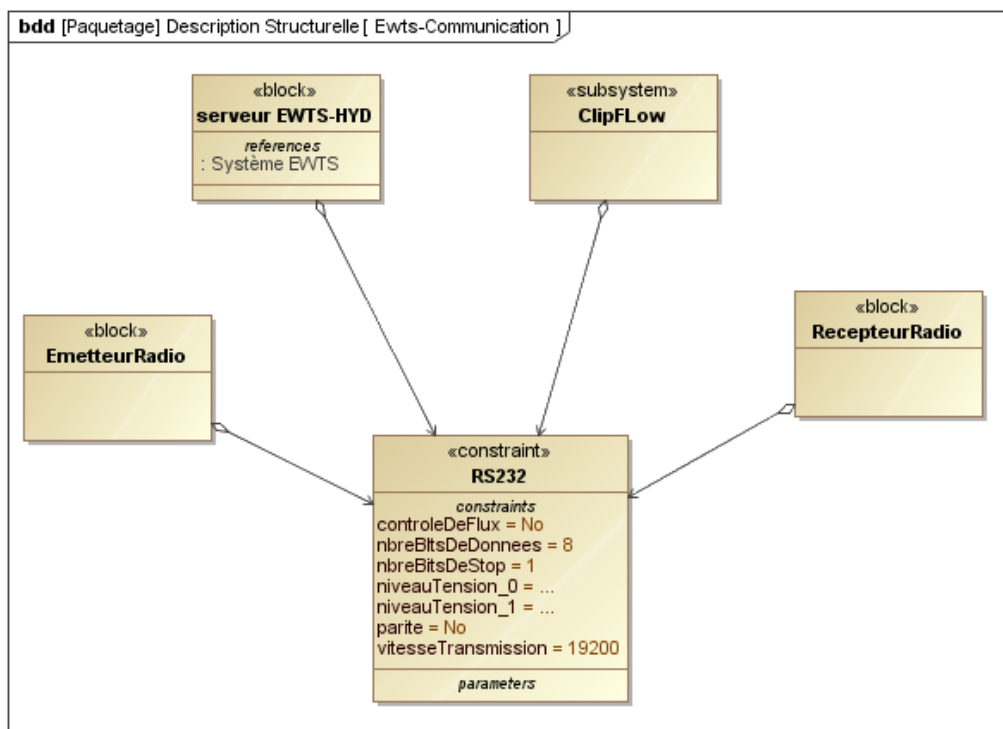
.

Q8. A quel type de connecteur correspond le port RS232 du bloc serveur EWTS-HYD ?

.

Activité 2 : Analyse de la transmission d'une trame du ClipFlow

🖱️ *Quels sont les paramètres de la liaison série ?*



Les données sur la liaison série RS232 sont transmises caractère par caractère, bit à bit. La transmission d'un caractère est constituée au minimum de :

- 1 bit de START à la valeur logique 0
- 7 ou 8 bits de données **en commençant par le bit de poids faible**
- 1 bit de STOP à la valeur logique 1

A la lecture du document « *caracLiaisonSerie.pdf* » et du diagramme de définition de blocs précédent, répondre aux questions suivantes :

Q9. Sur combien de bits les données sont-elles codées ?

.

Q10. Combien de bits de STOP sont-ils utilisés pour les transmissions de chaque caractère au sein du système EWTS ?

.

Q11. Quelle est la vitesse de transmission choisie pour la communication série pour le système EWTS ? Celle-ci s'exprime en bauds. Dans le cas du système EWTS un baud est équivalent à 1bits/s. Ce n'est pas toujours le cas.

.

Remarque :

La communication RS232 permet de mettre en œuvre un mécanisme de détection d'erreur en rajoutant un bit de parité avant le bit de STOP qui marque la fin de l'envoi du caractère. Il existe deux types de parité :

- parité paire : on compte le nombre de bit à 1 dans l'octet à transmettre et on rajoute un bit à 1 ou à 0 afin que le nombre total de bits à 1 soit pair
- parité impaire : même chose, mais cette fois on donne la valeur au bit de parité pour que le nombre de bits à 1 soit impair.

Dans les deux cas, le récepteur vérifie lors de la réception que le nombre de bits à 1, bit de parité compris, est bien pair (cas d'une parité paire) ou impair (cas d'une parité impaire). Si ce n'est pas le cas, c'est que au moins un bit a changé de valeur durant la transmission, synonyme d'une erreur.

Ex: La transmission de l'octet $(4A)_h = (01001010)_b$ avec une parité paire se fera ainsi :

Start	Données	Parité	Stop
0	01010010	1	1

car cet octet contient 3 bits à 1. Le bit de parité à 1 rend le nombre total de 1 pair (le bit de stop n'est pas compté dans le calcul)

Q12. Le bit de parité est-il mis en œuvre dans notre cas ?**Remarque :**

La liaison série RS232 propose un mécanisme de contrôle de flux qui permet à un récepteur d'interrompre l'émetteur dans le cas où il ne peut pas traiter les données reçues suffisamment vite et de demander la poursuite des échanges dès qu'il est à nouveau en mesure de les traiter. Il existe deux types de contrôle de flux :

- Contrôle de flux matériel : ce contrôle s'appuie sur deux broches supplémentaires RTS (Request to Send) et CTS (Clear to send).
- Contrôle de flux logiciel : le récepteur envoie un caractère spécial nommé XOFF pour interrompre l'émetteur et lui envoie le caractère XON pour lui signifier qu'il peut reprendre la transmission

Q13. Ces deux types de contrôle de flux pourraient-ils être mis en œuvre dans le cas où le ClipFlow est relié à son émetteur? Justifier.

Contrôle de flux logiciel :

.

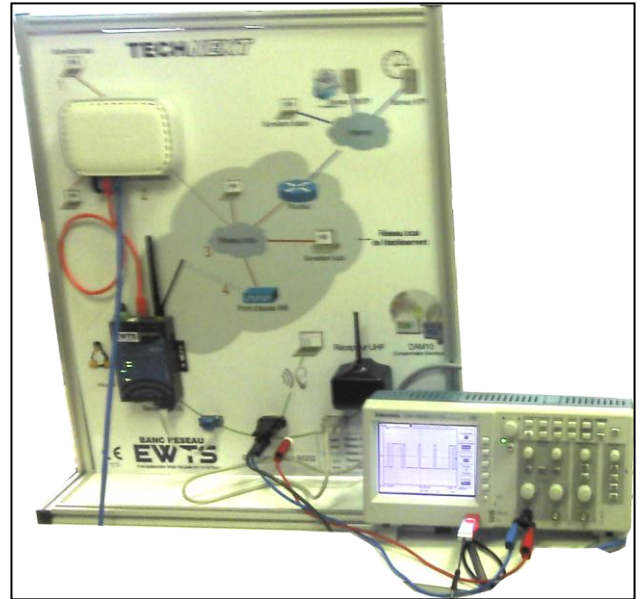
Contrôle de flux matériel :

.

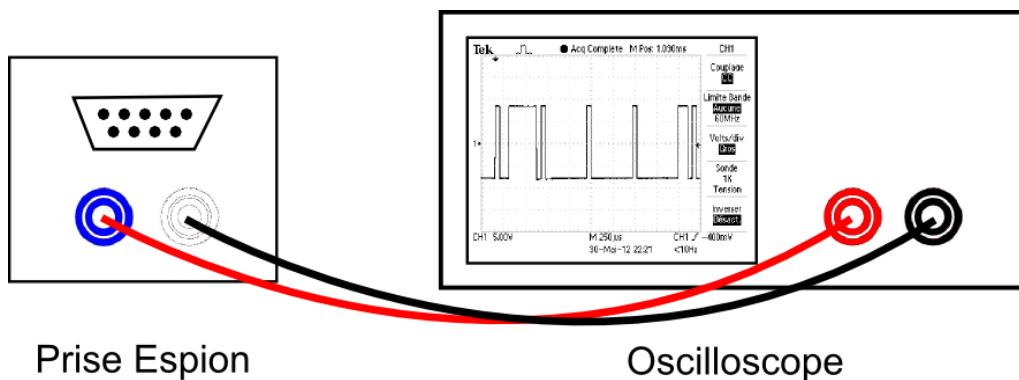
Acquisition de la trame

Remarque :

Le banc réseau est pourvu d'une prise espion qui va nous permettre de capturer les échanges de données entre le ClipFlow et le serveur EWTS.



Regardons tout d'abord l'allure générale d'une trame à l'oscilloscope. Pour cela, relier l'oscilloscope aux fiches bananes de la prise espion du banc réseau comme présenté sur la figure ci-dessous :

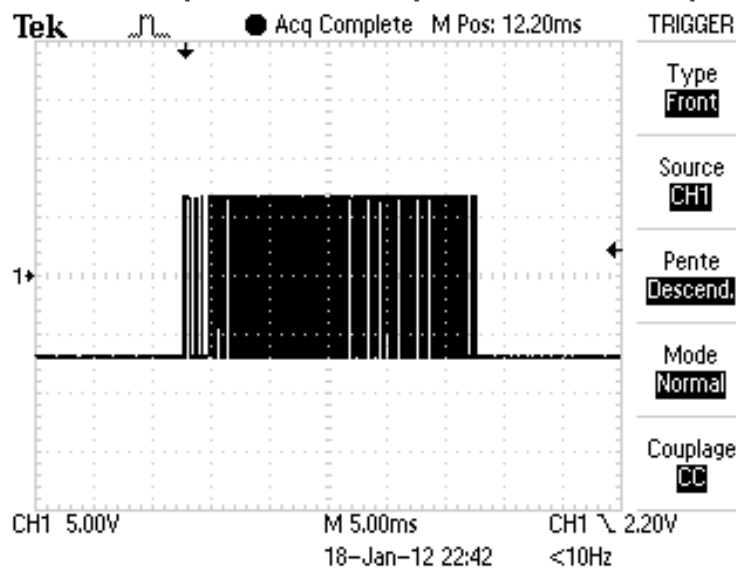


Q14. Régler l'oscilloscope comme suit :

- Régler l'oscilloscope pour qu'il fasse une unique acquisition (one shot) sur un déclenchement (mode Trigger) par franchissement d'un seuil de tension légèrement positif.
- Placer le seuil aux alentours de 2 volts.
- Configurer le calibre et la période de balayage de la voie sur laquelle est reliée la prise espion avec les valeurs ci-dessous :

Calibre	Période de balayage
5V/div	5ms/div

Q15. A l'aide du banc hydraulique, faire couler l'eau jusqu'à ce que le ClipFlow transmette une trame et que l'oscilloscope déclenche l'acquisition.



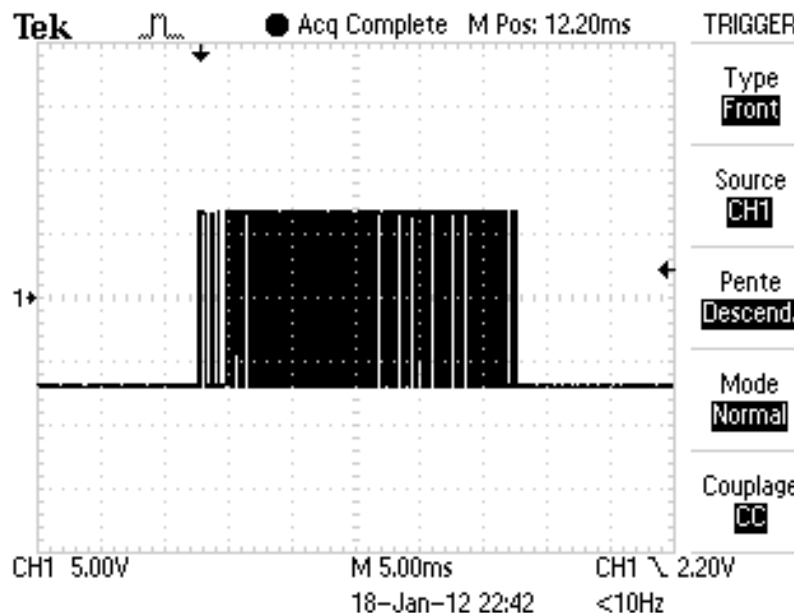
Q16. De quel type de trame s'agit-il, probablement ?

.

 **Mesure de la durée d'une trame**

Q17. A l'aide des curseurs de l'oscilloscope, mesurer la durée de cette trame.

.



 **Vérification de la durée de la trame par le calcul**

D'après le document *caracLiaisonSerie.pdf* à la page 4,

Q18. Combien de caractères compte une trame de type « suivi de fonctionnement » transmise par le clipflow, sachant que celui-ci transmet systématiquement deux caractères de début de trame (LF) ?

Remarque :

Rappelons que la description du protocole précise qu'il en faut au moins un.

Q19. Combien de bits sont-ils effectivement transmis via la RS232 pour chaque caractère ?

Q20. Déduire des deux questions précédentes, le nombre de bits transmis pour chaque trame de type « suivi de fonctionnement.

Q21. Rappeler la vitesse de transmission choisie pour les échanges entre le clipflow et le serveur via la RS232.

Q22. Déduire des questions précédentes la durée d'une trame de type « suivi de fonctionnement » sur la liaison RS232.

Q23. Comparer la valeur calculée et la valeur mesurée.

Activité 3 : Analyse de la transmission d'un caractère **Analyse du code de réveil**

Q24. D'après la documentation sur le protocole de communication développé par la société Hydrelis (*caracLiaisonSerie.pdf*), pourquoi a-t-on besoin d'envoyer un code de réveil ?

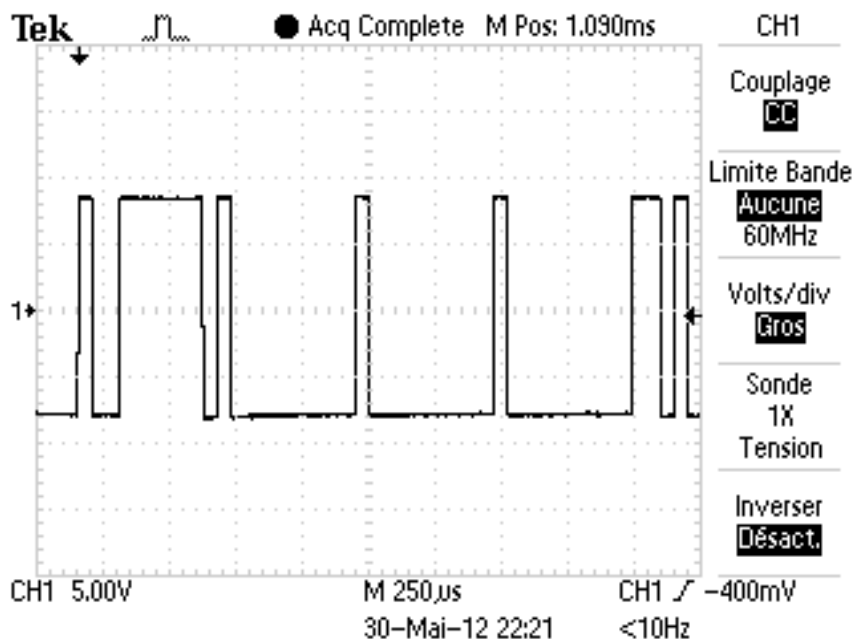
Q25. D'après la documentation, quels sont les 4 octets qui composent ce code de réveil ?

Q26. Quel est donc, en hexadécimal, le premier octet que le serveur reçoit du ClipFlow sur sa liaison série ?

Q27. Réaliser une nouvelle capture de trame à l'oscilloscope en faisant couler l'eau sur le banc hydraulique. Pour cette capture, régler l'oscilloscope de façon à visualiser intégralement le code de réveil. Utiliser les paramètres suivants :

Calibre	Période de balayage
5V/div	250us/div

Q28. Estimer, à l'oscilloscope et sur le relevé suivant, le début et la fin de chacun des 4 octets du code de réveil.



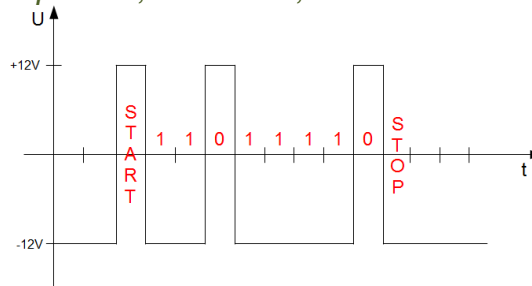
Q29. Relancer une capture afin d'observer uniquement le premier caractère du code de réveil en utilisant les paramètres suivants :

Calibre	Période de balayage
5V/div	100us/div

Remarque :

Une transmission série RS232 (ou EIA-232) s'appuie sur un codage nommé NRZ (Non Return to Zero). La valeur des bits est représentée par un niveau de tension positif ou un niveau de tension négatif. De plus, ces niveaux électriques sont inversés. Ainsi la valeur positive définira un 0 logique, la valeur négative correspondra à un 1 logique. La norme définit des niveaux de tension compris entre +3 à +25 pour le 0 logique et -3 à -25 pour le 1 logique

Par exemple, l'octet 0x7B qui vaut, en binaire, 01111011 sera représenté comme suit :

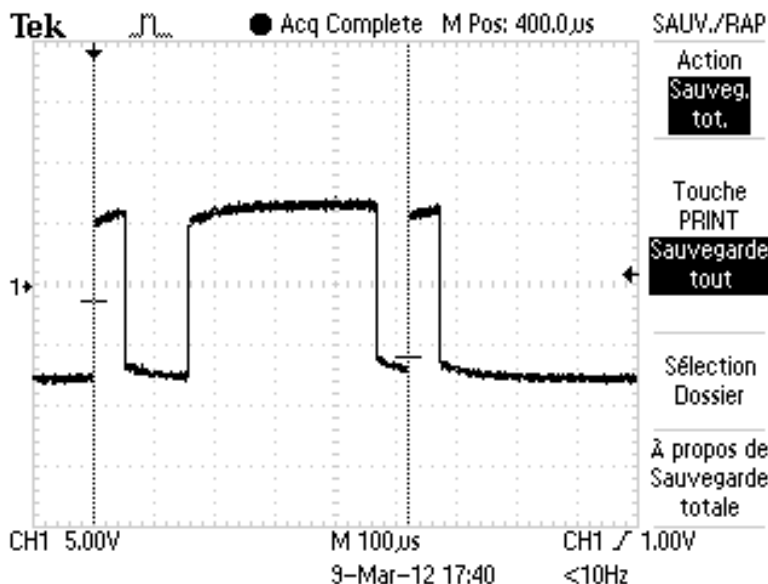


Q30. Relever les niveaux de tension en précisant les correspondances avec la valeur 0 ou 1 et compléter le diagramme de définition de bloc du début de l'activité 2.

bit	0	1
Niveau de tension (en V)		

Q31. Ces niveaux respectent-ils les préconisations de la norme RS-232 ?

Q32. A l'oscilloscope et sur le relevé suivant, identifier le bit de START, la valeur de chacun des 8 bits de données et le bit de STOP.



Q33. Rappelons que c'est le bit de poids faible qui est transmis en premier. Donner la valeur en binaire de ce premier caractère reçu.

.

Q34. Quelle est sa valeur en hexadécimal ? Comparer avec la valeur attendue.

.

Q35. A l'aide des curseurs de l'oscilloscope, mesurer la durée d'un bit (effectuer la mesure sur le bit de START).

.

Q36. A partir de la vitesse de transmission, retrouver par le calcul la durée théorique d'un bit lors de la transmission.

.

Activité 4 : La communication sans fil



Le fait de relier les capteurs au serveur par une liaison sans fil permet de s'affranchir d'une infrastructure câblée coûteuse et difficile à installer.

Le CLipFlow peut être relié à deux types d'émetteurs :

- alimenté avec une de 4km
- autonome avec une portée de 400m

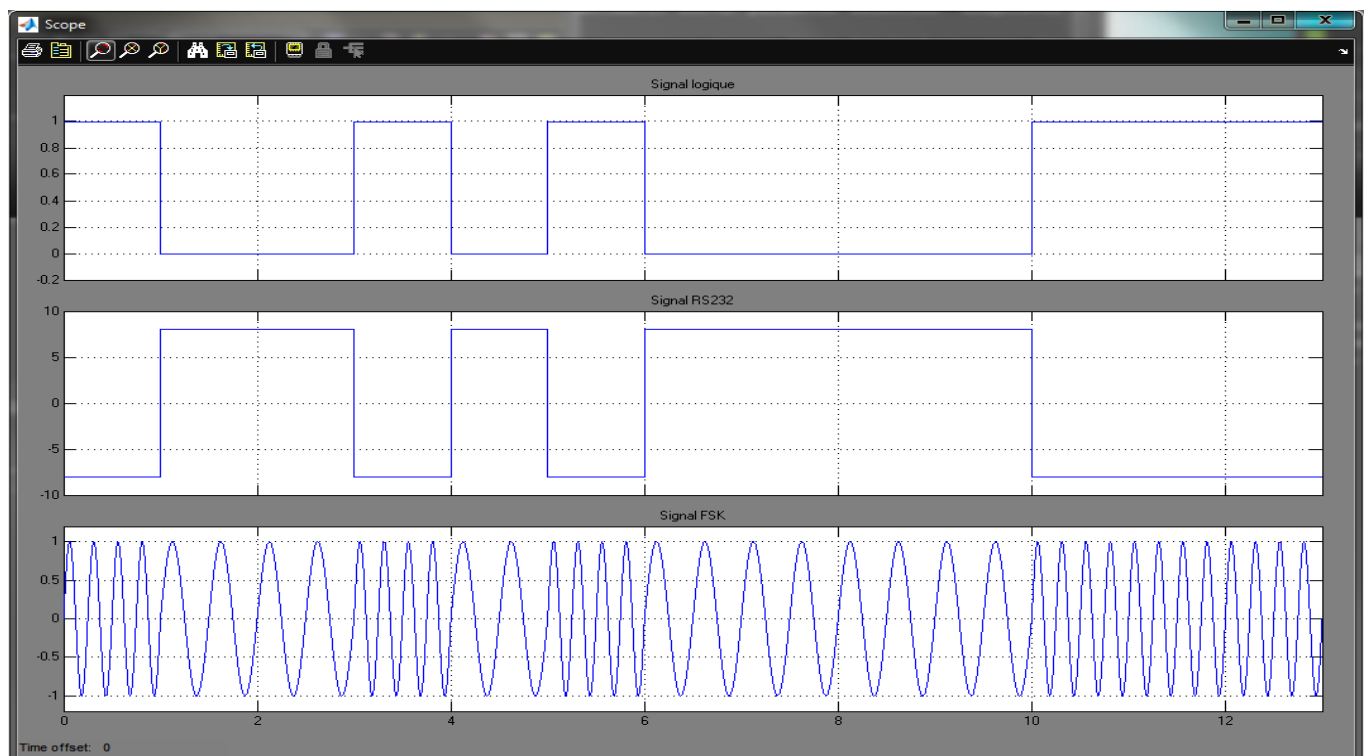
L'émetteur transmet le signal numérique par ondes hertziennes en utilisant une modulation FSK (Frequency Shift Keying).

Le fichier Matlab/Simulink nommé FSK.mdl simule la transmission d'un caractère via la liaison série RS232 puis modulé en FSK.

Remarque :

Les périodes des différents signaux ne sont pas réalistes. Le but de ce modèle est seulement de présenter le principe de la modulation FSK.

Q37. Ouvrir le fichier FSK.mdl et lancer la simulation



Q38. Repérer sur la courbe du signal logique, le bit de START, les huit bits de données ainsi que leur valeur et le bit de STOP.

Q39. De quel caractère s'agit-il, d'après la table des codes ASCII fournie en annexe ?

Q40. En analysant la courbe « Signal FSK », expliquer le principe de la modulation FSK. On observera notamment comment les 0 et les 1 sont codés avec ce type de modulation.

Q41. Quel est alors le rôle du récepteur connecté au serveur ?

Le signal utile est « transporté » par un signal de fréquence 868MHz. Ce signal est appelé « porteuse ». Les deux fréquences utilisées pour coder le 0 et le 1 se situent autour de cette porteuse.

Niveau logique	Fréquence (en Hz)
0	$868.10^6 + f_d$
1	$868.10^6 - f_d$

Remarque :

En fait le nom exact de la modulation utilisée par les émetteurs est GFSK. Elle consiste à faire passer le signal issu de la modulation FSK dans un filtre (qualifié de Gaussien) afin d'adoucir les transitions lorsque le signal émis passe d'une fréquence à une autre. Cette technique permet de réduire l'étalement du spectre du signal afin de ne pas empiéter sur la bande passante d'un canal de transmission voisin.

Activité 5 : Adéquation avec le cahier des charges

Le cahier des charges du système EWTS stipule que le serveur peut gérer jusqu'à 200 capteurs.

Q42. En supposant que chacun de ces capteurs émette l'un après l'autre une trame de « suivi de fonctionnement », quel serait le temps que durerait l'ensemble de ces transmissions ?

.

Q43. Les clipflows transmettent une trame toutes les 2 minutes ou une trame tous les litres consommés. En se plaçant dans le cas le plus défavorable où tous les clipflows sont traversés par un débit maxi de 2500L/h, calculer le nombre de trames qui seraient envoyées par seconde par l'ensemble de ces clipflows.

.

Q44. Quelle serait la durée de l'envoi de l'ensemble de ces trames en supposant qu'elles soient transmises l'une après l'autre ?

.

Q45. Conclure sur la capacité du serveur à gérer autant de trames et donc autant de capteurs dans ces conditions extrêmes.

.

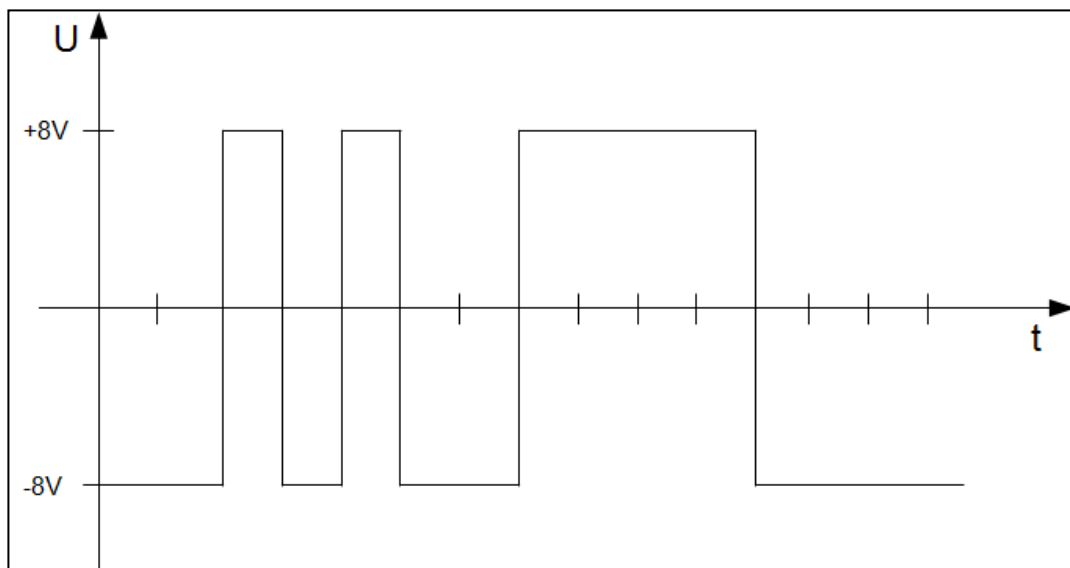
Q46. Combien de ClipFlows seulement le serveur EWTS pourrait en théorie gérer dans ces conditions ?

.

Activité 6 : Synthèse sur la liaison série RS232**Q47. Compléter le résumé suivant :**

Les caractéristiques de la liaison série RS232 appartiennent à la couche du modèle OSI. Une transmission sur une liaison série RS232 s'effectue par . Chaque caractère est codé sur ou bits. Chacun de ces caractères est transmis à en commençant par le bit de poids . Chaque caractère transmis est précédé d'un bit de dont le niveau logique vaut et suivi d'un ou plusieurs bits de dont le niveau logique vaut . La liaison série adopte un codage (Non Return to Zero) pour lequel un niveau de tension positif correspond à un logique et un niveau de tension négatif à un logique.

Q48. Le chronogramme suivant représente un extrait de la capture d'une trame émise par le ClipFlow. Il s'agit d'un des caractères de la trame. Repérer sur celui-ci, le bit de START, les 8bits de données et leur valeur logique (1 ou 0) ainsi que le bit de STOP.

**Q49. En déduire la valeur en binaire de l'octet qui a été transmis**

Q50. Quelle est la valeur de cet octet en hexadécimal ?

Q51. En s'appuyant sur la table des codes ASCII fournie en annexe, retrouver à quel caractère ASCII correspond cet octet.

Q52. A quel endroit dans la trame retrouve-t-on ce caractère et quel est son rôle ?

Annexes

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

Source: www.LookupTables.com