

I - Caractéristiques et branchements

1 - Modes de fonctionnement

La carte comporte deux modes, externe et autonome. Le mode externe fonctionne avec un ordinateur et un logiciel gérant le Midi, le mode autonome n'en a pas besoin. La carte démarre dans l'un ou l'autre mode en fonction de l'interrupteur numéro 2 :

Inter 2	
OFF	Mode externe
ON	Mode autonome

2 - Entrées / Sorties

En mode externe, la carte se comporte comme une carte d'entrées / sorties avec :

- 4 entrées pour capteurs analogiques utilisables en mode 7 bits ou 12bits

La résolution se choisit sur la carte à l'aide de l'interrupteur n°1

OFF = 7 bits, ON = 12 bits.

- 4 entrées numériques (tout ou rien)

- 4 sorties transistors de puissance

PWM, 128pas

- 4 sorties relais Reed

(100V 500mA) tout ou rien

- 4 sorties servomoteurs en 128 pas.

Les sorties servomoteurs se rajoutent sur la carte grâce à une plaque optionnelle.

3 - Alimentations électriques : attention au sens + / -

a) Alimentation de la carte et des capteurs (bleu foncé sur le schéma)

Elle doit délivrer une tension continue (bloc secteur/pile/batterie), comprise entre 11V et 20 V, et un courant de 100 mA minimum.

Il y a une polarité à respecter, la carte ne fonctionnera pas si l'alimentation est branchée à l'envers. Le (-) est du côté des connecteurs pour capteurs.

Les fils d'alimentation doivent être dénudés proprement au bout sur 5 mm avant d'être vissés dans les dominos de la carte conformément à la photo ci-dessous.



Alimentation de la carte

b) Alimentation des servomoteurs

Elle n'est pas fournie par la carte, il faut la brancher sur la carte de connexion des servomoteurs. La tension doit être continue et correspondre aux servomoteurs utilisés (souvent 5V). Il y a une polarité à respecter. Le (+) est vers le bord de la carte, le (-) vers les connecteurs.

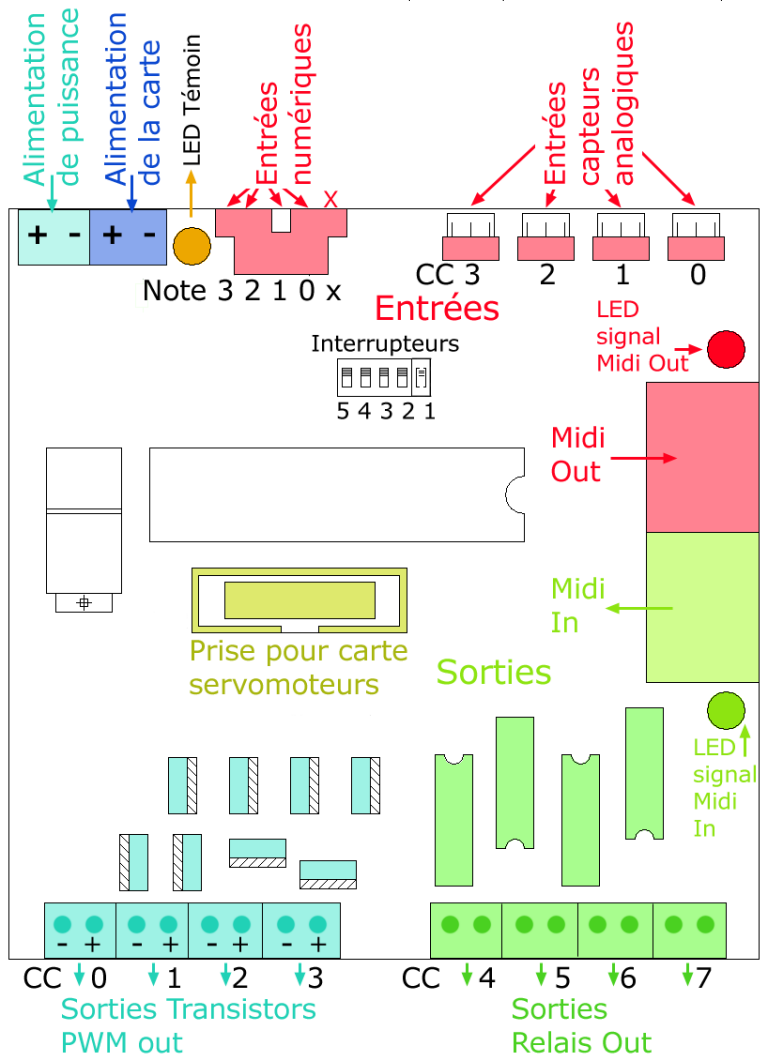
c) Alimentation de puissance PWM (turquoise sur le schéma)

Elle correspond à l'alimentation des actionneurs branchés sur les sorties PWM. Elle n'est QUE continue et doit être comprise entre 1V et 60 V. Il y a une polarité à respecter, sous peine de destruction des sorties, des actionneurs voire de la carte. Le (+) est vers le bord de la carte.

4 - Connectique

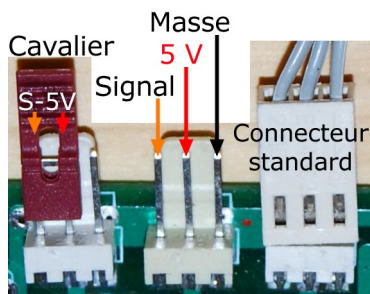
a) Midi - Connectique standard (DIN 5 broches)

La sortie Midi Out est située du côté des connecteurs pour capteurs. Elle se branche sur une entrée Midi. En mode externe, elle transmet les données des capteurs vers l'extérieur, sur l'entrée de l'ordinateur (via une interface Midi-USB) ou d'une autre carte ou d'un merger. En mode autonome, la sortie Midi envoie ce que le patch mémorisé lui demande d'envoyer, vers un ordinateur ou vers d'autres cartes.



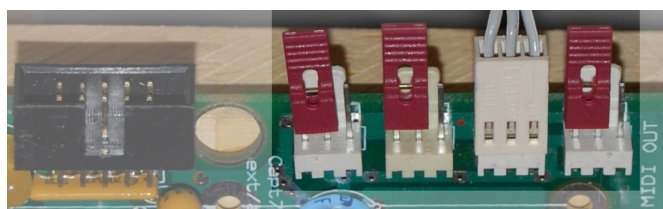
L'entrée Midi In est située du côté des actionneurs. Elle reçoit les consignes de pilotage des actionneurs (en mode externe) et permet le téléchargement des patches dans la carte. Elle se branche sur une sortie Midi d'ordinateur en mode externe. En mode autonome, elle peut aussi recevoir des informations envoyées par d'autres cartes, par exemple des capteurs supplémentaires.

b) Capteurs analogiques / graduels



Les capteurs sont connectés via des rallonges standards IZ. Le connecteur Interface-Z est muni d'un détrompeur qui évite de se tromper de sens et d'un verrouillage qui évite le débranchement involontaire des capteurs. En cas d'utilisation d'une autre connectique, il faut faire attention au sens de branchement et ne pas intervertir les fils pour ne pas provoquer de court-circuit.

LES ENTREES CAPTEURS NON UTILISEES DOIVENT ETRE MUNIES D'UN BOUCHON (entrée signal reliée à la broche voisine 5V) pour éviter la transmission d'informations parasites.



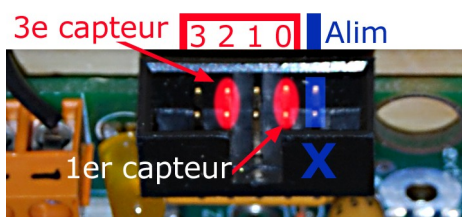
ATTENTION : une erreur de positionnement du jump provoque un **court-circuit** sur la carte et empêche complètement la carte de fonctionner (avec risques de destruction).

Sur la carte, le bouchon est toujours sur les deux picots le plus vers le connecteur pour entrées numériques.

c) Capteurs numériques / interrupteurs

Quatre capteurs tout ou rien peuvent se brancher sur cette carte, à condition que ce soient des interrupteurs à deux fils. Le schéma indique l'ordre de branchement en fonction des numéros des Note On envoyés (cf plus loin).

Le plus simple est d'utiliser un connecteur HE10 pour câble en nappe. Les fils se raccordent deux par deux aux capteurs (le schéma montre en rouge deux entrées différentes de capteurs).



Attention, les picots marqués d'un X, du côté des connecteurs analogiques, ne sont pas une entrée capteur mais fournissent une alimentation. Il est inutile d'y brancher un interrupteur.

d) Actionneurs - Sorties transistors de puissance - Gradation PWM

- **Types d'actionneurs** : Les actionneurs pilotables par ces sorties doivent supporter une alimentation variable en basse tension continue, comme par exemple les électro-aimants, les ampoules 12 Volts halogènes ou LED ou les moteurs à courant continu. Certains moteurs pas à pas sont compatibles avec la carte mais un seul occupe les 4 sorties.

- **Le branchement** et la commande de ces actionneurs sont spécifiques à chacun. Par exemple les ampoules 12V ne sont pas polarisées, les LED le sont et le sens de rotation d'un moteur CC dépend du sens d'alimentation, certains électroaimants sont alimentés en 5 V, d'autres en 24 V. D'une façon générale, le sens +/- des sorties est donné sur le premier schéma de cette documentation. ATTENTION, si l'actionneur est polarisé, il ne faut surtout pas le brancher à l'envers pour ne pas risquer de le détruire.

- **Alimentation** : Il faut toujours alimenter ces actionneurs par le connecteur d'alimentation de puissance, situé à côté de celui de la carte (voir §3-c). Il **ne faut pas inverser** les fils (+) et (-).

La tension de cette alimentation doit correspondre aux actionneurs branchés. Ce doit être de la basse tension, continue, de 60 V maximum. Les quatre sorties utilisent cette même alimentation, elle doit donc être dimensionnée en courant pour l'ensemble des actionneurs utilisés. Il suffit d'additionner les consommations des 4 actionneurs.

- **Puissance** : chaque sortie ne supporte que 2 A au maximum en courant, pour des actionneurs alimentés en 12V. Certaines versions de la carte permettent de piloter des actionneurs plus puissants mais pas la version de base.

e) Actionneurs - Sortie Relais Reed - Tout ou rien

- **Interrupteurs commandés** : les relais Reed s'intercalent sur des interrupteurs de configuration (par exemple sur certains gradateurs, sur des télécommandes) ou bien sur des actionneurs de faible puissance. Il ne s'agit pas de relais de puissance. Il ne supportent pas d'être connectés sur le secteur, dans le cas de la version de base de la carte.

- **Branchement** : les relais sont sur la carte. Il y a un domino de sortie à deux plots par relais.

Dans le cas d'un pilotage d'interrupteur, le relais remplace le bouton (touche de clavier, touche de télécommande, etc). Les fils branchés dans le domino arrivent de par et d'autre du bouton cible. Activer le relais revient à appuyer sur le bouton.

Dans le cas d'un pilotage d'actionneur en tout ou rien, le relais s'insère sur l'un des fils d'alimentation. Actionner le relais rétablit la continuité du fil et revient à alimenter l'actionneur.

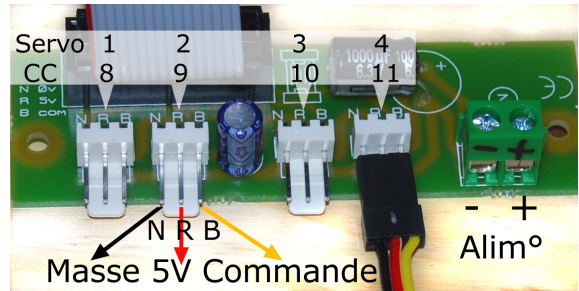
- **Tension maximale supportée** : 100 V en tension continue ou alternative (pour la version de base).

- **Puissance** : le courant doit être inférieur à 0,4 A.

f) Actionneurs - Servomoteurs

- **Carte additionnelle** : les moteurs se branchent sur une petite carte optionnelle. Cette carte est raccordée à la carte ZIP par un câble en nappe 16 fils. Il y a un sens à respecter pour ce raccord. Les connecteurs du câble fourni sont équipés de détrompeurs et ne peuvent être branchés à l'envers.

- **Alimentation** : ne pas inverser le (+) et le (-). Le (-) est du côté des connecteurs. La tension doit correspondre aux servomoteurs.



- **Branchement des moteurs** : La carte optionnelle comporte 4 connecteurs à 3 picots. Les moteurs **ne doivent pas être branchés à l'envers**. Le marquage NRB permet de se repérer par rapport aux couleurs des fils des servos.

Marquage de la carte	N	R	B
Electricité	0 V - Masse	5 V	Commande
Couleurs	Noir, Marron	Rouge	Blanc, Jaune, Orange

5 - Configuration par interrupteurs

Seuls les deux premiers ont un rôle, les autres sont réservés à un usage futur.	ON	OFF
Interrupteur 1 : résolution des capteurs analogiques	12 bits - 4096 pas	7 bits - 128 pas - Midi standard
Interrupteur 2 : mode de fonctionnement.	Mode autonome	Mode externe

Mode externe ou autonome : l'interrupteur donne le mode de démarrage de la carte. Dans les deux cas l'ordinateur peut reprendre le contrôle de la carte.

II - Mode externe - Réception et envoi de données à partir d'un logiciel

1 - Compatibilité

En mode externe, la carte est une carte d'entrées / sorties simple, avec un ordinateur ou tout autre matériel recevant et envoyant des messages Midi. Les données des capteurs sont reçues dans un logiciels, les commandes des actionneurs sont envoyées depuis ce même logiciel. La carte est alors compatible avec tous les logiciels gérant le protocole Midi. Dans ce mode, il n'y a pas de programme chargé dans la carte, les données sont gérées à l'extérieur de la carte.

2 - Messages Midi entrant - Capteurs

a) Entrées analogiques

La carte utilise des messages Midi Control Change. Les entrées analogiques peuvent envoyer les données en deux résolutions, 128 pas (7 bits) ou 4096 pas (12 bits). Le canal Midi est fixe, c'est le premier.

Entrées analogiques	Inter n° 1	Numéros de Control Change (CC)			
		0	1	2	3
Mode 7 bits	OFF				
Mode 12 bits	ON	0 à 31	32 à 63	64 à 95	96 à 127

En mode 7 bits, la valeur du capteur est directement accessible par la valeur du CC. Les numéros de CC servent à distinguer les différentes entrées. Par exemple un capteur branché sur la 2e entrée envoie des messages CC de numéro 1.

En mode 12 bits, le numéro de Control Change participe aussi à la valeur du capteur. Il faut reconstituer cette valeur à l'arrivée en combinant valeur et numéro de CC.

- Diviser le numéro de CC par 32
- Le résultat de cette division permet de trier les capteurs, c'est l'équivalent des 0, 1, 2 ou 3 du mode 7 bits. Par exemple, un capteur branché sur la première entrée correspond au résultat 0.
- Le reste de la division est utilisé pour reconstituer la valeur du capteur :

$$\text{Valeur capteur} = (\text{reste} \times 128) + \text{valeur CC}$$

Par exemple, le message CC n° 34 valeur 58 est une donnée envoyée par le 2e capteur, de valeur $(2 \times 128) + 58 = 314$.

b) Entrées numériques

Les entrées numériques envoient des Note On de vélocité 0 ou 64. Les pitches des Note servent à trier les capteurs et la vélocité correspond à l'état du capteur. Dans certains logiciels, un Note On vélocité 0 est interprété comme un Note Off 64.

Pitches des Note On des entrées numériques	0	1	2	3
--------------------------------------------	---	---	---	---

3 - Messages Midi sortant - Actionneurs

La carte utilise des messages Midi Control Change seulement. La résolution de sortie est de 128 pas. Le canal Midi est le premier (0 ou 1 selon le logiciel). Attention au running status en sortie Midi.

Actionneurs	Numéros de Control Change			
Sorties transistors PWM	0	1	2	3
Sorties relais Reed	4	5	6	7
Sorties servomoteurs	8	9	10	11

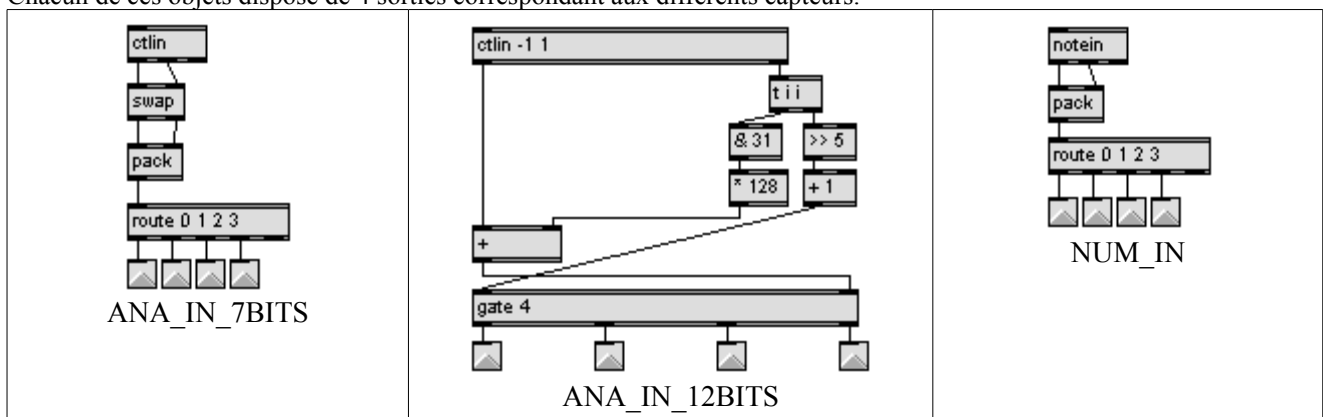
4 - Abstractions Max/MSP

Des patches sont disponibles sur le site www.interface-z.com, section Programmation. Les patches détaillés dans cette documentation sont ceux pour Max, mais sont adaptables à tout autre logiciel de gestion Midi temps réel. Ils ne sont compatibles qu'avec le mode externe de la carte. Consultez le site pour des fichiers d'autres logiciels.

a) Les données d'entrée sont accessibles à l'aide des abstractions

ANA_IN_7BITS et ANA_IN_12BITS pour les entrées analogiques ;
 NUM_IN pour les entrées tout ou rien.

Chacun de ces objets dispose de 4 sorties correspondant aux différents capteurs.



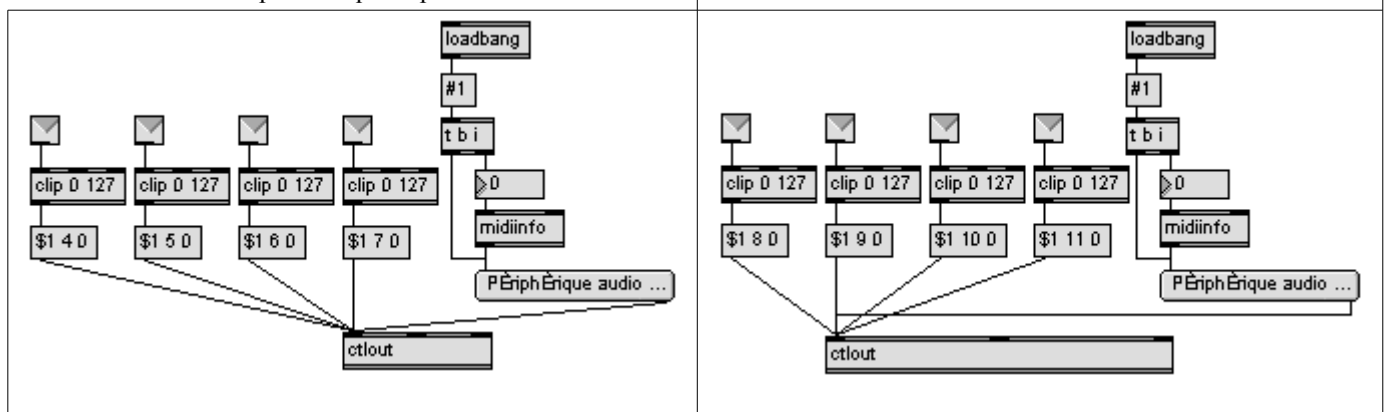
b) Les sorties sont pilotables à l'aide des abstractions

PWM_OUT pour les sorties transistors ;
 RELAI_OUT pour les sorties relais ;
 SERVO_OUT pour les sorties servomoteurs.

Chacun de ces objets dispose de 4 entrées correspondant aux différents actionneurs.

c) Utilisation des abstractions

Une abstraction Max est enregistrée dans un fichier indépendant du patch principal. Pour y accéder il faut mettre le fichier soit dans un répertoire dont le chemin est déclaré soit dans le même répertoire que le patch.



d) **Patch de chargement** : Le mode externe permet aussi de configurer la carte pour le mode autonome, à l'aide d'un patch spécial de chargement qui sera abordé plus loin.

II - Mode autonome

A - Caractéristiques : modes externe / autonome

L'intérêt de mettre la carte en mode autonome est qu'elle fonctionne à terme sans ordinateur. Elle utilise les données des capteurs et pilote les actionneurs grâce à un patch spécifique chargé dans la carte. Actuellement, la carte est seulement compatible avec Max 4.6.

En mode autonome la carte est capable d'exécuter le MEME PATCH MAX qu'utilisé en mode externe, à condition de :

- respecter certaines règles de programmation énoncées ci-dessous,
- respecter certaines limitations énoncées ci-dessous,
- avoir téléchargé le patch dans la mémoire embarquée.

La carte peut stocker 128 patches différents, avec au maximum 64 objets par patch, et peut basculer automatiquement d'un patch à l'autre.

Le passage du mode externe au mode autonome se fait par configuration d'un interrupteur sur la carte. Il est aussi possible de basculer entre ces deux modes par le patch Gestion, qui permet de tester et de charger le patch dans la carte.

Interrupteur n°2	Actions	Logiciels	Fonctionnement final	LED témoin
OFF	Mode externe	Tous	Avec ordinateur	Allumée
	Test du patch à télécharger	Max 4.6	Sans ordinateur	Allumée
	Patch Gestion : Chargement du patch final dans la carte, passage en mode autonome	Max 4.6	Sans ordinateur	Séteint
ON	Mode autonome	Max 4.6	Sans ordinateur	Cligne*
	Patch Gestion : Chargement du patch final dans la carte, reprise du contrôle de la carte par l'ordinateur, passage en mode externe pour tester le patch final.	Max 4.6	Sans ordinateur	Se rallume

*Cligne : Allumée pour jump ou débordement Midi.

La position de l'interrupteur est surtout importante lors du démarrage de la carte. Lorsque la carte est connectée à un ordinateur elle peut passer d'un mode à l'autre sans qu'il faille toucher l'interrupteur.

B - Règles de programmation à respecter, limitations par rapport à MAX

En mode autonome, la carte ZIP exécute un programme, elle fait fonctionner un équivalent de Max comme langage de programmation, compatible avec Max mais nécessitant des abstractions et une programmation adaptée avec les objets qu'elle peut comprendre. Un patch de traduction permet d'envoyer le programme dans la carte via le protocole Midi.

Les objets de programmation que la carte peut comprendre sont de deux types :

- des objets standards de max
- ou bien des abstractions contenant des objets standards.

Les deux types s'utilisent de la même façon, il faut seulement s'assurer d'avoir les fichiers des abstractions disponibles avant de les utiliser. La carte comprend des objets similaires à Max, mais tous les objets de Max ne sont pas présents : il y a donc des abstractions qui fournissent des fonctions similaires à certains objets Max mais qui fonctionnent de manière un peu différente. Les abstractions sont traduites par le patch de gestion. Certaines nécessitent que le patch de gestion soit ouvert.

Comme pour les abstractions en programmation classique, il faut que le fichier soit disponible, soit dans le même répertoire que le patch, soit dans un répertoire dont le chemin est déclaré dans le logiciel.

Les règles de programmation explicitées ci-dessous sont impératives : si elles ne sont pas respectées, le patch ne sera pas chargé dans la carte et ne fonctionnera pas, même s'il est correct en programmation classique dans le logiciel.

0 - Compatibilité logicielle

Le mode autonome ne fonctionne qu'avec Max4.xx.yy ou plus ancien pour l'instant, le traducteur Max5 n'étant pas encore écrit. Une fois écrit le traducteur de patch, la carte fonctionnera aussi avec Pure Data.

1 - Le patch doit comporter au plus 64 objets avec les restrictions suivantes :

- Le premier objet posé est obligatoirement un **loadbang**.
- Il y a un seul objet ANA_IN_7BITS (ou ANA_IN_12BITS) et un seul objet NUM_IN dans tout le patch.
- Il ne faut pas dépasser une limite de 16 objets de type **metro** ou **delay** (les deux confondus).

2 - Le patch doit être sauvé en .txt (texte) pour pouvoir lancer l'interpréteur qui le charge dans la carte.

3 - Restriction sur le type de variables

Seuls deux types de variables sont reconnus :

- **bang** : envoyé par les metro et delay ou le trigger
- **int** : nombres entiers sur 16 bits de -32768 à 32737

REMARQUES - pas de listes ;
- pas de nombres à virgule (float) ;
- attention si un calcul dépasse les bornes, apparition d'un résultat erroné (repliement).

4 - Restriction sur les messages

Seuls deux types de message sont reconnus :

- **bang** : plus efficace qu'un objet "t b" en vitesse d'exécution ;
- **nombre** : nombres entiers sur 16 bits de -32768 à 32737.

REMARQUE : - pas de liste de nombres, un seul nombre ou une seul bang par message.
- pas de message stop, clear, reset, utiles pour certains objets.
- donc ces messages ne sont pas interprétables par les objets en question.

5 - Restriction sur l'objet trigger

- un maximum de 3 paramètres : l'objet ne peut pas comporter plus de 3 sorties.
- les seuls types autorisés sont les suivants :
 - b : pour déclencher un bang
 - i : pour recopier en sortie le nombre en entrée
 - 0 : pour envoyer une valeur 0
 - 1 : pour envoyer une valeur 1

REMARQUE : - Sont interdits les paramètres de types s, l, a, ou tout nombre différent de 0 ou 1.

5 - bis Restriction sur l'objet bangbang (lettre b)

- Maximum de 5 paramètres : l'objet ne peut pas comporter plus de 5 sorties : "b 5".

6 - Restriction sur les LIENS entre objets

- Un seul fil de connection par sortie (usage de l'objet trigger obligatoire pour les envois multiples).
- Pas de limitation sur le nombre de fils connectés sur une entrée.

REMARQUE : - de 1 à 3 fils, il faut l'objet trigger, ensuite il faut un trigger de plus tous les 2 fils supplémentaires car le trigger doit lui aussi respecter la même règle, pas plus d'une connection par sortie.




7 - Restrictions sur les types d'objets et la structure du patch


- pas d'objet patcher ou d'externals autres que ceux définis ci-dessous dans le patch
- pas d'objets nombres dans le patch à télécharger (les détruire avant de télécharger)
- pas d'objets slider
- pas d'objets print
- pas de button (bang jaune), les remplacer par un message contenant bang
- pas de send et receive

Les chapitres suivants explicitent les objets et abstractions utilisables par la carte ZIP. Tout objet non reconnu rend le patch non téléchargeable.



C - Objets de base

1 - Objets de programmation générale (objets standards)




toggle 	Comportement	- Comportement identique à Max : bascule entre On et Off, entre 0 et 1. - Le bang active et désactive le toggle ; - Le 1 l'active et le 0 le désactive ; - Il fait passer les valeurs positives et négatives.
int, i 	Comportement	- Comportement identique à Max : garde un nombre en mémoire. - Il peut s'écrire int, i ou juste le nombre par défaut (par exemple 10).
messages 	Comportement	- Comportement identique à Max.
	RESTRICTIONS	- bang ou b : plus efficace qu'un objet "t b" en vitesse d'exécution. - Il est impossible de changer le contenu d'un message avec "set". - Nombre : nombres entiers sur 16 bits de -32768 à 32767. Ces bornes sont incluses dans les limites des nombres ; au-delà, une boucle apparaît (32768 devient - 32768, ou -32769 devient 32767).

		- Le seul mot accepté est bang, les messages ne contiennent sinon que des nombres.
comment	Comportement	Commentaire.
	RESTRICTIONS	Il ne faut jamais faire de copier/coller sur un commentaire pour en faire un nouveau. Il faut créer à chaque fois un nouveau commentaire.


2 - Objets gérant les envois multiples et l'absence de plusieurs ficelles (objets standards)

trigger, t	Comportement	- Comportement identique à Max : envoie l'entrée sur plusieurs sorties dans l'ordre depuis la droite vers la gauche.
	RESTRICTIONS	- Maximum de 3 paramètres: l'objet ne peut pas comporter plus de 3 sorties - Les seuls types autorisés sont les suivants: - b : pour déclencher un bang - i : pour recopier en sortie le nombre en entrée - 0 : pour envoyer une valeur 0 - 1 : pour envoyer une valeur 1 - Sont interdits les paramètres de types s, l, a, ou tout nombre différent de 0 ou 1. - La restriction sur le câblage rend l'objet indispensable et force à réfléchir sur l'ordre d'exécution des objets.
bangbang, b	Comportement	- Comportement identique à Max : envoie des bangs de droite à gauche.
	RESTRICTIONS	- Maximum de 5 paramètres : l'objet ne peut pas comporter plus de 5 sorties. - La restriction sur le câblage rend l'objet indispensable et force à réfléchir sur l'ordre d'exécution.

3 - Objets temporels, activés par le scheduler (Objets standards)

loadbang	Comportement	- Comportement identique à Max : envoie un bang au démarrage du patch.
	RESTRICTIONS	- Il faut un loadbang unique par patch obligatoirement présent et posé en premier. - Il ne faut rien brancher sur l'entrée de cet objet.
	REMARQUE	- Encas de remaniement du patch, pour remettre l'objet loadbang en premier, il suffit de tout couper sauf le loadbang et de tout remettre
closebang	Comportement	- Comportement identique à Max : envoie un bang à la fermeture du patch. - A utiliser si un objet jump (voir plus loin) est présent dans le patch.
metro	Comportement	- Comportement identique à Max : métronome envoyant des bangs sur sa sortie.
	RESTRICTION	- Pas plus de 16 objets metro et delay présents dans le patch.
	REMARQUE	- Un metro 0 dans la carte se comporte comme un metro 1, contrairement à Max pour lequel ce serait metro full speed.
delay, del	Comportement	- Comportement identique à Max : quand il est activé par un message ou un bang, envoie un bang après un délai donné en ms. L'entrée droite permet de modifier la durée d'attente.
	RESTRICTIONS	- Pas plus de 16 objets metro et delay présents dans le patch. - N'accepte pas de message STOP donc pas d'arrêt possible. - Les nombres manipulés étant sur 16 bits, le delay max est de 32,767 secondes.
	REMARQUES	- Un delay négatif correspond à delay 0. - Si delay est activé en permanence, il ne sort rien car il redémarre en permanence. - Un nombre envoyé sur l'entrée gauche remplace la durée d'attente par sa valeur en ms et déclenche le delay.

4 - Gestion des tables (Abstractions spécifiques)

accestable	Comportement	- Fonctionnement comme l'objet standard table de Max - Simulation par une abstraction max avec présence obligatoire d'un patch maitre
	RESTRICTIONS	- Tables accessibles en lecture seule. - Limitation à 256 tables sur la carte. - Chaque table a une taille de 128 entiers 16bits.
	REMARQUE	Il n'y a pas de contrôle sur l'index auquel on cherche à accéder, il est donc possible de

créer des tables plus grandes mais c'est à vos risques et périls...

La table max est donc de 32768 int 16 bits.

Pour faire une table de 512 entiers, il faudra charger 4 tables consécutives en mémoire et faire l'appel sur la première table des 4.

Si l'on veut diviser la mémoire en 4 tables identiques, il faudra charger les 256 tables et travailler avec les tables 0, 64, 128, 192 à qui on passera un index entre 0 et 8191.

ECRIT_TABLE : description en cours de rédaction.

D - Gestion du protocole Midi

1 - Entrées Midi

a) Objets Midi In standards

ctlin	Comportement	- Comportement identique à Max : reçoit un message Midi Control Change. - Sortie gauche : valeur, centrale : numéro, droite : canal.
notein	Comportement	- Comportement identique à Max : reçoit un message Midi Note On. - Sortie gauche : pitch, centrale : vitesse, droite : canal.
pgmin	Comportement	- Comportement identique à Max : reçoit un message Midi Program Change.

b) Abstractions Midi In

Ces abstractions permettent de recevoir facilement des informations d'autres cartes IZ.

QUATRE_NOTEIN





QUATRE_CTLIN

PAIRECTL

MINIHF

2 - Sorties Midi

a) Objets Midi Out standards

ctlout 	Comportement	- Comportement identique à Max : envoie un message Midi Control Change. - Entrée gauche : valeur, entrée centrale : numero, entrée droite : canal.
	RESTRICTIONS	- La pile d'envoi midi comporte 8 messages complets ; en cas de pile pleine, le message part à la poubelle et ne sera jamais envoyé, contrairement à Max qui bufferise. - Il ne faut pas mettre une lettre en paramètre, on peut mettre numero et canal midi.
noteout 	Comportement	- Comportement identique à Max : envoie un message Midi Note On. - Entrée gauche : pitch, entrée centrale : vitesse, entrée droite : canal.
	RESTRICTIONS	Idem ctlout.
pgmout 	Comportement	- Comportement identique à Max : envoie un message Midi Program Change. - Entrée gauche : valeur, entrée droite : canal.
	RESTRICTIONS	Idem ctlout.
bendout 	Comportement	- Comportement identique à Max : envoie un message Midi Pitch Bend. - Entrée gauche : valeur, entrée droite : canal.
	RESTRICTIONS	Idem ctlout.

b) Abstractions Midi Out

Ces abstractions permettent de contrôler en Midi d'autres cartes IZ depuis la ZIP ou d'envoyer de l'information au logiciel.

quatredoublectlout (pour piloter des servomoteurs 3500 pas)

MIDI_DMX

QUATRE_CTLOUT

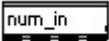
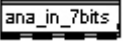
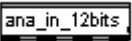
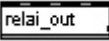
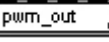
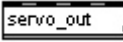
QUATRE_NOTEOUT

c) Abstraction pour contrôler la pile Midi Out

INFOMIDI permet de savoir combien il reste de place pour écrire ou pas un nouvel ordre midi ; il permet de gérer au plus vite un flux midi, en écrivant tant qu'il y a de la place, sans risquer la perte d'ordres.

3 - Abstractions spécifiques aux entrées/sorties de la carte

Elles donnent accès aux capteurs et actionneurs de la carte.




num_in 	Comportement	<ul style="list-style-type: none"> - Permet de recevoir les données des 4 entrées numériques. - Simule à l'identique l'abstraction num_in en mode externe.
	RESTRICTIONS	Un seul objet num_in par patch, sinon seul le dernier posé est utilisé
	Description	<ul style="list-style-type: none"> - Pas d'entrées - 4 sorties permettant de récupérer les entrées num (valeur OFF 0, valeur ON 127)
ana_in_7bits ana_in_12bits  	Comportement	<ul style="list-style-type: none"> - Permet de recevoir les données des 4 entrées analogiques. - Simule les abstractions ana_in_7bits ou ana_in_12bits en mode externe.
	RESTRICTIONS	Un seul objet par patch, sinon seul le dernier posé est utilisé
	Description	<ul style="list-style-type: none"> - Pas d'entrées, - 4 sorties pour récupérer les entrées analogiques en résolution configurable : <ul style="list-style-type: none"> - interrupteur 1 baissé configure des données de 0 à 127, - interrupteur 1 levé configure des données de 0 à 4095 .
relai_out 	Comportement	<ul style="list-style-type: none"> - Permet de piloter les 4 relais Reed en sortie. - Simule à l'identique l'abstraction relai_out en mode externe.
	Description	<ul style="list-style-type: none"> - Pas de sorties - 4 entrées permettant de piloter les 4 relais individuellement <ul style="list-style-type: none"> - entrée valeur nulle : relai off - entrée valeur non nulle : relai on
pwm_out 	Comportement	<ul style="list-style-type: none"> - Permet de piloter les 4 Transistors en sortie - Simule à l'identique l'abstraction pwm_out en mode externe.
	Description	<ul style="list-style-type: none"> - Pas de sorties - 4 entrées permettent de piloter les 4 transistors individuellement <ul style="list-style-type: none"> - entrée valeur nulle ou négative: transistor OFF - entrée valeur x non nulle de 1 à 126: <ul style="list-style-type: none"> transistor ON x/127ième du temps, OFF le reste du temps - entrée valeur 127 ou plus : transistor ON
	REMARQUE	Les PWM fonctionnent à 500 Hz.
servo_out 	Comportement	<ul style="list-style-type: none"> - Permet de piloter les 4 Servomoteurs en sortie. - Simule à l'identique l'abstraction servo_out en mode externe.
	Description	<ul style="list-style-type: none"> - Pas de sorties - 4 entrées permettent de piloter les 4 servomoteurs individuellement <ul style="list-style-type: none"> - entrée valeur nulle ou négative : impulsion de 500ms toutes les 20ms - entrée valeur x non nulle de 1 à 126 : <ul style="list-style-type: none"> impulsions de duree proportionnelle à x toutes les 20ms - entrée valeur 127 ou plus : impulsion de 2,4ms toutes les 20ms
	REMARQUE	Les servomoteurs fonctionnent à 50 Hz (50 actualisations par seconde), fréquence des servomoteurs standards.



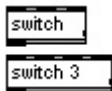





E - Gestion de flux

La donnée est aiguillée, dispatchée, transmise / non transmise...



1 - Traitement des données

a) Objets standards de Max

gate   	Comportement	<ul style="list-style-type: none"> - Comportement identique à Max : arrête ou autorise la transmission d'un flux de données sur une sortie. - Si consigne de porte (entrée chaude) = 0 : les valeurs ne passent pas (gate fermé) ; - Si consigne de porte différent de 0 (en + ou en -) : les valeurs passent (gate ouvert). - Quand la consigne de porte est supérieure au nombre de sorties, c'est la dernière sortie qui est ouverte, quand elle est négative, c'est la première sortie.
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

		<p>Par exemple pour un gate 6 :</p> <ul style="list-style-type: none"> si consigne = 0, les valeurs ne passent pas (gate fermé) si consigne > 6, la sortie 6 est ouverte si consigne < 0, la sortie 1 est ouverte.
	RESTRICTION	<ul style="list-style-type: none"> - Limite de 6 sorties maximum. - Les autres options du gate ne sont pas implémentées (activation sur front montant ou descendant).
Gswitch	Comportement	<ul style="list-style-type: none"> - Comportement identique à Max : bascule entre les deux entrées de droite (froides). - Au démarrage c'est l'entrée 0 qui est ouverte ; - Si bang sur l'entrée chaude à gauche, l'entrée 1 bascule sur la sortie. - Si bang sur l'entrée chaude, bascule entre les deux entrées ; - Si entrée chaude = 0, l'entrée 0 est ouverte ; - Si entrée chaude différente de 0 (+ ou -), l'entrée 1 est ouverte. - Un bang sur une entrée froide passe de la même façon qu'un nombre.
		
Gswitch2	Comportement	<ul style="list-style-type: none"> - Comportement identique à Max : bascule l'entrée de droite entre deux sorties. - Au démarrage les valeurs (entrée froide droite) sortent sur la sortie de gauche ; - Si bang sur l'entrée chaude à gauche, le gate droit est ouvert. - Si bang sur l'entrée chaude, bascule entre les deux sorties ; - Si entrée chaude = 0 (= numero de gate), les valeurs sortent sur la sortie gauche ; - Si entrée chaude différente de 0 (+ ou -), les valeurs sortent sur la sortie droite. - Un bang sur l'entrée froide passe de la même façon qu'un nombre.
		
switch	Comportement	<ul style="list-style-type: none"> - Comportement identique à Max : permet de choisir entre plusieurs entrées. - Si consigne de porte (entrée chaude gauche) = 0, les valeurs ne passent pas. - Si consigne = 1, les valeurs passent par la 1^{ère} entrée ; - Pour "switch", si consigne porte <0 ou >=2, les valeurs passent par la 2nde entrée. - Pour "switch 3", si consigne porte <0 ou >=3, les valeurs passent par la 3^ee entrée.
		
	RESTRICTIONS	<ul style="list-style-type: none"> - Limite de 3 entrées maximum, seulement jusqu'à switch 3.
sel	Comportement	<ul style="list-style-type: none"> - Comportement identique à Max : envoie un bang sur une sortie de gauche quand l'entrée de gauche correspond au paramètre. - Un paramètre peut être modifié par l'entrée de droite. - Ce qui ne correspond à aucun paramètre sort à droite.
		
	RESTRICTIONS	<ul style="list-style-type: none"> - Seulement un ou deux paramètres autorisés. - "sel bang" n'est pas possible.
split	Comportement	<ul style="list-style-type: none"> - Comportement identique à Max : sépare un flux de données en envoyant à gauche les nombres compris entre les limites et à droite les autres.
		
	REMARQUES	<ul style="list-style-type: none"> - Des bornes inversées dans les paramètres par défaut (1er > 2nd) sont équivalentes aux bornes non inversées, par exemple "split 50 60" = "split 60 50". - Si les bornes sont inversées en cours de programme, toutes les données sortent à droite.
clip	Comportement	<ul style="list-style-type: none"> - Comportement identique à Max : contraint un nombre entre deux valeurs limites. - Les nombres inférieurs sortent égaux à la borne inférieure, les nombres plus grands sortent égaux à la borne supérieure.
		
onebang	Comportement	<ul style="list-style-type: none"> - Comportement identique à Max : envoie un bang quand il est activé puis bloque le flux tant que l'entrée de droite ne reçoit pas de bang. Un argument autre que 0 préactive l'objet.
		
	RESTRICTION	<ul style="list-style-type: none"> - N'accepte pas de message STOP donc pas d'arrêt possible
change	Comportement	<ul style="list-style-type: none"> - Comportement identique à Max : filtre les répétitions et ne transmet que le premier d'une série de nombres identiques sur la sortie de gauche. Il sort 1 sur la sortie du milieu si les données passent de 0 à un autre nombre et 1 sur la sortie de droite si les données passent à 0.
		
	RESTRICTION	<ul style="list-style-type: none"> - Pour l'instant les modes change 0- et change 0+ ne sont pas implémentés.

b) Abstractions spécifiques à la carte

rampe 	Comportement	<ul style="list-style-type: none"> - Abstraction spécifique à la carte. - Crée une gradation de nombres entre une valeur de départ et une valeur à atteindre. Avance d'un pas vers la valeur à atteindre à chaque bang sur l'entrée chaude. - Permet de compenser l'absence de l'objet line, inutilisable à cause de son besoin de messages à 2 paramètres.
Description		<ul style="list-style-type: none"> - 3 entrées décrites de gauche à droite : <ul style="list-style-type: none"> - entrée chaude : <ul style="list-style-type: none"> - si nombre, la sortie rejoint immédiatement cette valeur. Cependant cette valeur n'est pas conservée en mémoire en tant qu'objectif, elle devient un point de départ pour rejoindre l'objectif précédemment défini. - si bang, la sortie progresse en direction de l'objectif, ajout en + ou - du pas. - entrée milieu : l'envoi d'un nombre met à jour l'objectif à atteindre. Par défaut l'objectif est 0. Par exemple si l'objectif n'est pas renseigné, un nombre envoyé sur l'entrée chaude sort immédiatement, puis des bangs créent une rampe rejoignant 0. - entrée droite : mise à jour du pas de progression, de valeur minimale 1 ; si le nombre est négatif ou nul, le pas devient 1. Si le pas est supérieur au chemin qui reste à parcourir, rampe sort la valeur à atteindre. - 1 sortie : valeur courante de la rampe qui ne sort que s'il y a changement (notion de change intégrée). - L'external écrit pour max accepte un paramètre : le pas. - Simule à l'identique le fonctionnement de l'abstraction en mode externe.
zcounter 	Comportement	<ul style="list-style-type: none"> - Abstraction spécifique à la carte. - Crée un compteur entre 0 et un nombre. Le comptage se fait en boucle et peut être croissant, décroissant ou palindromique. - Permet de compenser l'absence de l'objet counter, inutilisable à cause du nombre d'entrées / sorties simultanées.
Description		<ul style="list-style-type: none"> - 3 entrées : <ul style="list-style-type: none"> - entrée gauche : informations de progression dans le compte si bang : incrémente ou décrémente le compteur ; si nombre : est transmis directement à la sortie ; un bang ensuite incrémente ou décrémente à partir de ce nombre. - si le nombre est supérieur au maximum spécifié, rien ne sort mais un bang ensuite fait repartir le compteur depuis le maximum. - entrée du milieu : maximum à atteindre ou d'où partir. Il doit être supérieur à 0, sinon il est considéré comme 0. - entrée droite : sens du comptage 0 : comptage croissant ; 1 : comptage décroissant ; 2 : comptage palindromique ; si nombre autre : considéré comme un 0. - Simule à l'identique le fonctionnement de l'abstraction en mode externe.
REMARQUE		Attention à l'ordre des entrées et des arguments et aux différences avec Max.
scale127	Comportement	<ul style="list-style-type: none"> - Objet spécial, - Simule à l'identique le fonctionnement de l'abstraction en mode externe. - Permet de calculer le barycentre pondéré entre deux valeurs (poids compris entre 0 et 127).
scale255	Comportement	<ul style="list-style-type: none"> - Objet spécial. - Simule à l'identique le fonctionnement de l'abstraction en mode externe. - Permet de calculer le barycentre pondéré entre deux valeurs (poids compris entre 0 et 255).
scale1023	Comportement	<ul style="list-style-type: none"> - Objet spécial. - Simule à l'identique le fonctionnement de l'abstraction en mode externe. - Permet de calculer le barycentre pondéré entre deux valeurs (poids compris entre 0 et 1023).
scale4095	Comportement	<ul style="list-style-type: none"> - Objet spécial. - Simule à l'identique le fonctionnement de l'abstraction en mode externe. - Permet de calculer le barycentre pondéré entre deux valeurs (poids compris entre 0 et 4095).

comparateur2 seuils	Comportement	- Abstraction spécifique à la carte. - Comparateur à hystérésis : envoie 0 ou 1 en fonction de deux seuils différents. Le nombre doit être supérieur au 2e seuil pour que l'objet envoie 1, puis doit descendre en dessous du premier seuil pour que l'objet bascule vers 0.
	Description	- 3 entrées : - entrée gauche : valeur à comparer si nombre : compare aux bornes. si bang : envoie le résultat précédent. - entrée du milieu : borne inférieure - entrée droite : borne supérieure Si la borne inférieure est supérieure à la borne supérieure, seul le nombre le plus grand (cette borne inférieure donc) est pris en compte. - Simule à l'identique le fonctionnement de l'abstraction en mode externe.

2 - Objets opérateurs

a) Opérateurs logiques

&	- Objet standard de Max, comportement identique, and.
 	- Objet standard de Max, comportement identique, or.

b) Opérateurs mathématiques

rotation a gauche <<
rotation a droite >>

+	- Objet standard de Max, comportement identique : addition.
-	- Objet standard de Max, comportement identique : soustraction.
*	- Objet standard de Max, comportement identique : multiplication.
/	- Objet standard de Max, comportement identique : division. - / 0 équivaut à / 1 et ressort le nombre à division.
%	- Objet standard de Max, comportement identique : modulo, donne le reste de la division.
!-	- Comportement identique à Max : soustraction avec les opérands inversés.
!/	- Comportement identique à Max : division avec les opérands inversés. - !/ 0 (0 sur l'entrée chaude) sort 0 comme résultat.
minimum	- Comportement identique à Max : sort le minimum des deux nombres sur les deux entrées.
maximum	- Comportement identique à Max : sort le maximum des deux nombres sur les deux entrées.
abs	- Comportement identique à Max : donne la valeur absolue d'un nombre.

3 - Objets de comparaison

Ce sont des objets de tests qui renvoient 0 ou 1.

a) Compérateurs logiques

&&	- Objet standard de Max, comportement identique, "et logique".
 	- Objet standard de Max, comportement identique, "ou logique".



b) Compérateurs mathématiques

>	- Comportement identique à Max : comparaison "strictement supérieur à".
<	- Comportement identique à Max : comparaison "strictement inférieur à".
<=	- Comportement identique à Max : comparaison "supérieur ou égal à".
>=	- Comportement identique à Max : comparaison "supérieur ou égal à".
=	- Comportement identique à Max : comparaison "égal à".
!=	- Comportement identique à Max : comparaison "différent de".

F - Objets permettant de gérer la pluralité des patches

closebang lancé avant le changement de patch

loadbang en début du nouveau patch

memo 	Comportement	<ul style="list-style-type: none">- Abstraction spécifique à la carte.- Permet de passer des paramètres entre différents patches, son comportement est identique à l'objet int pour les deux entrées de gauche.- Simulation par une abstraction Max avec présence obligatoire d'un patch maître, simulation par coll.
	Description	<ul style="list-style-type: none">- 3 entrées :<ul style="list-style-type: none">- entrée droite : numéro x de la mémoire concernée- entrée milieu :<ul style="list-style-type: none">mise à jour de la valeur de la mémoire sans provoquer la sortie- entrée chaude :<ul style="list-style-type: none">si nombre, mise à jour de la valeur de la mémoire avec sortie ;si bang, sortie du nombre stocké dans la mémoire x.- une sortie : contenu de la mémoire numéro x.- Au démarrage de la carte, les memo sont réinitialisés à 0.
	REMARQUE	<ul style="list-style-type: none">- Il y a 16 mémoires disponibles, numérotées de 0 à 15 ;- Si argument de memo > 15 : devient memo 15 ;- Si argument de memo < 0 : devient memo 0.
jump 	Comportement	<ul style="list-style-type: none">- objet implémenté, pas d'équivalent Max MSP- simulation par une abstraction max avec présence obligatoire d'un patch maître, simulation par scripting
	Description	<ul style="list-style-type: none">- 128 patches de 64 objets sont stockables et disponibles sur la carte- deux entrées, pas de sortie :<ul style="list-style-type: none">- entrée froide: mise à jour du numéro du patch à lancer- entrée chaude: lancement d'un autre patch avec mise à jour éventuelle du numéro du patch à lancer- objet permettant de changer dynamiquement de patch- la réception d'un nombre x ou d'un bang provoque<ul style="list-style-type: none">- le lancement du closebang éventuel du patch en cours,- arrêt brutal de tous les metro et delay lancés du patch en cours,- le chargement du patch "Px" en mémoire vive si "x" est le nombre reçu, (ou du patch "Py" si réception d'un bang et que "y" est le numéro en mémoire)- lancement du loadbang du nouveau patch en mémoire.
	RESTRICTIONS	à vos risques et périls, il est de la responsabilité de l'utilisateur d'avoir pensé à charger les patches

G - Chargement du programme dans la carte

H - Classement alphabétique

1 - Abstractions spécifiques à la carte

Ces abstractions remplacent des objets courants qui ne sont pas implémentables directement dans la carte.

ACCESTABLE	QUATRE_CTLIN
ANA_IN_12BITS et ANA_IN_7BITS	QUATRE_CTLOUT
COMPARATEUR2SEUILS	QUATRE_NOTEIN
ECRIT_TABLE	QUATRE_NOTEOUT
INFOMIDI	quatredoublectout
JUMP	RAMPE
MEMO	RELAJ_OUT
MIDI_DMX	SCALE1023
MINIHF	SCALE127
NUM_IN	SCALE255
PAIRÉCTL	SCALE4095
PWM_OUT	SERVO_OUT
	ZCOUNTER

2 - Objets standards de Max

-	bangbang	message
!=	bendout	metro
!/	change	minimum
!-	clip	notein
*	closebang	noteout
/	ctlin	onebang
&	ctlout	pgmin
&&	delay	pgmout
%	gate	sel
+	Gswitch	split
<	Gswitch2	switch
<<	int ou i	toggle
<=	loadbang	trigger
==	maximum	
>		
>=		
>>		
abs		

V - Précautions d'emploi

Veuillez lire attentivement les consignes de sécurité et les conseils d'utilisation suivants. Vous minimiserez ainsi les risques d'accident et augmenterez la durée de vie des appareils.

Nos produits sont vendus en temps que parties destinées à être intégrées dans des installations ou à être utilisées en démonstration. Nous considérons que ceux qui les manipulent ont le niveau de compétence requis et appliquent toutes les précautions voulues pour le bon fonctionnement du système. Interface-Z se dégage de toute responsabilité concernant un quelconque dommage ou accident causé par une mauvaise utilisation de ses produits. Il est de la responsabilité de l'utilisateur de s'assurer que toute installation utilisant ces produits soit conforme aux normes de sécurité en vigueur et de compatibilité électromagnétique.

Interface-Z se décharge également de toute responsabilité concernant l'usure du matériel et de tout problème mécanique ou électrique causé une utilisation inadéquate du matériel. Par exemple, les modules sortis de leur boîtier ne sont pas garantis contre les problèmes électriques dus à des court-circuits en cas de mise en contact avec une surface métallique. Il est évident que des montages sans boîtier doivent être utilisés avec précaution. Les protéger leur assure une durée de fonctionnement plus élevée.

Interface-Z décline toute responsabilité pour tous dommages causés dans les conditions suivantes et ne garantit pas les montages lorsque les précautions indiquées dans chaque cas ne sont pas respectées :

- Sortie du boîtier, **fixation inappropriée des cartes.**

Si une carte est sortie de son boîtier ou que le boîtier est changé, précisons que les cartes comportent des emplacements (dans les coins) prévus pour une fixation par vis ou petits boulons, avec des rondelles isolantes. La carte peut aussi être tenue par des adhésifs fixés aux mêmes emplacements. Quelle que soit la méthode de fixation choisie, il ne faut pas que quoi que ce soit de **métallique** ou de **conducteur** entre en contact avec le circuit électronique ou avec les composants soudés. Il est donc recommandé de ne mettre de vis de fixation qu'aux endroits prévus à cet effet.

- **Maniement contraire à l'utilisation normale des appareils.**

Comme pour tout circuit imprimé, il ne faut pas provoquer de court-circuit sur les cartes, donc :

- ne jamais poser une carte hors boîtier sur une **surface conductrice** (objet métallique, surface mouillée, etc), cela pourrait l'endommager irréversiblement. Rien ne doit interférer avec les pistes ou avec les picots soudés ;
- éviter les décharges **électrostatiques** (toucher une surface métallique reliée à la terre, pour se « décharger » avant de manipuler la carte, surtout si l'on se sent « électrique ») ;
- de même, ne pas mettre de carte en contact avec un écran ou tout autre objet chargé d'électricité statique. Hors les dommages possibles occasionnés au module, cela pourrait provoquer des parasites et interférer avec le fonctionnement normal des modules ;
- éviter tout contact avec des éléments de masse électrique, par exemple tuyaux, radiateurs, cuisinières et réfrigérateurs.
- ne pas avaler, mâcher ou mordre.
- en ce qui concerne les boîtiers ou les dalles en bois, ne pas les exposer à des flammes, des gaz ou des liquides inflammables, des mégots allumés, ou quoi que ce soit susceptible de les endommager par le feu. Ne pas les stocker en plein soleil ou à l'humidité, pour une meilleure conservation.

- **Ne pas utiliser une carte en contact avec la peau**

Ne pas toucher les composants ou le circuit imprimé d'une carte ou d'un capteur branché, cela peut d'ailleurs interférer avec son fonctionnement et provoquer des résultats non souhaités. Ne pas utiliser de carte non protégée sur la peau, le corps, le visage, cela risque de provoquer des égratignures ou des piqures.

- **Non respect des consignes de sécurité.**

- Ne pas exposer ses oreilles aux ultrasons ;
- Ne pas exposer ses cheveux, ses doigts, son nez aux moteurs ou à ce qui est fixé dessus (même s'ils tournent lentement) ;
- Ne pas toucher une lampe ou ampoule allumée, cela peut brûler.

- **Mauvais entretien.**

- Les modules ne doivent pas être exposés à l'humidité, à la pluie, à des substances corrosives, à la chaleur, à la flamme, à des liquides ou gaz inflammables. Ils ne doivent pas être ouverts avec des objets métalliques, être mouillés ou écrasés.
- Les câbles et les fils doivent être protégés de la chaleur et des objets coupants et disposés de façon à ce qu'ils ne soient pas tirés.
- Ne pas soulever ou transporter les modules en les tenant par les câbles, surtout s'ils sont branchés.
- Vérifier avant l'utilisation que les modules sont en bon état (non fendus, non mouillés, etc).
- Nettoyer immédiatement en cas d'exposition à des liquides (boue, encre, alcool, nourriture, etc).
- Débrancher les appareils après utilisation.
- Les modules et les rallonges ne sont pas prévus pour une utilisation en extérieur. Dans le cas d'une installation en extérieur, il est de la responsabilité de l'utilisateur de s'assurer que son matériel est convenablement protégé (contre les intempéries, les animaux, les déprédations, etc...).

- **Mauvaise alimentation des cartes.**

Ne pas inverser les fils de masse et les fils positifs, si la documentation ne précise pas qu'il n'y a pas de polarité à respecter. Ne pas appliquer d'alimentation ne correspondant pas aux spécifications décrites dans la documentation. Une erreur peut détruire la carte.

Ne pas débrancher la prise en tirant sur le fil.

Ne pas forcer le fonctionnement des appareils : si le fonctionnement est inhabituel, débrancher immédiatement.

Ne pas débrancher les actionneurs en arrachant les fils.

- **Réparation**

Les réparations ou modifications, s'il y a lieu, ne doivent être effectuées que par un électronicien ayant la compétence voulue.

- **Limites des capteurs et des actionneurs**

Les capteurs FSR par exemple sont fragiles et ne supportent pas d'être écrasés au-delà de la limite prévue de 10 kilogrammes. Il ne faut donc ni marcher dessus ni les placer sous des objets lourds.

Les moteurs sont prévus pour une charge maximale précise et ne doivent pas être forcés.

- **Attention aux enfants :**

Ne pas les laisser manipuler le 220 Volts.

Les surveiller en permanence s'ils manipulent de petits capteurs, des actionneurs, des interfaces.

Ne pas les laisser manipuler des moteurs pas à pas (dont la connectique est complexe), des lampes halogènes ou des ampoules à incandescence (qui chauffent).

Utiliser avec les enfants des alimentations électriques par piles ou batteries. Ne pas utiliser de bloc secteur ou d'alimentation branchée sur le secteur.

- Attention aux interférences possibles avec l'appareillage et l'électronique **médicaux**.

- Ne **jamais** utiliser ce matériel dans le cas où la vie ou la santé d'une personne dépendrait de ce matériel.