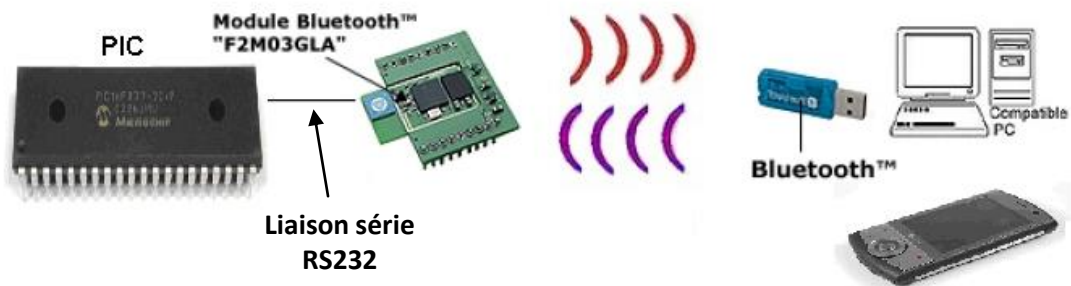


## 1. présentation

Grace à l'application Hyper Terminal, vous allez envoyer **à distance et sans fil** des ordres depuis le clavier de votre PC ou téléphone portable, lesquels seront transmis au module Bluetooth F2M03GLA via la communication Bluetooth de l'ordinateur ou du téléphone.

Il est donc nécessaire de faire dialoguer le microcontrôleur PIC et module Bluetooth



Il faut donc les relier pour qu'ils échangent des signaux électriques et qu'ils se comprennent (langage commun = **protocole**)

Dans notre application on utilisera une liaison série au protocole RS232.

## 2. La liaison RS232

### Cette liaison, c'est quoi ?

La liaison RS-232 est issue de la norme du même nom qui permet l'envoi de données via une chaîne de niveaux logiques envoyés en série (d'où le nom du port du PC). Elle permet de faire dialoguer deux systèmes (et seulement deux) entre eux. Les données sont envoyées par trames de 8 bits, soit autant de niveaux logiques

Cette liaison est de type **asynchrone**, c'est-à-dire qu'elle n'envoie pas de signal d'horloge pour synchroniser les deux intervenants de la liaison : il est donc nécessaire que ces derniers soient configurés de la même manière (vitesse de transmission, nombre de bits par trame, etc.). La vitesse de transmission s'exprime en **bauds** (**bds** = bits par seconde) : les valeurs les plus courantes sont 2400, 4800 et 9600 bauds.

### Comment ça fonctionne ?

Au repos, la ligne de transmission est au niveau logique 1. Lorsque l'un des systèmes veut commencer à communiquer, il prévient le système à l'autre bout de la liaison par une mise de la ligne au niveau 0 : c'est le bit de **Start**. Viennent ensuite les bits de données au nombre de 8 : ils sont soit au niveau 1, soit au niveau 0, en fonction des données. Un dernier bit peut être ajouté, il s'agit du bit de parité (ou d'erreur). Enfin, après ce bit de parité viennent un ou deux bits de **Stop**, qui signalent au receveur que la trame est terminée.



Une liaison, par définition, c'est un lien entre deux éléments ou plus. Ce lien peut-être visible (un câble) ou invisible (les ondes radio). Notre liaison RS-232 a donc besoin d'un lien physique pour fonctionner.

Les prises utilisées sur les ordinateurs pour le port RS-232 (port **Série**) sont de type **DB9**.



Pour une liaison RS-232 simple, seules trois broches de la prise sont utilisées ; dans la majorité des cas, les systèmes en communication ne nécessitent guère plus que ce type de liaison.

### 3. Le code ascii

L'ordinateur est une grosse machine à calculer : tout ce qu'il sait faire, c'est effectuer des calculs sur des **nombre**s. Il est incapable de comprendre le texte.

Il faut donc faire un choix : par quel nombre représenter la lettre 'A' ? Et pour les signes de ponctuation, quels nombres utiliser ?



Il existe différentes conventions (ou codes). L'un des plus connus est le code **ASCII** (American Standard Code for Information Interchange). C'est un standard américain, mais c'est l'un des plus utilisés, en particulier sur la plupart des ordinateurs.

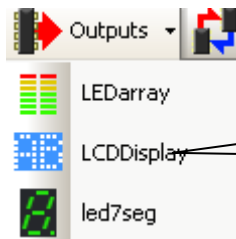
Le code ASCII définit précisément la correspondance entre symboles et nombres jusqu'au nombre 127.

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

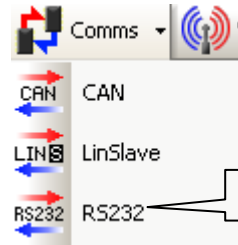
Source: [www.LookupTables.com](http://www.LookupTables.com)

#### 4. Prise en main de la liaison RS232

- Lancez  Flowcode.exe
- Puis  Create a new FlowCode flowchart...
- Choisir un microcontrôleur (Puce) : 16F877A
- Choisir

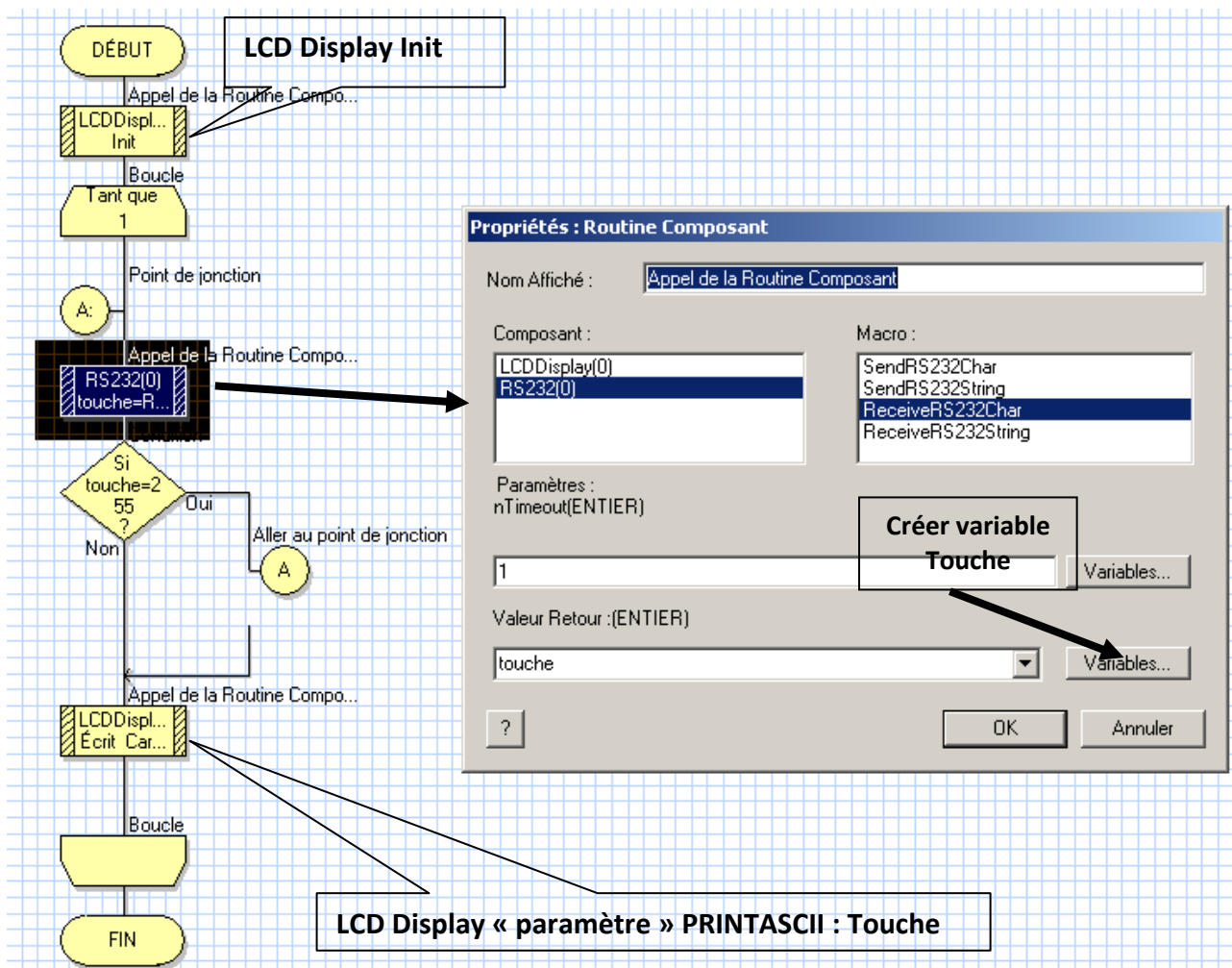



Module LCD

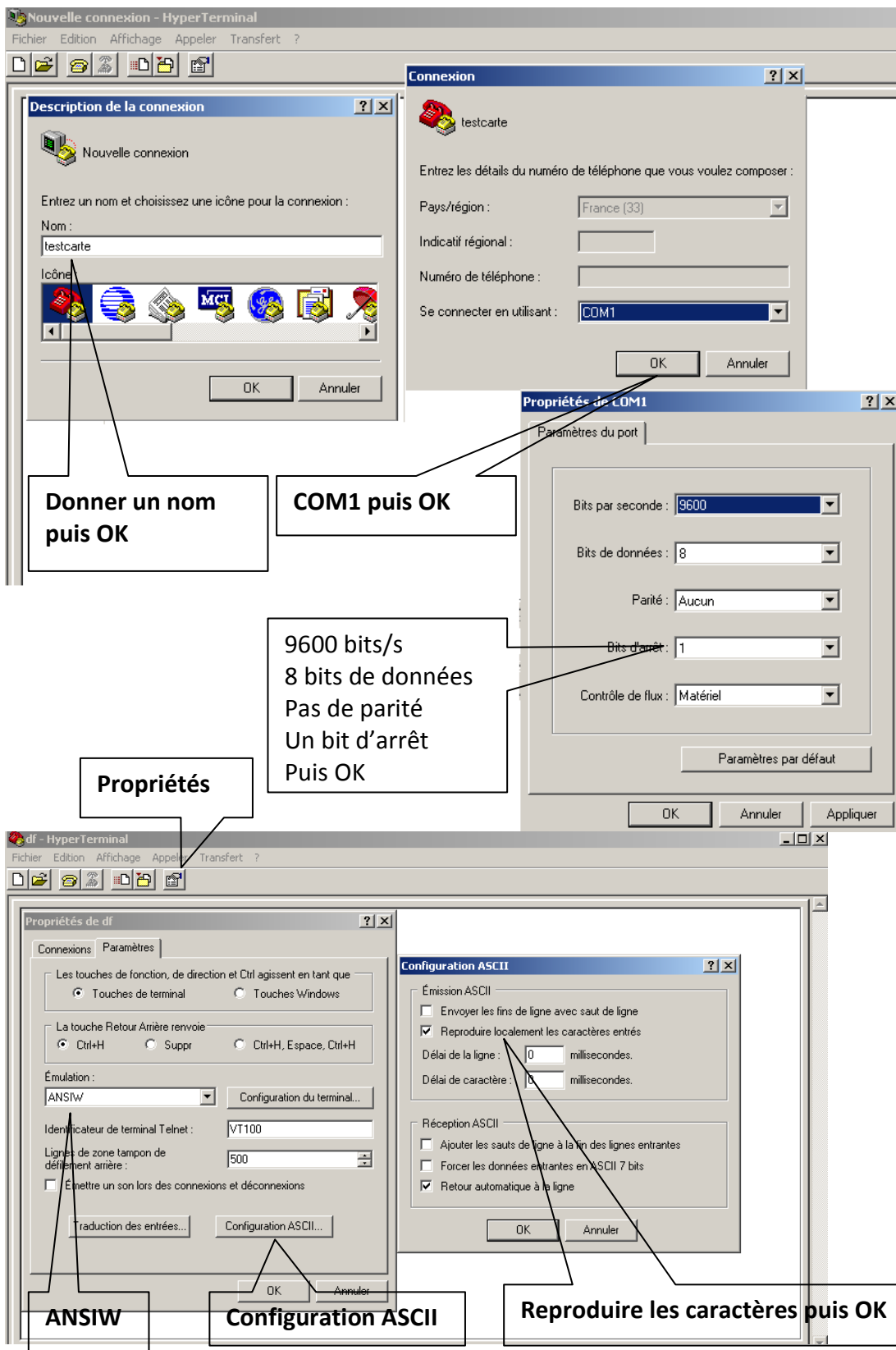



Module RS232

- Réaliser le programme suivant.

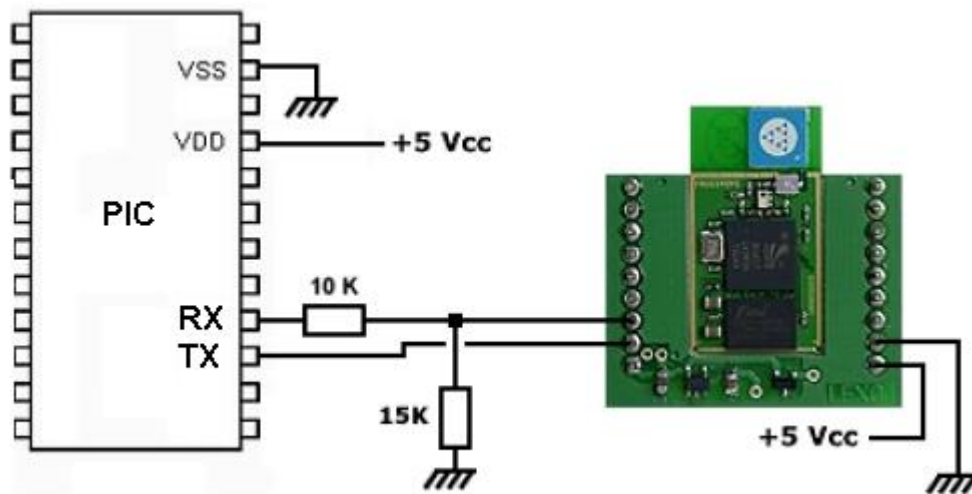


- Lancer le programme , puis connecter le PC au Kit de développement Flowcode via la carte RS232 et le cordon série.
- Ouvrir Hyper terminal « Démarrer/Tous les programmes/Accessoires/Communications/Hyper Terminal




- Se connecter 
- Dans le logiciel Hyper terminal, appuyer sur les touches du clavier du PC que se passe-t-il ?


## 5. Pilotage des servomoteurs via la liaison RS232

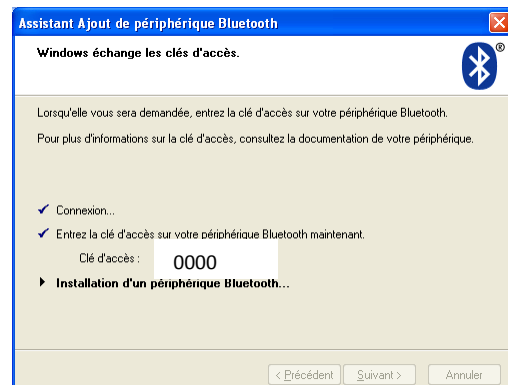
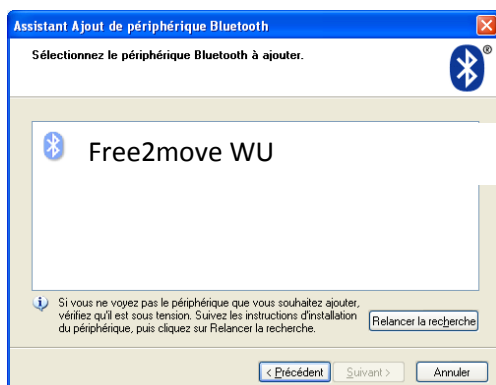


Les touches 4 et 6 permettront au char d'aller à GAUCHE et à DROITE, les touches 8 et 2 permettront de BORDER et CHOQUER.

- Par quoi doit-on remplacer les variables a0, a1, b0 et b1 de notre programme pour qu'il puisse fonctionner ?
- Quel est le code ASCII des touches 2, 4, 6 et 8 ?
- Ouvrir le programme « char\_a\_voile.fcf ».
- Modifier le programme « char\_a\_voile.fcf », afin que :  
L'appuie sur 4 et 6 autorisent la rotation à GAUCHE et DROITE de la roue avant  
L'appuie sur 2 et 8 autorisent le BORDER et CHOQUER.
- Exécuter votre programme . Faites le valider par votre enseignant.

## 6. Pilotage via le module Bluetooth

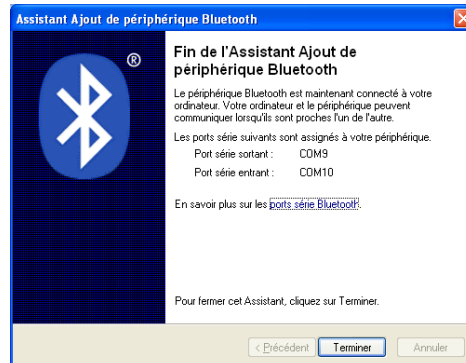
- Brancher sur un port USB du PC l'adaptateur Bluetooth. Attendre que Windows indique que la clé soit correctement installée et prête à l'emploi. Faire ensuite un clic droit sur l'icône Bluetooth à côté de l'heure , puis « Ajouter un périphérique Bluetooth ». Cocher « Mon périphérique est préparé et prêt à être détecté ». Le logiciel scan alors les différents nœuds disponibles. Si le module Bluetooth est correctement branché et correctement initialisé, il apparaît dans la liste. Les modules s'appellent « **Free2move WU** » suivi de leur adresse. Vous pouvez lire l'adresse de votre module sur l'étiquette collée au dos de l'Eblock.



Cliquer sur l'icône de votre module, puis faire suivant.

Vous devez ensuite « Utiliser la clé de sécurité se trouvant dans la documentation » et remplir par « 0000 ».

Cliquer sur suivant. La connexion s'établit et le logiciel vous indique le port série virtuel du module USB. C'est le numéro du port sortant qui nous intéresse (COM9 dans cet exemple).



- Lancer maintenant HyperTerminal (voir partie 4) et utiliser le port (COM9 dans cet exemple) pour envoyer des données.