

créez un neurone d'intégration- et-tir avec fuite



Intelligence artificielle sans logiciel

Clemens Valens (Elektor)

Les réseaux neuronaux artificiels constituent la base de l'intelligence artificielle. Ils sont généralement implémentés dans des logiciels, mais leur efficacité énergétique est relativement faible. Les réseaux neuronaux biologiques sont beaucoup plus performants. Dans cet article, nous présentons un circuit qui simule un neurone biologique, vous permettant de construire et d'expérimenter votre propre réseau neuronal.

Dessiner une moustache sur une affiche dans le métro est rarement perçu comme un acte d'intelligence. En revanche, lorsqu'un filtre Instagram ajoute des oreilles de chien à votre image cela est souvent perçu comme une prouesse technologique. Cela souligne combien la perception de l'intelligence, qu'elle soit artificielle ou non, est empreinte de subjectivité.

D'autre part, ce qui n'est pas subjectif, c'est l'énorme quantité d'énergie requise pour accomplir des tâches jugées intelligentes. Des millions, voire des milliards de transistors, sont sollicités pour créer des oreilles de chien virtuelles, consommant ainsi une quantité considérable d'énergie. En comparaison, chaque être vivant, qu'il s'agisse d'une huître ou même de brins d'herbe, réalise des opérations bien plus complexes que les systèmes d'IA les plus avancés, et ce, avec une efficacité énergétique bien supérieure.

Pour les animaux, quelques neurones intelligemment interconnectés et un peu de nourriture suffisent à accomplir des merveilles.

Réseaux neuronaux

Les neurones sont généralement considérés comme les éléments fondamentaux du cerveau, et donc de l'intelligence. Lorsqu'ils sont interconnectés, ils forment un réseau neuronal (**figure 1**). Les réseaux neuronaux artificiels (RNA) ont été efficacement appliqués à la reconnaissance vocale, à l'analyse d'images et au contrôle adaptatif dans le cadre de l'apprentissage automatique et de l'intelligence artificielle. [1]

Au sein d'un réseau neuronal, qu'il soit artificiel ou biologique, un neurone reçoit des signaux de neurones connectés. Il traite ces signaux, et s'il y a un résultat, il le transmet à d'autres neurones. Chaque signal d'entrée est pondéré. Les poids sont ajustables, permettant ainsi au neurone de s'adapter et au réseau de d'apprendre. Le résultat (instantané) est déterminé par une fonction non linéaire appliquée à la somme des entrées.

En général, les neurones artificiels fonctionnent avec des données numériques car ils sont principalement implémentés dans des logiciels. Les neurones biologiques utilisent des signaux électriques et chimiques. L'avantage des neurones logiciels est leur facilité de production en grande quantité et la simplicité avec laquelle un réseau peut être modifié. Cependant, il existe une tendance croissante à implémenter des neurones directement sur du matériel, en utilisant des

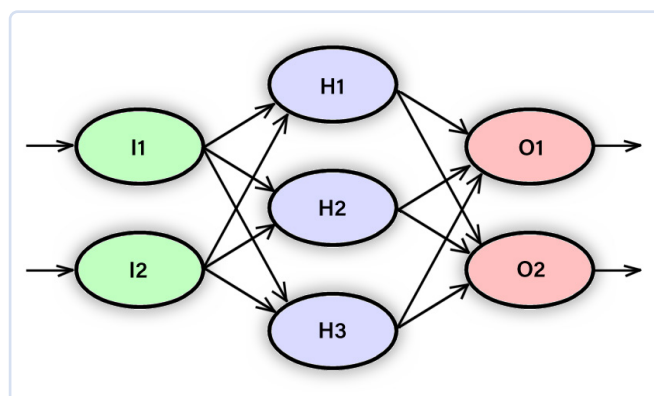


Figure 1. Un réseau neuronal artificiel simple avec deux neurones d'entrée (I1 & I2), une couche cachée de trois neurones (H1-3), et deux neurones de sortie (O1 & O2). Une drosophile possède environ 100 000 neurones.

technologies telles que la logique programmable des FPGA. Ces neurones sont de petits circuits construits avec des transistors. Ces neurones artificiels tentent d'imiter les signaux électriques utilisés par les neurones biologiques. Le neurone présenté dans cet article est un tel circuit.

Les neurones biologiques ne génèrent pas des niveaux de signal continus, mais produisent plutôt des impulsions courtes (**figure 2**). La fréquence et la durée des impulsions sont variables, et il est supposé que l'information est véhiculée dans le temps séparant deux impulsions et dans la largeur de l'impulsion. Ce type de communication pourrait être assimilé à une modulation de position d'impulsion (PPM). Les neurones qui produisent de telles impulsions sont appelés neurones à impulsions (SN) ; un réseau composé de tels neurones est appelé réseau neuronal à impulsions (SNN).

Neurone à impulsions

Un neurone à impulsions peut être assimilé à un pluviomètre à auget basculant qui mesure les précipitations (**figure 3**). Les impulsions entrantes (les gouttes de pluie) remplissent le seau, et lorsque le seau est plein, il bascule et se vide, générant ainsi une impulsion. À

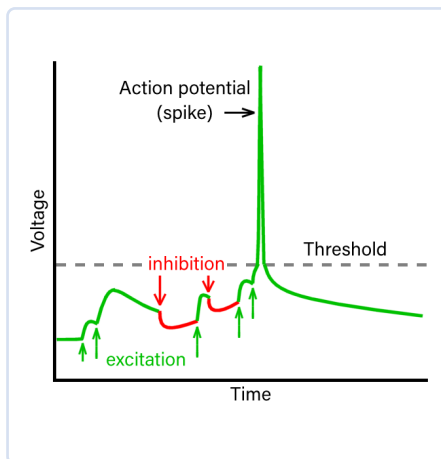


Figure 2. Les signaux d'un neurone à pointes (source : Alan Woodruff / QBI).

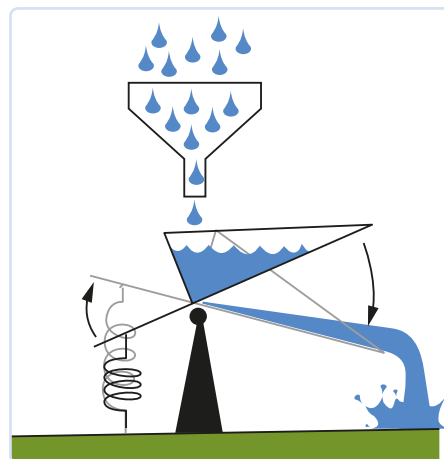


Figure 3. Un pluviomètre à auget basculant.

l'instar du pluviomètre, le SN ajoute des poids aux gouttes de pluie entrantes et peut contrôler la largeur de l'impulsion émise. En outre, le SN permet au seau de fuir, ce qui ajoute une dimension temporelle. Si les impulsions d'entrée arrivent trop lentement, le seau ne se remplira jamais et le SN ne se déclenchera jamais. En d'autres termes, le SN a la capacité d'oublier les stimuli précédents.

L'équivalent électronique du pluviomètre à auget basculant est l'oscillateur de relaxa-

tion. Dans un tel circuit, un condensateur est chargé et lorsqu'un certain seuil est atteint, un transistor ou un autre interrupteur contrôlé électroniquement décharge le condensateur, ce qui produit une impulsion. Le transistor unijonction (programmable) (UJT ou PUT) serait parfait pour construire un SN. Cependant, ces dispositifs sont aujourd'hui difficiles à obtenir à des prix raisonnables. Heureusement, il est possible de fabriquer un PUT en utilisant deux transistors bipolaires, une solution. Il est

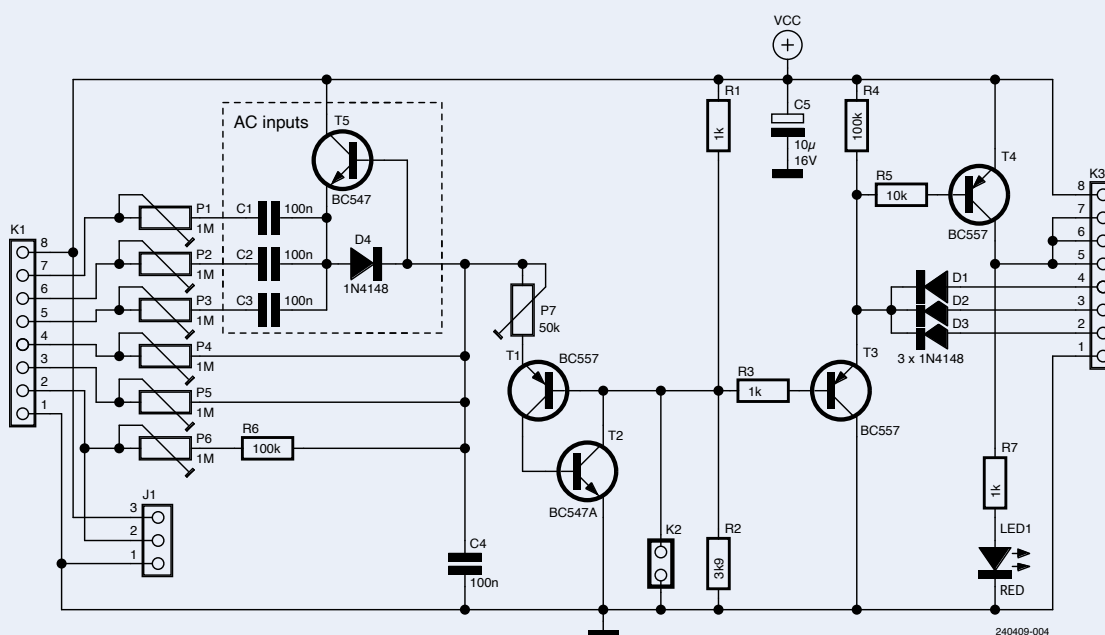


Figure 4. Ce neurone leaky integrate-and-fire spiking est très flexible et configurable. Pas aussi puissant que le Loihi d'Intel ou le TrueNorth d'IBM, il constitue un point de départ amusant.

à noter que pour implémenter un SN(N) dans un FPGA (ou dans du silicium, comme le Loihi d'Intel ou le TrueNorth d'IBM), les MOSFET restent les commutateurs de choix.

Circuit

Le circuit de notre neurone à impulsion est présenté dans la **figure 4**. Il est un peu plus complexe qu'un simple oscillateur à relaxation, mais cela est dû aux possibilités de connexion du neurone.

Le condensateur C4 joue le rôle du seau, tandis que les transistors T1 et T2 forment le commutateur qui permet de vider ce seau. Le niveau qui fait déclencher le commutateur est déterminée par les résistances R1 et R2. Le seau est rempli par les impulsions arrivant via D4, qui combine les signaux appliqués aux trois entrées supérieures des neurones (stimuli) (K1, broches 7, 6 et 5).

C1/C2/C3, D4, T5 et C4 forment une pompe à diodes avec trois entrées (C1, C2 et C3). Chaque impulsion entrante charge l'un des condensateurs C1, C2 ou C3. D4 transfère cette charge à C4, ce qui a pour effet d'augmenter la tension. T5, un émetteur suiveur, copie cette tension sur l'anode de D4, ce qui permet d'additionner les impulsions. Par conséquent, la tension sur C4 augmente à chaque impulsion. Sur un oscilloscope, la tension sur C4 ressemble à un escalier.

Lorsque la tension sur C4 devient supérieure à la tension de base de T1 (la tension réglée avec R1 et R2 plus sa chute de diode B-E), T1 commence à conduire. Un courant commence à circuler de C4 à travers P7 vers l'émetteur de T1 et de son collecteur dans la base de T2, jusqu'à la masse. Cela fait en sorte que T2 conduit également, faisant chuter la tension de base de T1 en dessous de celle de son émetteur. T1 est maintenant passant et C4 peut se décharger rapidement. Le seau bascule ainsi. Lorsque le courant de décharge de C4 devient suffisamment faible, T2 cesse de conduire et la base de T1 remonte au niveau de tension défini par R1 et R2. Comme la tension sur C4 est presque nulle, la tension de base de T1 est maintenant beaucoup plus élevée que sa tension d'émetteur et T1 se bloque. Le circuit revient à son état initial, prêt pour un nouveau cycle de remplissage.

Cette configuration particulière de T1 et T2 est parfois appelée thyristor transistorisé, populaire dans les années 1970. Il est crucial

que T2 n'ait pas une valeur bêta (h_{FE}) trop élevée, pour éviter une conduction continue même quand le courant de décharge est presque nul. Un transistor de type A ou B est idéal pour cette application, mais un transistor de type C pourrait être trop sensible. Le BC547A est recommandé pour T2, tandis que pour T5, un autre BC547, la valeur de bêta est moins critique, ce qui autorise l'utilisation d'un BC547A, simplifiant ainsi la nomenclature.

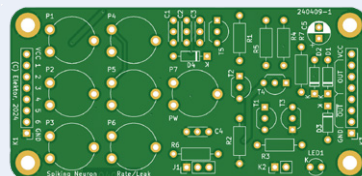
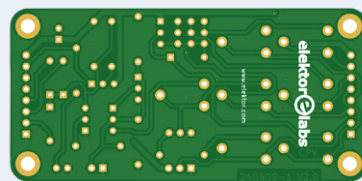
Entrées et sorties

Lorsque le condensateur C4 est en cours de charge, le signal au niveau du nœud formé par les résistances R1 et R2 est élevé ; lorsqu'il se décharge, ce signal devient bas. Le potentiomètre P7 permet de contrôler la durée de cette décharge. Une valeur basse sur P7 résulte en une impulsion active-basse courte sur R1/R2. Le point de test K2 permet de mesurer cette condition.

Le signal présent sur R1/R2 est tamponné par le transistor PNP T3. Les diodes D1, D2, et D3 génèrent trois sorties actives basses pour le neurone. Lorsque T3 est bloqué, ces sorties sont relevées par la résistance R4, et tirées vers le bas quand T3 est conducteur. De même, le blocage de T3 entraîne celui de T4, désactivant ainsi les sorties actives hautes du neurone (K3, aux broches 5, 6, et 7). Lorsque T3 conduit, T4 conduit également, activant ces sorties. La LED1 s'illumine pour signaler une impulsion, indiquant que les sorties sont à basse impédance lorsqu'une impulsion est émise, et à haute impédance dans le cas contraire. Cela évite que deux neurones



Liste des composants



Résistances

R1, R3, R7 = 1 kΩ
R2 = 3,9 kΩ
R5 = 10 kΩ
R4, R6 = 100 kΩ

Condensateurs

C1, C2, C3, C4 = 100 nF, pas de 2,5 mm
ou de 5 mm.
5 mm
C5 = 10 μF, 16 V, pas de 2,5 mm

Semi-conducteurs

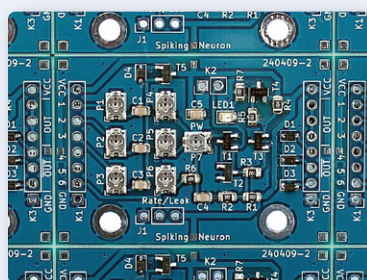
D1, D2, D3, D4 = 1N4148
T1, T3, T4 = BC557
T2, T5 = BC547A

Divers

JP1 = connecteur à 3 voies, pas de 2,54 mm
K1 = socket à 8 broches, pas de 2,54 mm
K2 = connecteur à 2 voies, pas de 2,54 mm
K3 = connecteur à 8 broches,
au pas de 2,54 mm

Essais avec des CMS

Pour optimiser l'expérimentation avec un réseau neuronal, j'ai conçu un modèle réduit de neurone en utilisant des composants montés en surface (CMS).



J'ai demandé au fabricant de circuits imprimés de produire un panneau 3 x 3 de ces neurones, avec des rainures en V (voir l'image ci-dessous). En incluant des plots de distribution de l'alimentation autour de la carte, ils peuvent tous être connectés facilement à l'alimentation avec des ponts de fils ou des résistances de 0 Ω. En utilisant du fil fin, il est rapide de câbler un réseau comme le montre la figure 1, et même de l'étendre avec un neurone d'entrée et de sortie supplémentaire (comme je l'ai fait). Pour faciliter les choses, il est recommandé de procéder à un câblage systématique afin de mémoriser aisément les connexions des sorties aux entrées.

s'influencent mutuellement lorsqu'ils pilotent le même neurone.

Les deux types de sorties permettent au neurone soit de stimuler soit d'inhiber un autre neurone. Cette fonctionnalité est plus claire lorsqu'on examine les entrées du neurone. Il existe également deux types d'entrées. Les trois entrées supérieures de type « AC » se connectent à la pompe à diodes déjà mentionnée. Les trois entrées inférieures de type « DC » se connectent chacune à C4 via un potentiomètre de poids, permettant de contrôler directement la charge du condensateur par chargement ou déchargement.

L'entrée inférieure (K1, broche 2) est équipée d'un cavalier J1 pour simplifier les connexions standard. Lorsque les broches 1 et 2 de J1 sont court-circuitées, le potentiomètre P6 offre une fuite réglable. En court-circuitant les broches 2 et 3, P6 charge C4, provoquant l'oscillation du neurone. La résistance R6 assure que cette oscillation est possible sur toute la gamme de P6.

Pour transformer les entrées AC en DC, il faut remplacer C1, C2, et C3 par une diode (avec l'anode connectée au potentiomètre), substituer D4 par un pont de fils, et ne pas monter T5. L'installation des trois diodes est nécessaire pour éviter que les entrées des différents neurones s'interfèrent mutuellement. Remplacer C1, C2, et C3 par un pont de fils créerait davantage d'entrées de fuite.

Le circuit n'est pas critique en termes d'alimentation et peut fonctionner avec des tensions variant de 3,5 V à 15 V ou plus, selon l'adaptation de C5. La consommation de courant est principalement déterminée par la LED et reste inférieure à 1 mA pour une alimentation de 5.

Caractéristiques

En résumé, nous pouvons maintenant dresser la liste suivante des caractéristiques de notre neurone à intégration-et-tir avec fuite :

- 3 entrées stimulus/AC avec des poids individuels
- 3 entrées d'inhibition/fuite/DC avec des poids individuels
- 3 sorties de stimulus (actif haut)
- 3 sorties d'inhibition/fuite (actif bas)
- Largeur d'impulsion de sortie réglable (« intensité »)
- Générateur d'impulsions à fonctionnement libre

Tester le réseau

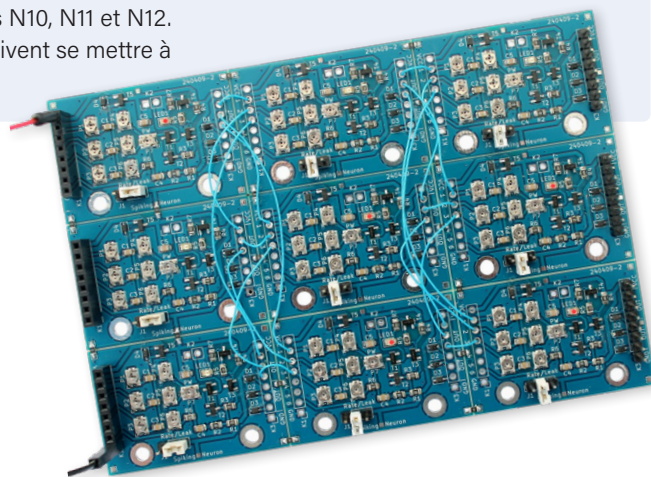
Voici une bonne méthode pour vérifier si le réseau fonctionne comme il se doit. Tournez les potentiomètres P1 à P6 de chaque nœud à fond dans le sens inverse des aiguilles d'une montre, c'est-à-dire réglez-les à leur valeur maximale. Mettez P7 de chaque nœud en position médiane. Sur le neurone situé dans le coin supérieur gauche du panneau, appelons-le N00 (Nrow,column), placez un cavalier sur les broches 2 et 3 de J1. Cela permet de connecter P6 à VCC. Alimenter le panneau. La LED de N00 commence à clignoter. Le taux peut être réglé avec P6 et l'intensité avec P7. Ajustez ensuite les potentiomètres de poids correspondants (P1 pour moi) sur les neurones connectés N10, N11 et N12. Leurs LED doivent commencer à clignoter. Plus la valeur du potentiomètre de poids est petite, plus la LED clignote rapidement.

Retirez le cavalier de N00 et placez-le sur N01, le neurone situé en dessous. Comme précédemment, la LED commence à clignoter, et vous devriez pouvoir ajuster la vitesse de clignotement (P6) et l'intensité (P7) de la LED. Comme précédemment, ajustez le potentiomètre de poids (P2 dans mon cas) des neurones connectés N10, N11 et N12. A nouveau, leurs LED doivent se mettre à clignoter.

Répétez cette procédure pour les autres neurones. Évidemment, les neurones de sortie n'étant pas connectés à d'autres, il n'est pas nécessaire de régler leurs poids. Si toutes les LEDs peuvent être contrôlées comme décrit ci-dessus, cela signifie que la partie stimulus du réseau est opérationnelle. Pour tester la partie fuite/inhibition de chaque nœud, oscillez les neurones d'entrée en plaçant un cavalier sur les broches 2 et 3 de J1. Puis, créez un court-circuit entre les broches 1 et 2 de JP1 sur un autre nœud, ce qui devrait modifier significativement le clignotement de sa LED.

Il ne vous reste plus qu'à entraîner le réseau pour une tâche spécifique. Je laisse cet exercice au lecteur, car je n'en ai pas encore exploré les possibilités.

Vous pouvez observer mon réseau neuronal 3x3 en action. Il semble que les diodes clignent au hasard, mais bien sûr, ce n'est pas le cas. Le réseau exécute une tâche complexe que je ne saisis pas encore totalement.



- LED pour visualiser le feu des neurones
- Tension d'alimentation : 3,5 V à 15 V

Créer un réseau de neurones à impulsions

Vous aurez probablement besoin d'au moins sept neurones pour un petit réseau neuronal capable d'accomplir une tâche très simple (Figure 1). Dans un tel réseau, il y a deux

neurones d'entrée, deux neurones de sortie et trois neurones dans la couche dite cachée. La raison pour laquelle il en est ainsi et son fonctionnement sortent du cadre de cet article. Consultez Wikipedia et ChatGPT pour plus d'informations.

Pour configurer notre petit réseau, les neurones d'entrée sont connectés à des capteurs comme des photorésistances ou

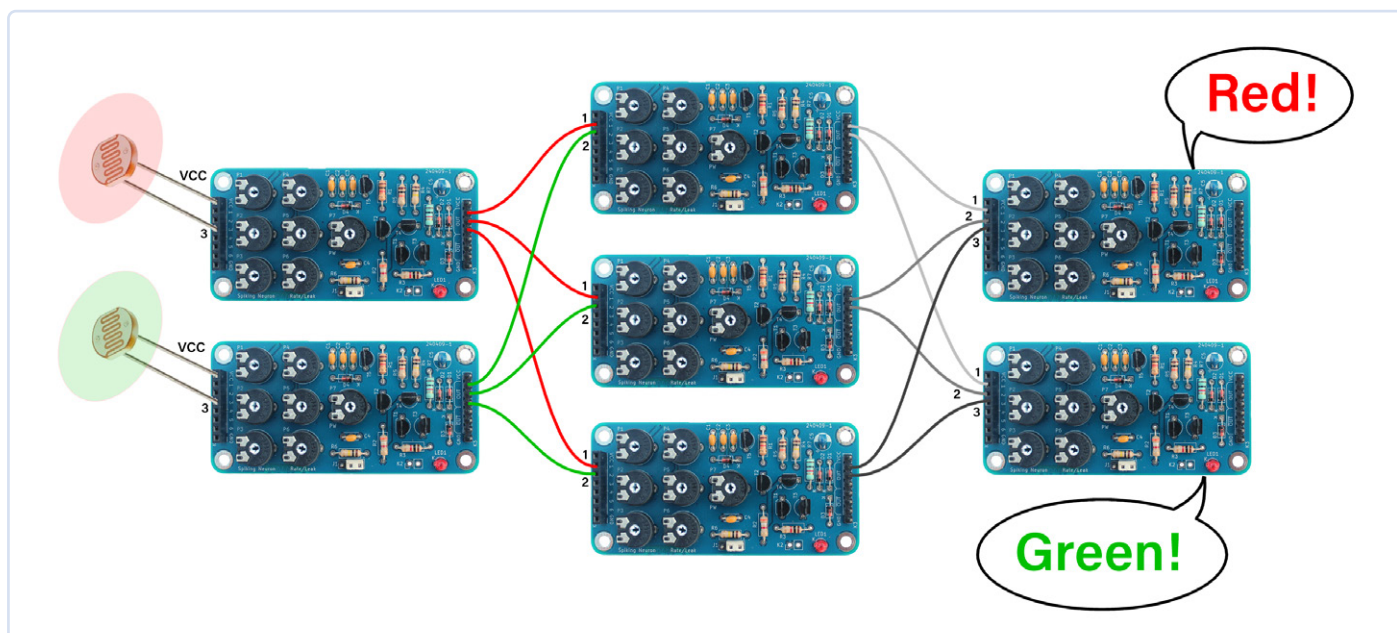


Figure 5. Lorsqu'il est configuré (entraîné) correctement, ce simple réseau neuronal peut faire la distinction entre la lumière rouge et la lumière verte et détecter par exemple l'état d'un feu de circulation. Très astucieux. Pensez-vous vraiment que vous pouvez accomplir une tâche aussi complexe avec moins de composants ? Notez que les connexions de l'alimentation électrique ont été omises pour plus de clarté.

des thermistances, ou à tout autre dispositif capable de convertir une grandeur physique en tension ou en courant. Ces capteurs, une fois connectés à une des entrées de courant continu du neurone, peuvent réguler le taux de stimulation du neurone. Les signaux de sortie des deux neurones d'entrée sont alors dirigés vers les trois neurones de la couche cachée, tandis que les sorties de cette dernière alimentent les neurones de sortie, comme montré dans la **figure 5**.

Il suffit maintenant de régler les potentiomètres de tous les neurones pour que le réseau génère le résultat souhaité. Certes, dans le réseau de la figure 5, cela implique de régler une vingtaine de potentiomètres et, honnêtement, je ne sais pas exactement comment procéder, mais je suis convaincu que c'est faisable. En quelque sorte. C'est ce qui rend ce genre de projet excitant : il laisse beaucoup de place à l'expérimentation.

En ce qui concerne l'expérimentation, les valeurs des résistances, potentiomètres et condensateurs à gauche du thyristor transistorisé T1/T2 (jusqu'à P7 inclus) peuvent être modifiées. N'hésitez pas à essayer d'autres valeurs. Je recommande toutefois de conserver des taux des impulsions relativement élevés, pour une meilleure visibilité lors de l'utilisation d'un oscilloscope. Les taux des impulsions dans les neurones biologiques sont plutôt faibles, souvent inférieurs à 100 Hz. Les fichiers de conception du neurone à impulsions (traversants et SMT) peuvent être téléchargés ici [1].

240409-04

Questions ou commentaires ?

Envoyez un courriel à l'auteur (clemens.valens@elektor.com), ou contactez Elektor (redaction@elektor.fr).



Produits

> Get Started with the MAX78000FTHR Bundle
www.elektor.fr/19798



FEATURED TOPIC

Visit our **Embedded & AI** page for articles, projects, news, and videos.

www.elektormagazine.com/
embedded-ai



LIENS

[1] S. Cording, "Understanding the Neurons in Neural Networks (Part 1): Artificial Neurons," [elektormagazine.com: https://www.elektormagazine.com/articles/neural-networks-part-1-artificial-neurons](https://www.elektormagazine.com/articles/neural-networks-part-1-artificial-neurons)

[2] This project at Elektor Labs: <https://www.elektormagazine.com/labs/spiking-neural-network-node>