

Dossier ressources : Stylophone

1. Découverte du produit et de la problématique technique

Avec plus de trois millions d'exemplaires écoulés, principalement comme jouet, le stylophone a marqué les années 1960 et 1970 par son succès populaire. Pourtant, malgré son adoption par des artistes emblématiques comme The Velvet Underground, David Bowie ou encore Queen, son utilisation musicale reste marginale, en raison de ses limitations techniques intrinsèques.

L'instrument se présente sous la forme d'un petit clavier métallique de 20 notes, sur lequel l'utilisateur appuie avec un stylo relié par un fil électrique. Ce contact ferme le circuit et génère la note correspondante. Cependant, le son produit est rudimentaire, proche d'un signal carré basique, et manque cruellement de richesse harmonique.

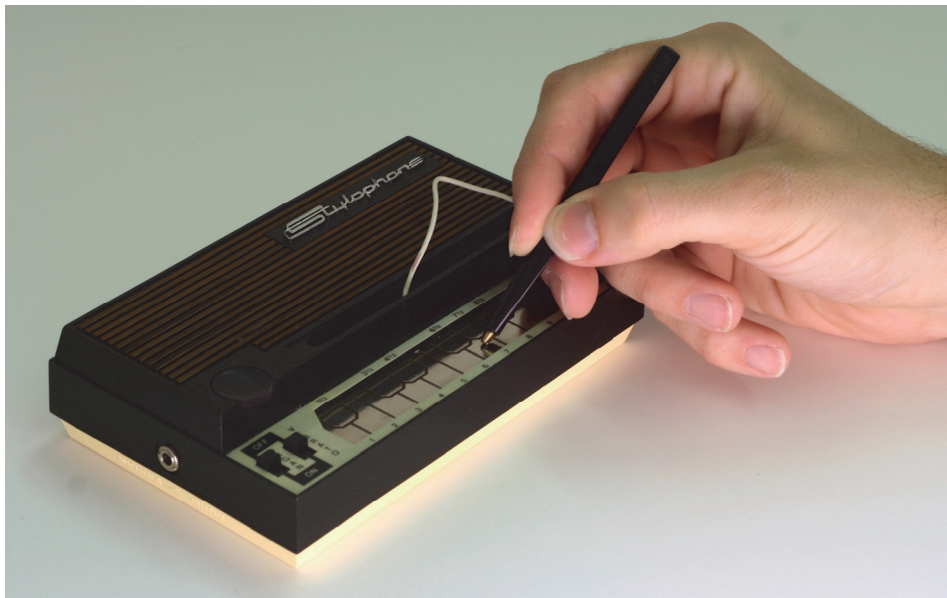


Figure 1 : Utilisation du stylophone

La principale problématique réside dans l'intégration d'une sortie audio exploitable : le signal natif du stylophone, faible et non amplifié, nécessite une adaptation complexe pour être connecté à des systèmes de sonorisation ou d'enregistrement professionnels, limitant ainsi son usage en contexte musical exigeant.

Le stylophone, avec son signal audio analogique rudimentaire, se heurte à une limite majeure pour une utilisation musicale avancée : l'absence native de connectivité MIDI¹. L'intégration d'une sortie MIDI permettrait de transformer les notes jouées sur le clavier métallique en données numériques standardisées, compatibles avec les logiciels de production musicale, les synthétiseurs modernes ou les séquenceurs.

¹ MIDI : Abréviation de Musical Instrument Digital Interface

2. Simulation du Stylophone

Le Stylophone génère des sons grâce à un réseau de résistances discrètes, où l'addition de résistances détermine la hauteur de chaque note.

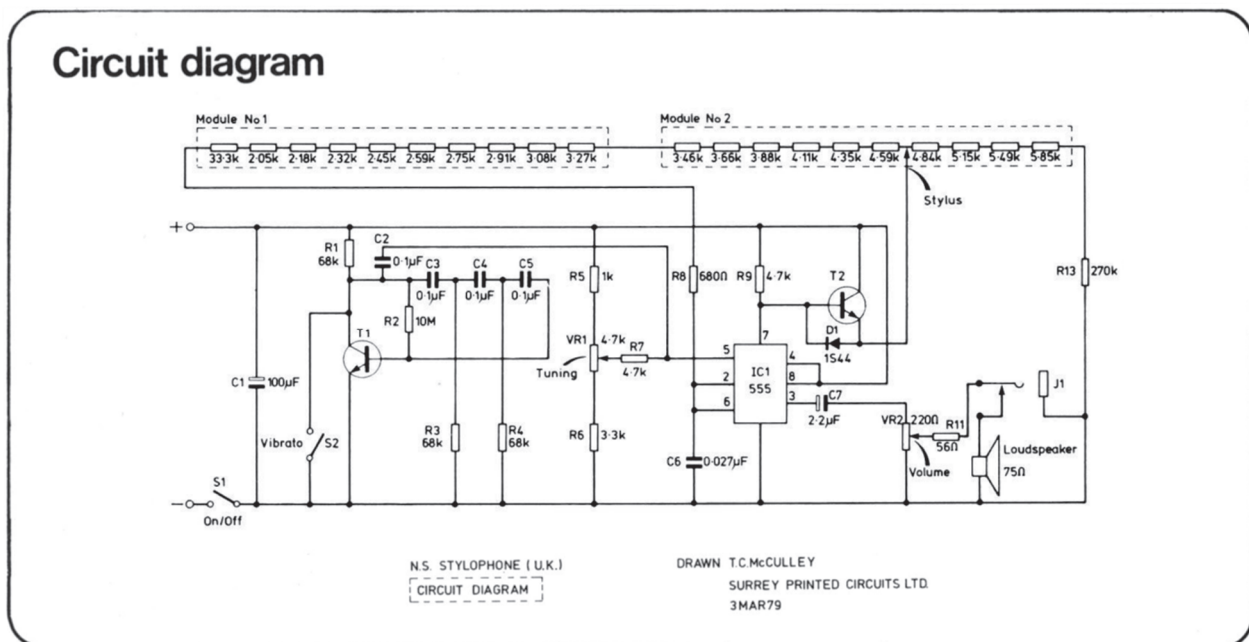


Figure 2 : Extrait d'un schéma de circuit du stylophone

Un générateur d'ondes, basé sur un circuit intégré 555, produit le signal audio de base.

Le clavier métallique est constitué de pistes conductrices reliées à des résistances.

Le stylo, doté d'une tige métallique, agit comme un contact mobile : lorsqu'il touche une touche, il ferme le circuit à cet endroit précis et ajoute la résistance correspondante au circuit principal.

Cette résistance modifie le comportement de l'oscillateur 555, ce qui change la fréquence du signal audio produit.

Dans la simulation suivante, chaque interrupteur permet l'activation d'une résistance supplémentaire :

Simple 555 piano



▶ Démarrer la simulation

Code

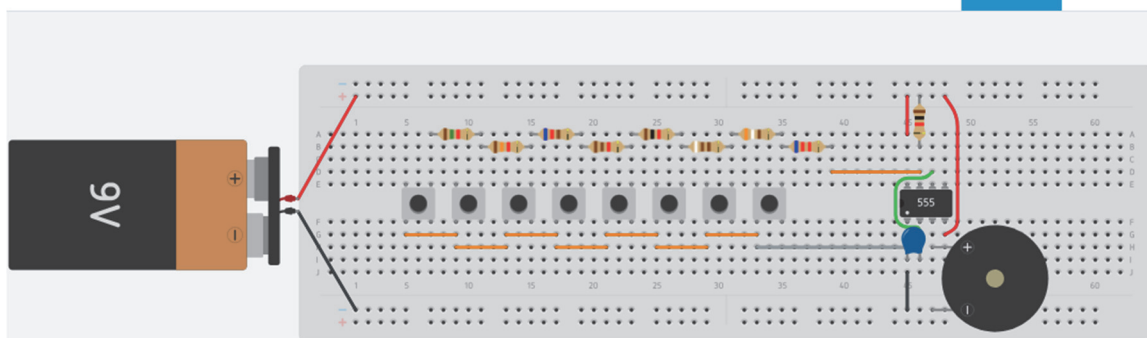


Figure 3 : Simple 555 piano © Mr.punch

<https://www.tinkercad.com/things/i0Fygixe6Pi-simple-555-piano>

3. Expérimentation

Pour effectuer l'expérimentation, le matériel suivant est à disposition :


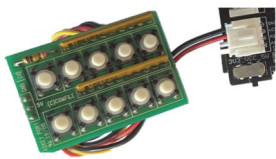

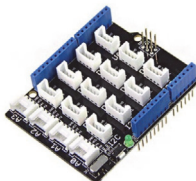
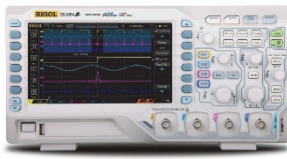

Système à analyser (synthétiseur analogique)	
Stylophone	
Matériel	
Clavier 10 boutons	
Carte de développement Arduino UNO	
Carte de connexion « Grove Base Shield »	
Buzzer	
Appareils de mesures :	
Accordeur matériel ou via une application smartphone	
Oscilloscope	
Ordinateur avec : <ul style="list-style-type: none"> • Un éditeur de code • Le logiciel de synthèse d'instruments de musique • Le logiciel de conversion Série / MIDI • Une connexion internet pour l'accès à la simulation 	

Figure 4 : Correspondance des notes et des fréquences



Fréquence	Note MIDI	Note Française	Note Anglaise	Note MIDI	Fréquence
		<i>Do8</i>	C8	108	4186.0
3729.3	106	<i>Si7</i>	B7	107	3951.1
3322.4	104	<i>La7</i>	A7	105	3520.0
2960.0	102	<i>Sol7</i>	G7	103	3136.0
		<i>Fa7</i>	F7	101	2793.8
		<i>Mi7</i>	E7	100	2637.0
2489.0	99	<i>Ré7</i>	D7	98	2349.3
2217.5	97	<i>Do7</i>	C7	96	2093.0
		<i>Si6</i>	B6	95	1975.5
1864.7	94	<i>La6</i>	A6	93	1760.0
1661.2	92	<i>Sol6</i>	G6	91	1568.0
1480.0	90	<i>Fa6</i>	F6	89	1396.9
		<i>Mi6</i>	E6	88	1318.5
1244.5	87	<i>Ré6</i>	D6	86	1174.7
1108.7	85	<i>Do6</i>	C6	84	1046.5
		<i>Si5</i>	B5	83	987.77
932.33	82	<i>La5</i>	A5	81	880.00
830.61	80	<i>Sol5</i>	G5	79	783.99
739.99	78	<i>Fa5</i>	F5	77	698.46
		<i>Mi5</i>	E5	76	659.26
622.25	75	<i>Ré5</i>	D5	74	587.33
554.37	73	<i>Do5</i>	C5	72	523.25
		<i>Si4</i>	B4	71	493.88
466.16	70	<i>La4</i>	A4	69	440.00
415.30	68	<i>Sol4</i>	G4	67	392.00
369.99	66	<i>Fa4</i>	F4	65	349.23
		<i>Mi4</i>	E4	64	329.63
311.13	63	<i>Ré4</i>	D4	62	293.66
277.18	61	<i>Do4</i>	C4	60	261.63
		<i>Si3</i>	B3	59	246.94
233.08	58	<i>La3</i>	A3	57	220.00
207.65	56	<i>Sol3</i>	G3	55	196.00
185.00	54	<i>Fa3</i>	F3	53	174.61
		<i>Mi3</i>	E3	52	164.81
155.56	51	<i>Ré3</i>	D3	50	146.83
138.59	49	<i>Do3</i>	C3	48	130.81
		<i>Si2</i>	B2	47	123.47
116.54	46	<i>La2</i>	A2	45	110.00
103.83	44	<i>Sol2</i>	G2	43	97.999
92.499	42	<i>Fa2</i>	F2	41	87.307
		<i>Mi2</i>	E2	40	82.407
77.782	39	<i>Ré2</i>	D2	38	73.416
69.296	37	<i>Do2</i>	C2	36	65.406
		<i>Si1</i>	B1	35	61.735
58.270	34	<i>La1</i>	A1	33	55.000
51.913	32	<i>Sol1</i>	G1	31	48.999
46.249	30	<i>Fa1</i>	F1	29	43.654
		<i>Mi1</i>	E1	28	41.203
38.891	27	<i>Ré1</i>	D1	26	36.708
34.648	25	<i>Do1</i>	C1	24	32.703
		<i>Si0</i>	B0	23	30.868
29.135	22	<i>La0</i>	A0	21	27.500

Matthijs Hollemans & Joe Wolfe, UNSW

Figure 5 : Algorithme MIDI

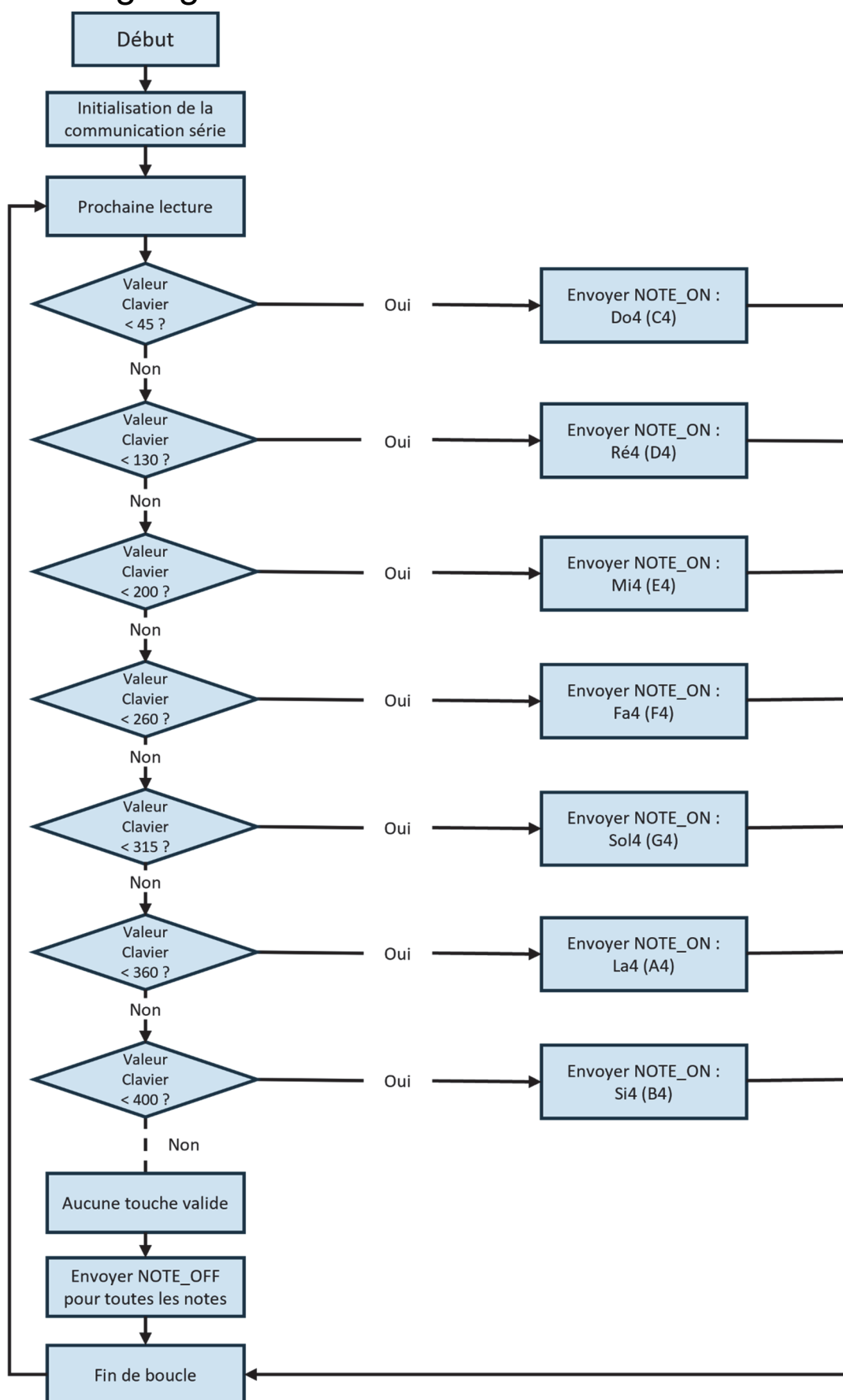


Figure 6 : « stylophone.ino »

```

stylophone.ino
1 // Définition des commandes MIDI
2 #define NOTE_ON 0x90 // Commande MIDI pour "Note On" (allumer une note sur le canal 1)
3 #define NOTE_OFF 0x80 // Commande MIDI pour "Note Off" (éteindre une note sur le canal 1)
4
5 const int BROCHE_CAPTEUR = A0; // Broche analogique A0 reliée au capteur ADKEY
6 const int BROCHE_BUZZER = 3; // Broche PWM pour le buzzer
7 int valeurClavier = 0; // Variable pour stocker la valeur lue du capteur
8
9 void setup() {
10   Serial.begin(9600); // Initialisation du port série
11   pinMode(BROCHE_BUZZER, OUTPUT); // Initialisation de la broche du buzzer en sortie
12 }
13
14 // Fonction d'envoi de messages MIDI via le port série
15 void envoyerMessageMIDI(byte commande, byte noteMIDI, byte velocite) {
16   Serial.write(commande); // Envoi de la commande (note on/off + canal)
17   Serial.write(noteMIDI); // Envoi de la hauteur de note (note MIDI)
18   if (commande == NOTE_ON || commande == NOTE_OFF) {
19     Serial.write(velocite); // Envoi de la vélocité (intensité de la note)
20   }
21 }
22
23 void loop() {
24   valeurClavier = analogRead(BROCHE_CAPTEUR); // Lecture de la valeur transmise par le clavier 10 boutons
25
26   /* Détection de la touche appuyée
27    et génération de la note correspondante sur le buzzer
28    puis envoi de la note MIDI correspondante via le port série */
29
30   /* Code à compléter : XXXXX et xx sont à remplacer */
31
32   if (valeurClavier < 45) { // Do4 (C4)
33     tone(BROCHE_BUZZER, 261.63);
34     envoyerMessageMIDI(NOTE_ON, 60, 100);
35   }
36   else if (valeurClavier < 130) { // Ré4 (D4)
37     tone(BROCHE_BUZZER, XXXXX);
38     envoyerMessageMIDI(NOTE_ON, xx, 100);
39   }
40   else if (valeurClavier < 200) { // Mi4 (E4)
41     tone(BROCHE_BUZZER, XXXXX);
42     envoyerMessageMIDI(NOTE_ON, xx, 100);
43   }
44   else if (valeurClavier < 260) { // Fa4 (F4)
45     tone(BROCHE_BUZZER, XXXXX);
46     envoyerMessageMIDI(NOTE_ON, xx, 100);
47   }
48   else if (valeurClavier < 315) { // Sol4 (G4)
49     tone(BROCHE_BUZZER, XXXXX);
50     envoyerMessageMIDI(NOTE_ON, xx, 100);
51   }
52   else if (valeurClavier < 360) { // La4 (A4)
53     tone(BROCHE_BUZZER, XXXXX);
54     envoyerMessageMIDI(NOTE_ON, xx, 100);
55   }
56   else if (valeurClavier < 400) { // Si4 (B4)
57     tone(BROCHE_BUZZER, XXXXX);
58     envoyerMessageMIDI(NOTE_ON, xx, 100);
59   }
60   else if (valeurClavier < 435) { // Do5 (C5)
61     tone(BROCHE_BUZZER, XXXXX);
62     envoyerMessageMIDI(NOTE_ON, xx, 100);
63   }
64   else if (valeurClavier < 470) { // Ré5 (D5)
65     tone(BROCHE_BUZZER, XXXXX);
66     envoyerMessageMIDI(NOTE_ON, xx, 100);
67   }
68   else if (valeurClavier < 512) { // Mi5 (E5)
69     tone(BROCHE_BUZZER, XXXXX);
70     envoyerMessageMIDI(NOTE_ON, xx, 100);
71   }
72   else {
73     // Arrêter le buzzer si aucune touche n'est appuyée
74     noTone(BROCHE_BUZZER);
75     // Éteindre toutes les notes si aucune touche n'est appuyée
76     envoyerMessageMIDI(NOTE_OFF, 60, 0);
77     envoyerMessageMIDI(NOTE_OFF, 62, 0);
78     envoyerMessageMIDI(NOTE_OFF, 64, 0);
79     envoyerMessageMIDI(NOTE_OFF, 65, 0);
80     envoyerMessageMIDI(NOTE_OFF, 67, 0);
81     envoyerMessageMIDI(NOTE_OFF, 69, 0);
82     envoyerMessageMIDI(NOTE_OFF, 71, 0);
83     envoyerMessageMIDI(NOTE_OFF, 72, 0);
84     envoyerMessageMIDI(NOTE_OFF, 74, 0);
85     envoyerMessageMIDI(NOTE_OFF, 76, 0);
86   }
87
88   delay(100); // Délai pour éviter les rebonds
89 }

```

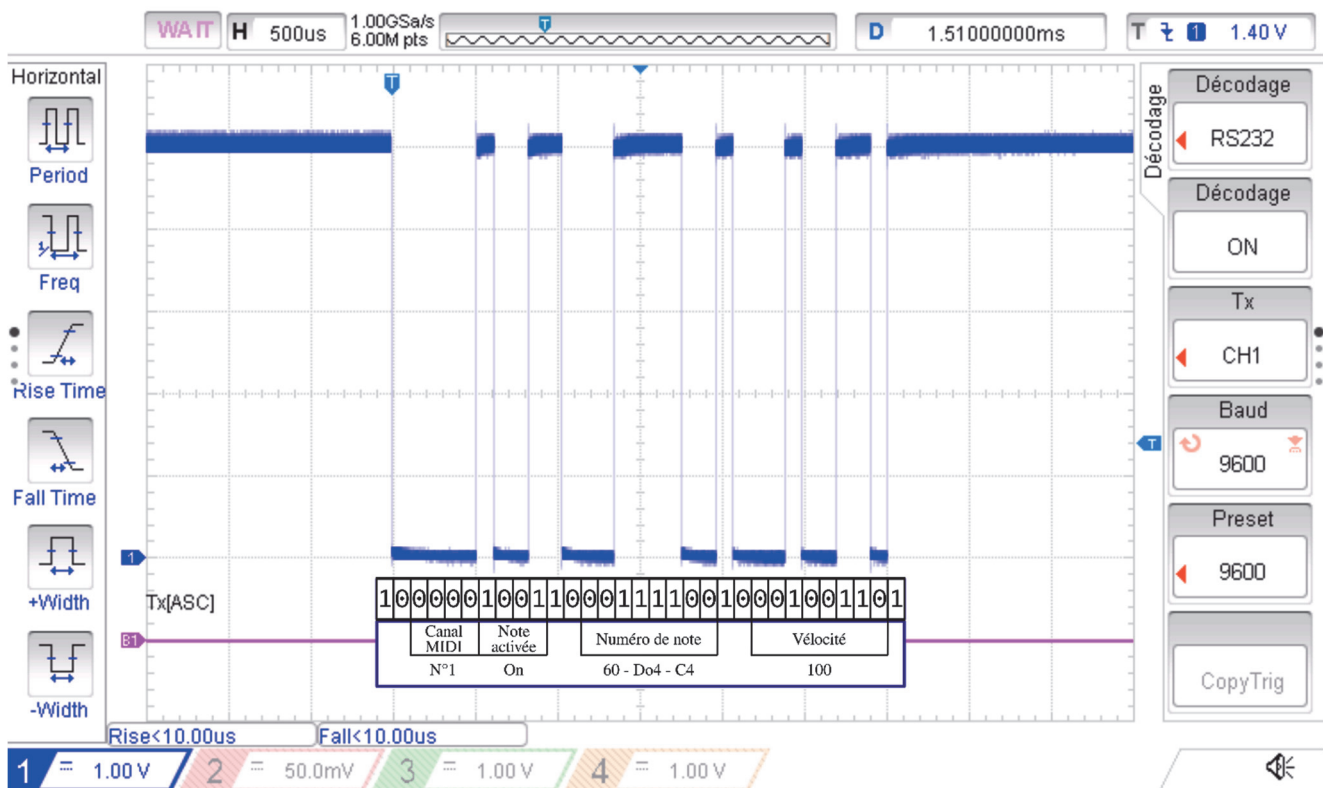
Figure 7 : Trame MIDI

Une trame MIDI est émise sur la liaison série octet après octet à $9600 \text{ bit}\cdot\text{s}^{-1}$

Elle se compose de 4 paramètres codés sur 3 octets :

- Le premier octet concatène² deux paramètres de 4 bits chacun :
 - L'activation ou l'extinction de la note : On (1001) ou Off (1000)
 - Le canal MIDI représente l'instrument joué : canal 1 pour un piano (0000), canal 2 pour une batterie (0001), canal 3 pour une flûte (0010)...
- Le numéro de note MIDI tel que décrit dans la Figure 4
- La vélocité qui correspond à la vitesse d'enfoncement d'une touche

Exemple d'activation d'un Do4 sur le Canal n°1 à la vélocité 100 :



Attention :

- Les bits de poids faible sont émis en premier :
La vélocité, ici à 100, soit en binaire 01100100, est reçue sur l'oscilloscope 00100110
- Les canaux MIDI sont numérotés de 1 à 16 sur les instruments, alors que l'indexation en informatique commence à 0.

² Concaténer : enchaîner, c'est-à-dire mettre bout à bout.