

# 03 L'Intelligence Artificielle

On entend beaucoup parler de l'Intelligence Artificielle depuis quelques temps grâce aux media qui relaient les progrès réalisés dans ce domaine. Et on a tous une idée plus ou moins précise sur la question. Nous allons voir dans ce chapitre que l'IA peut être une aide pour programmer et un excellent professeur pour progresser.

## 3.1 / QU'EST-CE QUE L'IA ? CE QU'ELLE PEUT FAIRE ET NE PEUT PAS FAIRE

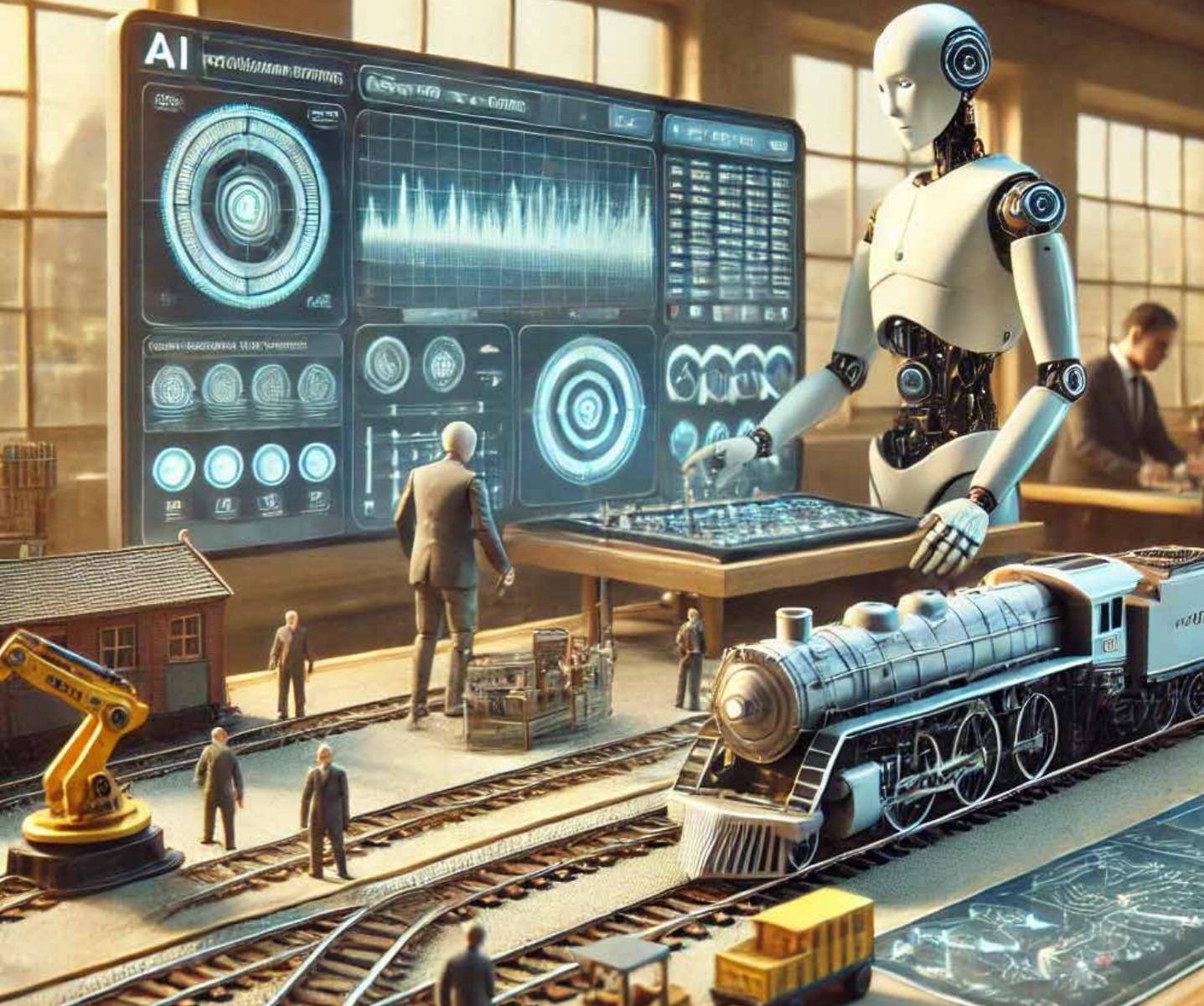
Qu'est-ce que l'IA ? Voilà un vaste sujet qui peut déboucher sur un vaste débat ! Il y a ceux qui sont farouchement pour et ceux qui sont farouchement contre, mais il n'est pas certain que ces deux catégories sachent exactement ce qu'est une IA, comment elle fonctionne, ce qu'on peut en attendre. Je me situe entre ces deux extrêmes, peut-être pas exactement au centre, penchant sans doute plutôt pour. En effet, je ne considère pas l'IA comme la panacée qu'on nous annonce mais plutôt comme **un nouvel outil** qu'il vaut mieux savoir utiliser. Car, n'en doutons pas, les IA vont se développer et finiront par s'imposer ; la preuve, elles sont déjà parmi nous. Je ne vais donc pas débattre des bienfaits et des méfaits des IA pour l'humanité et mon rôle se cantonnera à décrire comment il faut faire pour que l'IA soit une aide.

J'ai acquis en quelques mois de travail acharné, une certaine expertise sur l'utilisation des IA et j'ai écrit un premier article sur l'apport des IA dans le

modélisme ferroviaire dans Loco-Revue qui, une fois de plus, a été la première revue de modélisme ferroviaire en France à aborder ce sujet de haute technologie (LR 928 de novembre 2024). Dans cet article, **j'ai démontré qu'une IA peut produire du code pour nos cartes Arduino et que le programme peut fonctionner ... ou pas !** Et j'ai aussi démontré que si l'IA ne sait pas faire, elle peut apprendre de notre part et finir par y arriver. Tant que cela fonctionne, l'IA peut être une aide pour un débutant qui n'a plus besoin d'apprendre à coder. Oui, mais quand cela ne fonctionne pas, ce même débutant est bien incapable d'en comprendre la raison. Doit-il renoncer pour autant ?

Fort de ces premières expériences, j'ai persévééré, ce qui m'a permis de mieux comprendre comment utiliser les IA. L'expérience était tentante de concevoir un petit réseau miniature (un réseau de débutant), géré par une carte Arduino dont les programmes avaient été écrits par l'IA

Quand l'IA jouera au train !  
(Image réalisée par une Intelligence Artificielle)



ChatGPT à plus de 95%. Ce réseau, appelé **EX\_MACHINA**, a été décrit dans Loco-Revue de juillet 2025, la carte Arduino gérant la circulation de deux trains simultanément ou l'un après l'autre selon les conditions demandées, ainsi que la signalisation lumineuse. Si une IA a été capable de comprendre les problèmes à résoudre, de proposer des solutions et d'écrire les programmes en conséquence, alors c'est la preuve que **les IA peuvent nous aider dans notre hobby**.

Pour autant, la réussite de l'IA à écrire les programmes d'**EX\_MACHINA**, n'est pas due au fait qu'elle est supérieurement intelligente, mais au fait qu'elle a été **solicitude en lui fixant un cadre strict et des consignes rigoureuses**. Elle n'a pas conçu le circuit, ni le décor (d'ailleurs je n'en avais pas mis), ni le but à atteindre. Elle n'a fait que produire du code pour Arduino et trouver des algorithmes qui fonctionnent, et ce n'est déjà pas si mal.

## 3.2 / LES DIFFÉRENTES IA ET LEURS SPÉCIALISATIONS

Il n'existe pas qu'une seule IA mais bien plusieurs, développées par différentes entreprises, dans différents pays car **les IA représentent un enjeu majeur**. Certaines IA sont spécialisées dans un domaine particulier : moteur de recherche, traduction de texte, élaboration d'images ou de vidéos, création musicale, reconnaissance vocale ou faciale, reconnaissance et manipulation d'objets, conduite assistée, etc. Par exemple, FluxPro est une IA spécialisée dans la génération d'images alors que Novelai est spécialisée en littérature et Sora en production vidéo. **On ne choisit donc pas une IA parce que la télévision ou les réseaux sociaux en parlent, mais parce que cette IA est spécialisée dans ce qu'on veut faire**. Aux États-Unis, les IA sont déjà utilisées pour produire des fonds de décor sur les réseaux miniatures.

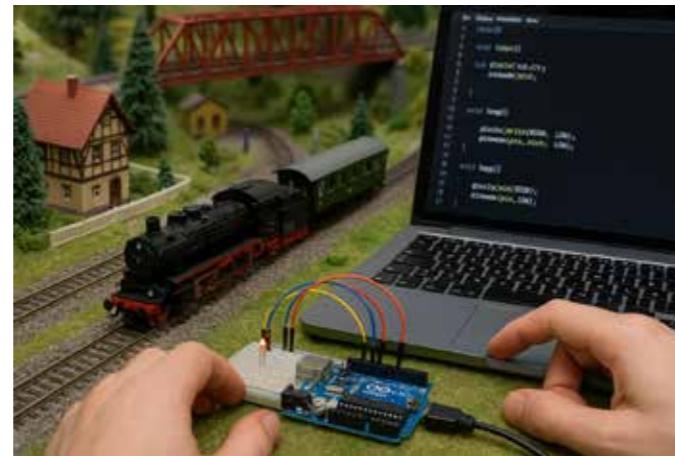


Figure 3-1

image générée par ChatGPT

La **figure 3-1** montre une image générée par une IA pour illustrer le train miniature associé à la programmation des cartes Arduino par l'intelligence artificielle.

## 3.3 / GOOGLE TRADUCTION : UNE IA QUI NOUS AIDE VRAIMENT

Google traduction est une IA qui fait de la **traduction de textes d'une langue à une autre**. Comme je vous l'ai déjà fait remarquer, il lui arrive de faire quelques erreurs, mais souvent les traductions sont faites sans que l'IA ne connaisse le contexte. Malgré ces petites erreurs, Google traduction va considérablement aider ceux qui ne maîtrisent pas la langue de Shakespeare utilisée dans toutes les notices de composants. Comme ces notices sont fournies sous forme électronique (souvent des fichiers PDF), il est très facile de faire un copier-coller d'une phrase dans Google traduction et avoir ainsi la traduction.

La **figure 3-2** donne un exemple de traduction du français vers l'anglais, du paragraphe précédent.

Ce n'est pas si mal, même si j'aurais traduit « notices de composants » par « datasheets » au lieu de « component instructions », mais encore une fois, le contexte n'était pas évident.

Pour les documents plus élaborés, il suffit d'entrer le document complet avec un glisser-déposer (**figure 3-3**). J'ai essayé avec la datasheet de la carte Giga R1 WiFi (document de 4 Mo contenant 22 pages) et le résultat pouvait être affiché ou bien téléchargé (**figure 3-4**). Même si je n'ai pas tout contrôlé, le résultat semblait probant.

Avec cette intelligence artificielle, vous avez un outil qui peut vous aider vraiment à comprendre toutes sortes de documents, en anglais ou autres langues.

## 3.4 / LES IA GÉNÉRATIVES

Les IA génératives sont les plus connues du public : **ChatGPT, DeepSeek, le Chat de Mistral**

(IA française) pour ne citer que quelques exemples. Issues de **LLM (Large Language Model ou**

**Modèle Linguistique Étendu**), elles sont capables de comprendre un texte et d'y répondre, dans plusieurs langues. On peut donc converser avec elles comme on le ferait avec un interlocuteur. Elles sont programmées pour simuler des réactions humaines, et grâce au « Deep Learning », leur domaine de connaissances est étendu. Elles ne connaissent pas tout pour autant et il n'y a pas (encore) d'IA spécialisée en modélisme ferroviaire !

Comme ces IA sont spécialisées dans le langage, **elles connaissent les langages de programmation comme le C/C++, HTML, Python, CSS, etc**. Elles sont donc capables de générer du code qui peut fonctionner ou pas



Figure 3.1

ChatGPT, DeepSeek, Perplexity

comme je l'ai déjà dit. Les chances de réussite d'un programme peuvent être améliorées **si on donne des instructions précises à l'IA** : c'est ce qu'on appelle **rédiger le prompt, ou dans le langage des IA « prompter »**.

Comme il y a plusieurs IA pour générer du code, on peut les mettre en concurrence et voir laquelle donne le code le plus compact et le plus efficace. Il suffit d'ouvrir un compte sur les IA qu'on veut utiliser : ceux-ci sont payants mais une formule de base, limitée à quelques interventions par jour, est gratuite et suffisante pour découvrir ce nouvel outil. La **figure 3-5** montre quelques logos d'Intelligence Artificielle.

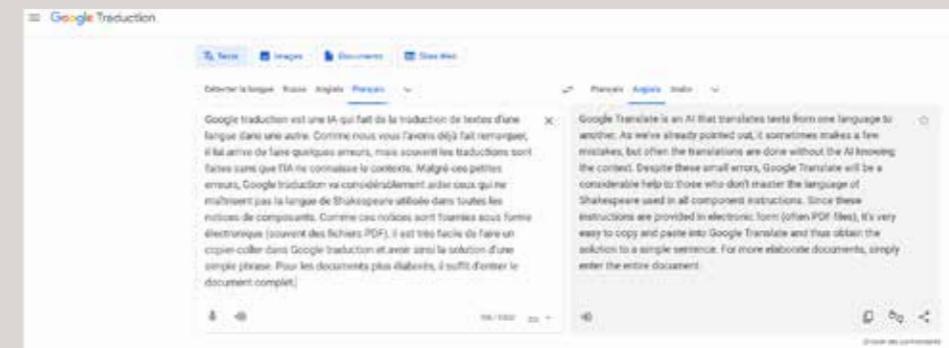


Figure 3-2

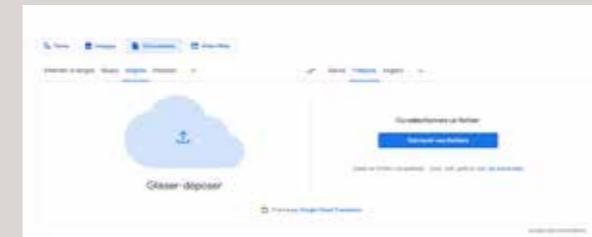


Figure 3-3

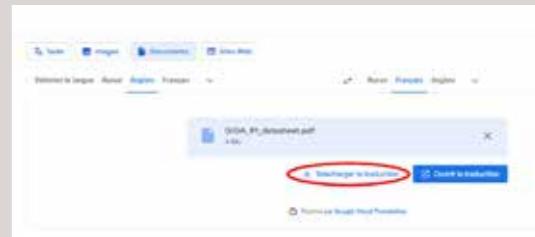


Figure 3-4

### 3.5 / COMMENT RÉDIGER LE PROMPT ?

Pour obtenir un résultat cohérent, la rédaction du prompt est très importante. Nous allons le découvrir ensemble sur l'exemple donné par la figure 3-6, qui représente un montage de motorisation d'une aiguille par servomoteur. Un poussoir permet de lancer la manœuvre de l'aiguille et deux LED permettent de connaître sa position : LED verte si aiguille droite et LED rouge si déviée. Le servomoteur se déplace entre deux positions extrêmes ( $70^\circ$  et  $110^\circ$ ) correspondant à aiguille droite ou déviée. Nous allons voir comment demander à une IA d'écrire le programme.

Nous devons expliquer le but à atteindre : obtenir un mouvement lent du servomoteur durant deux secondes, pour passer de la position  $70$  à  $110$  et réciproquement chaque fois

qu'on appuie sur le poussoir. Il faut dire également à l'IA comment sont connectés les différents composants : servomoteur, poussoir, LED verte et rouge. Voici le prompt :

**Écrire un programme pour carte Arduino Uno pour obtenir un mouvement lent d'un servomoteur durant deux secondes, pour passer de la position  $70^\circ$  à  $110^\circ$  et réciproquement chaque fois qu'on appuie sur un poussoir. Le servomoteur est connecté sur la broche 10, le poussoir sur la broche 2. Une LED verte connectée en 5 doit être allumée uniquement si la position du servomoteur est  $70^\circ$ . Une LED rouge connectée en 7 doit être allumée uniquement si la position du servomoteur est  $110^\circ$ .**

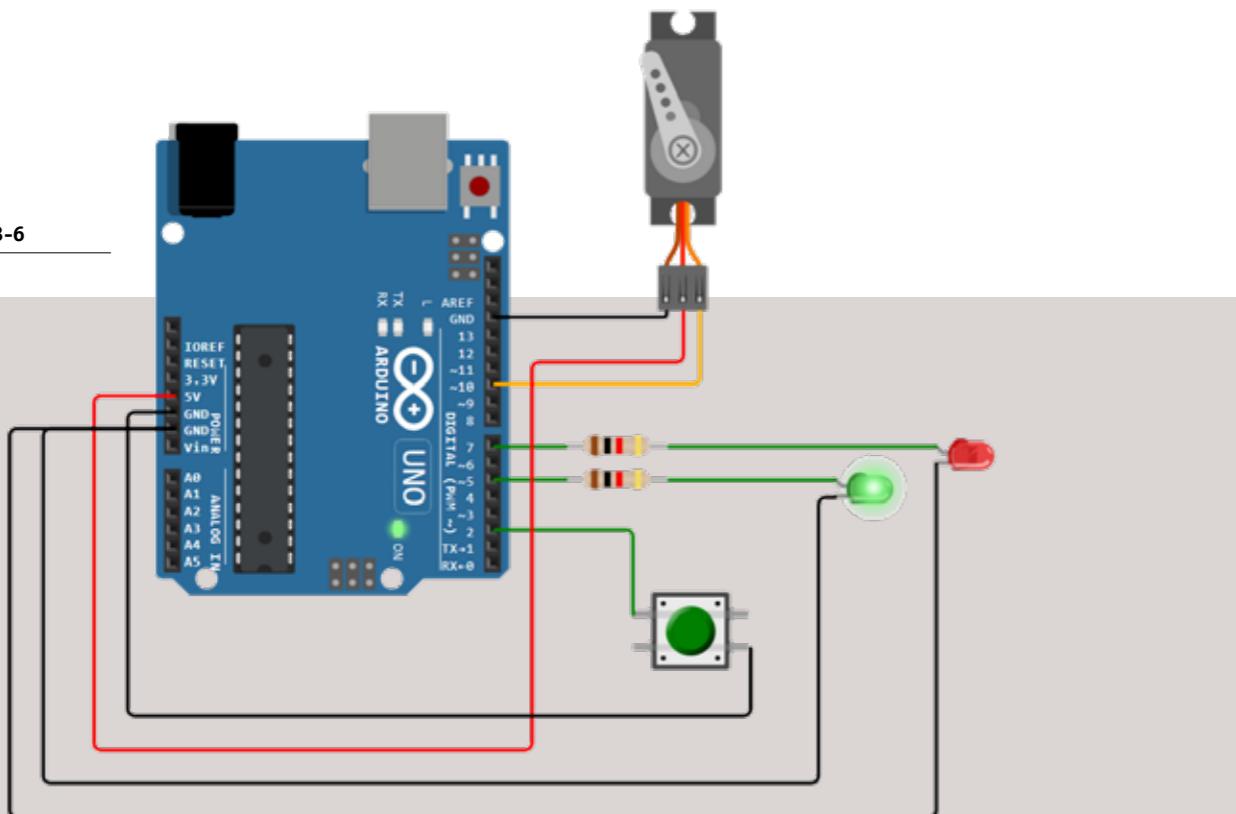


Figure 3-6

L'IA résume la demande, puis donne son programme. Elle donne aussi quelques conseils de branchement des composants. Une fois le programme importé sur le montage, je constate que le fonctionnement est identique au montage original. Voici le lien vers le programme proposé : <https://wokwi.com/projects/427841977627908097>.

J'en profite pour lui demander une modification à laquelle je n'avais pas pensé moi-même :

**Peux-tu corriger ce programme pour que les LED soient toutes les deux éteintes durant le mouvement du servomoteur et allumées ensuite pour indiquer la position du servomoteur ?**

Le programme est à nouveau opérationnel (<https://wokwi.com/projects/427842483370860545>) du premier coup. Je modifie une fois de plus la demande pour reproduire cette fois le fonctionnement d'une barrière de PN (avec deux LED rouges) :

**Peux-tu modifier ce programme pour que le servomoteur aille de la position  $90^\circ$  à  $180^\circ$  lors du premier appui, et faire clignoter les deux LED pendant et après le mouvement. Un deuxième appui doit éteindre les LED immédiatement, puis faire le mouvement du servomoteur de la position  $180^\circ$  à  $90^\circ$ . Et ainsi de suite.**

Cette fois, le programme ne fonctionne pas tout à fait comme il devrait. Je le lui indique en demandant la correction (<https://wokwi.com/projects/427843196418382849>) :

**Peux-tu corriger ce programme car lors du premier appui sur le poussoir, les LED ne clignotent pas dès le début du mouvement du servomoteur mais seulement quand il arrive en position finale. Le mouvement du servomoteur doit se faire en 5 secondes.**

L'IA commence par analyser son erreur (figure 3-7) et cette fois, son programme fonctionne comme prévu (<https://wokwi.com/projects/427843639794124801>).

Parfait, merci pour la précision !

Problème actuel : Le clignotement commence après le mouvement, car la boucle `for`, qui contrôle le servomoteur, bloque l'exécution du reste du code notamment la gestion du clignotement via `millis()`.

Solution :

On va rendre le mouvement non bloquant, c'est-à-dire qu'on va le découper dans le `loop()` en plusieurs petits pas, synchronisés avec `millis()`, tout en laissant le clignotement actif en parallèle.

Figure 3-7

Comme vous le voyez, dans chacun de mes prompts j'ai essayé d'être le plus précis possible sur ma demande et c'est sans doute pourquoi cela a fonctionné assez bien. Grâce à l'IA nous sommes passés d'une commande d'aiguille à une commande de barrière de PN. Cette démonstration nous a permis de voir que les IA peuvent aussi se tromper, mais elles peuvent corriger le tir si on leur explique ce qui ne fonctionne pas. Ce que nous venons de faire est à la portée d'un modéliste qui ne connaît pas la programmation.

L'erreur de l'IA pourrait devenir un problème sur un cas plus compliqué ; il faut toujours décomposer un cas compliqué en une suite de cas plus simples et on finit par y arriver.

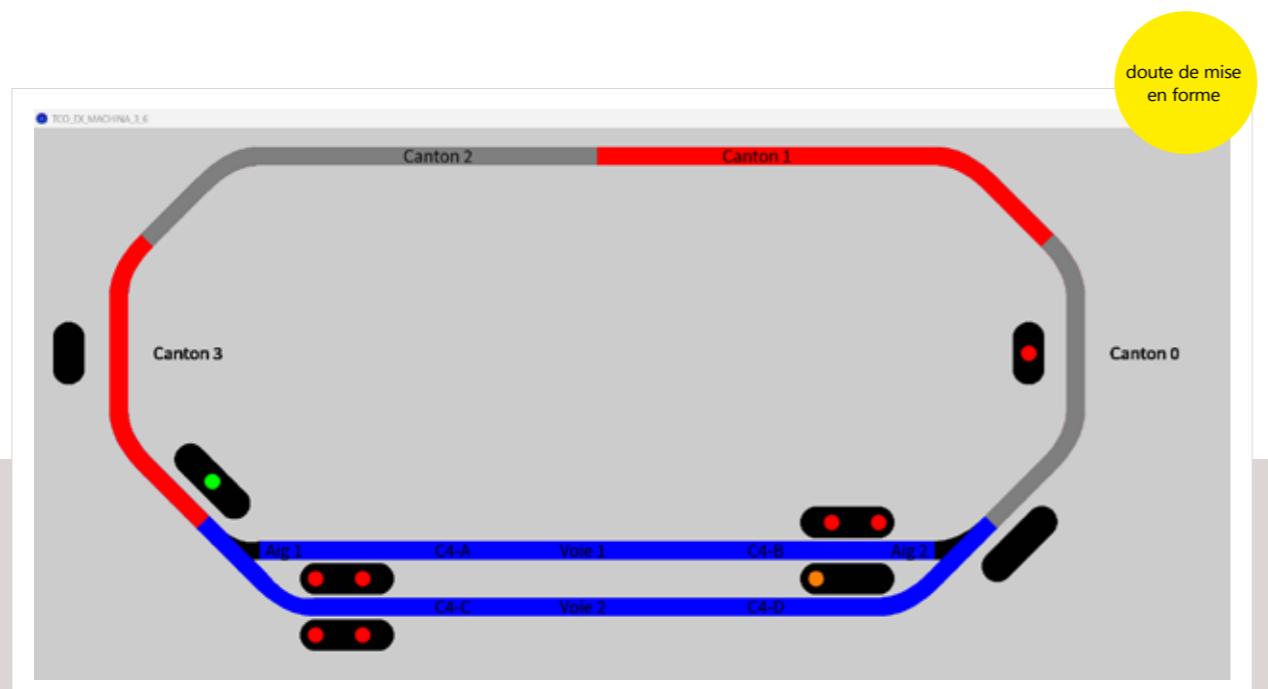
D'ailleurs, pour travailler sur un projet complexe, il est indispensable d'agir par étapes et de demander à l'IA de créer des fonctions qui peuvent ensuite être jointes au programme principal. Et pour que tout s'articule bien, il peut être nécessaire d'imposer à l'IA le nom et le type de certaines variables dans nos prompts. C'est comme cela que j'ai mis au point le réseau EX\_MACHINA (décris dans LR936 de juillet 2025). Tout cela n'est pas si compliqué, et travailler avec une IA permet aussi d'apprendre très vite car il suffit d'analyser son programme pour découvrir certaines règles d'écriture du langage C (ou d'un autre langage).

## 3.6 / COMMENT CONTRÔLER LE LOGICIEL DÉLIVRÉ ?

Il suffit de faire un copier-coller du programme de l'IA dans l'IDE d'Arduino pour vérifier qu'il compile sans problème pour la carte que vous utilisez. Honnêtement, je ne crois pas que l'IA puisse faire une erreur qui empêcherait la compilation dans la majorité des cas, mais cela m'est tout de même arrivé lorsque j'ai voulu la faire travailler sur une bibliothèque très peu connue, qui nécessitait d'instancier un objet d'une certaine façon.

### Le fait qu'un programme compile ne signifie pas qu'il fonctionne comme on voudrait.

C'est là que les simulateurs d'Arduino sont intéressants car si le montage peut être réalisé avec les bons composants, il suffit d'importer le programme de l'IA dans le simulateur et de passer en mode simulation. S'il manque des composants au simulateur pour reproduire le montage, alors il faut tester le programme sur le prototype et programmer la carte chaque fois qu'un nouveau programme est fourni par l'IA.



TCO\_dEX\_MACHINA, réseau conçu par une Intelligence Artificielle

## 3.7 / QUAND LES IA HALLUCINENT

Les IA ont tendance à vouloir nous donner une réponse systématiquement, même lorsqu'elles disposent d'assez peu de données sur un sujet ou bien si on leur demande un travail dans un cadre pour lequel elles ne sont pas conçues. Ces réponses sont alors farfelues mais pourraient passer pour crédibles car les IA vont les présenter d'une façon appropriée : **on dit alors que les IA hallucinent** et c'est pourquoi il est nécessaire de conserver un esprit critique sur ce qu'elles délivrent. Dans le domaine de la production de code, il est facile de vérifier le programme délivré, comme l'explique le paragraphe précédent. Dans d'autres domaines, on peut se rendre compte que les IA sont incapables de produire des résultats exploitables.

J'ai essayé par exemple d'utiliser ChatGPT pour réaliser un circuit imprimé (PCB) : l'IA devait produire les fichiers Gerber en utilisant KiCad 9.0 (voir chapitre 1). L'IA avait très bien compris la demande et devait nous fournir l'ensemble des fichiers après un certain délai. Ne voyant rien arriver au bout d'un quart d'heure, j'ai relancé l'IA qui m'a soutenu qu'elle travaillait à élaborer

les fichiers et qu'il n'était pas nécessaire de la relancer. J'ai pris mon mal en patience, mais après une heure d'attente, je l'ai à nouveau sollicitée pour finalement obtenir une suite de fichiers vides et donc inexploitables. Lorsque j'ai fait la remarque à l'IA, elle a reconnu qu'elle ne pouvait pas générer ces fichiers à partir du logiciel KiCad, ce qu'elle aurait pu me dire dès le début.

C'est la même chose dans le domaine de la génération d'images ou de vidéos qui peuvent être fournies avec des artefacts (personnages irréels avec six doigts, animaux dont la tête ne correspond pas au corps, etc.). Une chose est sûre : les IA n'en sont qu'au début et vont forcément s'améliorer puisque d'énormes programmes de recherche sont lancés dans tous les pays avec de forts financements (17 milliards d'euros en France, décision prise à « Choose France » à Versailles en mai 2025). Il est à parier qu'un jour, les IA seront capables de dessiner nos PCB et commander leurs fabrications directement auprès d'une entreprise ; ce sera toujours à nous de payer la facture, sauf si elles nous demandent notre numéro de carte bancaire !

### RÉSUMÉ DU CHAPITRE 3

L'Intelligence Artificielle n'est sans doute pas la solution à tout mais c'est une aide indéniable. Elle peut, comme je l'ai montré, écrire des programmes à notre place. Mais surtout, elle peut nous aider à progresser à condition d'avoir envie de comprendre ce qu'elle délivre. Il suffit de regarder comment elle s'y prend et il n'y a certainement pas de professeur plus patient pour nous enseigner la programmation. Cet outil s'imposera de plus en plus dans l'avenir et c'est maintenant qu'il faut apprendre à le maîtriser !