

08 Un pont tournant pour le réseau

Un pont tournant alimentant une jolie rotonde où on peut parquer ses plus beaux modèles de locomotives est très certainement la pièce maîtresse d'un réseau de trains miniatures. En cas de manque de place, une simple plaque tournante permettant de virer une locomotive dans une petite gare terminus est également une animation intéressante. Nous allons donc voir comment concevoir un pont tournant ou une plaque tournante, en utilisant un moteur pas à pas géré par Arduino. Ce projet, conçu pour un réseau analogique, fait appel à un petit moteur pas à pas peu puissant et est plutôt destiné à un pont à l'échelle N ou une plaque en N ou H0 pour virer une petite locomotive légère. Cependant, il peut servir de base d'inspiration pour un pont motorisé avec un moteur plus puissant dans un réseau numérique.



Prototype du pont de l'auteur pour vérifier le programme



Pont tournant sur un réseau miniature (photo Yann Baude LR908)

8.1 / CAHIER DES CHARGES

J’ai déjà évoqué la commande d’un moteur pas à pas par Arduino dans le guide « Animez votre réseau » (paragraphe 4.6 page 86) et j’ai indiqué quelques considérations théoriques à prendre en compte pour une plaque tournante. Cette fois, **je vais décrire le projet de A à Z en appliquant la méthode décrite au chapitre 6**. Commençons donc par décrire ce que doit réaliser le projet et définir ainsi ce qu’on pourrait appeler cahier des charges.

Le pont (terme utilisé par la suite bien que ce soit la même chose pour une petite plaque tournante) doit permettre plusieurs choses :

- 1. Le pont doit être relié à une voie permettant l’entrée et la sortie des locomotives.
- 2. Des voies de garage doivent permettre de parquer les locomotives.
- 3. L’alimentation de ces voies doit permettre à la locomotive d’avancer ou de reculer et également d’être à l’arrêt.
- 4. La polarité des voies doit être coordonnée afin de ne pas créer de court-circuit lors des manœuvres d’entrée ou de sortie.
- 5. La voie située sur le pont doit permettre toutes ces manœuvres (entrée, sortie, arrêt).
- 6. Les fils d’alimentation de la voie du pont ne doivent pas s’entortiller suite à de nombreuses manœuvres du pont.
- 7. La rotation du pont doit permettre un alignement parfait avec la voie sélectionnée.
- 8. La rotation du pont doit être commandée par une interface simple à utiliser.
- 9. Un écran doit permettre de contrôler toutes les manœuvres du pont.

Comme on le voit, on doit à la fois régler des **problèmes de mécanique** (positionnement du pont) et des **problèmes d’électricité** avec l’alimentation des voies sans créer de court-circuit. Le **point 1** est évident, la voie d’entrée est reliée au reste du réseau alors que le pont et ses voies de garage constituent un sous-ensemble à part. Les **points 2 et 7** sont faciles à solutionner en utilisant un moteur pas à pas : pour aller d’une voie à une autre, le logiciel sait combien de pas il faut parcourir dans un sens ou dans l’autre. On peut utiliser un **clavier à 16 touches pour commander le pont dans sa globalité** (positionnement et alimentation des voies), ce qui résoud le **point 8**, ainsi qu’un écran I2C pour résoudre le **point 9**. Les **points 3 et 5** nécessitent des relais inverseurs afin de changer la polarité des rails (sens d’avancement) ainsi que des **relais simples** pour établir ou couper les alimentations des voies. Le logiciel s’occupera de coordonner la polarité des voies pour éviter les courts-circuits, ce qui résoud le **point 4**. Enfin, **pour éviter aux fils d’alimentation du pont de s’entortiller (point 6)**, il faut interdire à celui-ci de faire un tour complet et définir une position par laquelle il ne doit jamais repasser. Notre cahier des charges a été défini et quelques solutions envisagées. Il reste à les mettre en pratique.

Le projet est ensuite construit en plusieurs étapes : lecture des touches du clavier afin de définir la voie, mouvement du pont dans un sens ou dans l’autre de la quantité de pas permettant de s’aligner avec la voie choisie, affichage sur l’écran des différentes commandes ou positions du pont, commande des cartes à relais. On ne peut passer qu’à l’étape suivante **que si l’étape précédente donne entièrement satisfaction**.

8.2 / CHOIX DES COMPOSANTS

Pour ce projet, on utilisera le **moteur pas à pas 28BYJ-48** (2048 pas entier par tour, 4096 demi-pas), **afin de motoriser le pont tournant Peco échelle N** vendu en kit à monter. Le choix de l’échelle N est imposé par le fait que le moteur choisi n’est pas très puissant et qu’il faut donc limiter le poids sur le pont à manœuvrer. L’interface de ce moteur est commandée avec quatre sorties d’Arduino.

On utilise également **un clavier à 16 touches** pour sélectionner les voies (interface homme-machine), ce qui nécessite 8 sorties supplémentaires (4 lignes et 4 colonnes). Ce clavier doit permettre de choisir la voie devant laquelle le pont se positionne, mais aussi de faire manuellement des mouvements de large amplitude ou bien d’un seul pas.

Enfin, il faut aussi commander l’alimentation des voies avec des **cartes relais** et faire en sorte de ne pas créer de court-circuit ; plus on a de voies reliées au pont, plus il faut de relais et de sorties pour les

commander. La polarité du courant alimentant les voies est confiée à des **relais DPDT** (Double Pole Double Throw) **ou relais 2RT** (deux contacts repos-travail).

L’écran de type I2C ne nécessite que les sorties SDA et SCL pour être commandé. Il est très vite apparu qu’une carte Arduino Uno n’est pas suffisante (sauf si le pont n’est relié qu’à deux ou trois voies) ; le choix de la carte Arduino s’est donc porté sur la **carte Mega 2560 Rev3**, qui permet de compléter ultérieurement le projet en ajoutant des voies. **Le projet est décrit ici avec une voie d’entrée et quatre voies de sorties vers la remise.**

8.3 / GÉOMÉTRIE DU PROJET

Comme on l’a dit, l’écart angulaire entre deux voies doit permettre un nombre entier de pas de déplacement. En choisissant un angle de 22,5°, il faut 128 pas entiers ou bien 256 demi-pas pour ce moteur pour aller d’une voie à la voie adjacente. On peut travailler en pas entiers et utiliser la bibliothèque Stepper, mais pour une plus grande précision, il vaut mieux travailler en demi-pas car ce n’est pas plus compliqué comme on le verra un peu plus loin. Avec un angle de 22,5°, nous obtenons 16 positions possibles pour les voies (numérotées C0 à C15, C pour cible). La **figure 8-1** montre la géométrie du projet et la façon dont ces voies sont positionnées sur ces positions cibles. Cette figure définit un index (bleu ciel) situé en face de C0 qui représente l’arrière d’une machine qui entre sur le pont (on y reviendra plus loin).

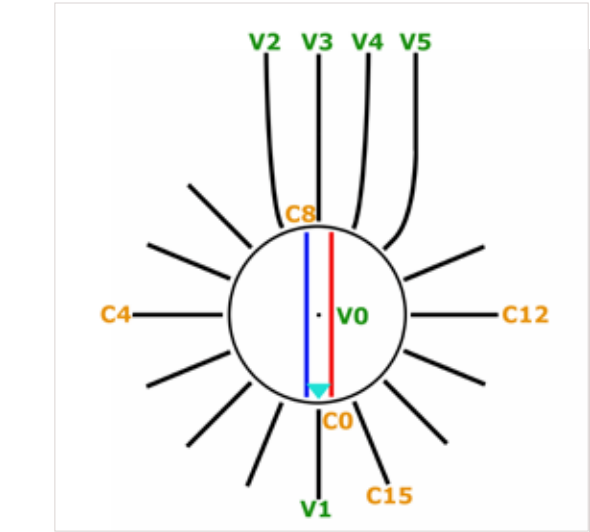


Figure 8-1

La voie d’entrée du pont est la voie V1, sur la position C0. Les voies de sortie vers la remise sont les voies V2 à V5, positionnées sur C7 à C10. Ce choix du nombre de voie et de leur position est arbitraire ; vous pouvez en faire un autre, c’est le programme qui fera la correspondance voie-position cible. **La voie sur le pont est la voie V0.**

8.4 / UTILISATION
D’UN MOTEUR PAS À PAS
UNIPOLAIRE EN DEMI-PAS

Pour faire tourner un moteur unipolaire en pas entiers, il suffit d’**alimenter successivement ses bobines dans un certain ordre** (qui détermine le sens de rotation) et à une certaine fréquence (qui détermine la vitesse de rotation). Donc, on alimente successivement bobine 1, bobine, 2, bobine 3, bobine 4. **Pour travailler en demi pas**, il faut alimenter la bobine 1 (le rotor se cale juste en face d’elle), puis on alimente simultanément bobine 1 ET bobine 2 (le rotor se cale entre la bobine 1 et la bobine 2 et a donc parcouru un demi-pas), puis on alimente QUE la bobine 2 (le rotor parcourt un demi-pas et se cale en face d’elle), etc. avec les autres bobines. Or, pour alimenter les bobines,

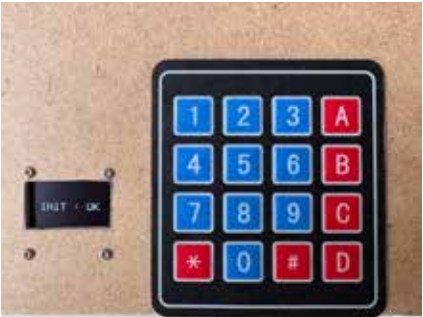
il suffit d’alimenter les entrées de l’interface de commande à base d’ULN2003 ou ce qui revient au même, les sorties de la carte Arduino reliées à ces entrées de l’interface. Si ces sorties sont choisies sur un même Port, on peut écrire directement dans le Port au lieu d’utiliser des digitalWrite. Ici, **nous utilisons le Port F** (sorties 54 à 61) de la carte Mega pour commander le moteur pas à pas qui n’a besoin d’être relié qu’aux quatre premières sorties du Port (54 à 57 ou encore A0 à A3). Les valeurs à afficher sur le Port sont successivement 1, 3, 2, 6, 4, 12, 8, 9 (si vous convertissez en binaire, vous retrouverez la séquence décrite plus haut pour alimenter les bobines).

8.5 / INTERFACE
HOMME-MACHINE

Un **clavier à 16 touches**, comme on en trouve dans les kits de début, suffit pour commander le pont tournant ; il est relié aux broches 4 à 11 de la carte (de 4 à 7 pour les colonnes et de 8 à 11 pour les rangées). Sa gestion est confiée à la **bibliothèque Keypad** qu’il faut installer dans l’IDE. La fonction getKey() permet de récupérer le caractère de la touche enfoncée, une variable de type char appelée keyPressed : en fonction de ce caractère, on réalise telle ou telle fonction. C’est le caractère récupéré qui permet de faire la correspondance entre voie sélectionnée et cible à aller chercher (ce qui dépend de votre disposition de voies).

Les touches **1 à 9** permettent de sélectionner la voie en face de laquelle le pont se positionne. La touche ‘**A**’ (comme ahead ou avance) permet la marche avant, la touche ‘**B**’ (comme back ou recule) permet la marche arrière et la touche ‘**C**’ (comme calm ou stop) permet d’arrêter la machine.

Interface Homme
Machine (IHM) du pont :
un clavier pour
commander, un écran
pour contrôler



D’autres touches permettent un déplacement précis du pont en rotation. La touche ‘**D**’ du clavier sert à effectuer un déplacement de **256 demi-pas sens des aiguilles d’une montre (CW pour clockwise)**, ceci afin de faire un positionnement grossier du pont. La touche ‘**0**’ permet un déplacement de **16 demi-pas dans le sens CW**. La touche ‘*****’ permet un déplacement **d’un demi-pas dans le sens CW** alors que la touche ‘**#**’ permet **la même chose dans le sens contraire des aiguilles d’une montre (CCW pour counterclockwise)**. On utilise ces déplacements possibles autant de fois que nécessaire pour obtenir l’alignement parfait avec la voie d’entrée V1.



Pont et rotonde (photo Yann Baude)

8.6 / INITIALISATION DU PONT

Pour que le pont sache combien de demi-pas il doit parcourir pour aller chercher une voie, il faut que **son initialisation soit faite juste en face de la voie d’entrée qui constitue alors sa référence (C0)**. Une fois que cette initialisation est faite, tout devient très simple. Si CD est la cible de départ et CA la cible d’arrivée, le déplacement à faire est $(CA - CD) * 256$: si ce déplacement est positif, le sens de rotation doit être celui des aiguilles d’une montre (CW), si le déplacement est négatif, c’est le sens inverse (CCW pour counterclockwise). Ce déplacement peut être long, **mais il ne traverse jamais la position C0**, ce qui permet de ne pas entortiller les câbles d’alimentation de la voie du pont. Le sens de déplacement est géré par une variable booléenne qui vaut ‘true’ pour CCW et ‘false’ pour CW. De plus, **le déplacement du pont ne peut se faire que si la voie V0 n’est pas alimentée**, ce qui assure que la machine sur le pont est bien arrêtée.

Il faut maintenant aller chercher la voie d’entrée pour procéder à l’initialisation du pont. Comme

le moteur est démultiplié, on ne peut pas le manœuvrer à la main ; il faut donc aller chercher l’alignement parfait avec la voie d’entrée en utilisant le moteur. Grâce aux touches ‘**D**’, ‘**0**’, ‘*****’ et ‘**#**’, on peut **manœuvrer le pont jusqu’à être parfaitement en face de la voie d’entrée V1**.

Une fois le pont amené précisément en face de la voie d’entrée, **il suffit de reseter la carte** (donc le programme) pour que le pont prenne la voie V1 (donc la cible C0) pour référence de ses déplacements. Cette référence peut être perdue si le pont n’est pas en face de la voie V1 et que l’alimentation est coupée, qu’on effectue un reset, qu’on ouvre le moniteur série (ce qui équivaut à un reset) ; **dans ce cas, la procédure d’initialisation est à refaire**. Mais cette référence peut aussi être conservée si on prend soin de mettre le pont face à la voie 1 avant de couper l’alimentation du montage.

Lorsque l’initialisation est faite, il suffit de taper sur le clavier la voie voulue (de 1 à 5 pour ce projet) pour que le pont aille la chercher.

8.7 / GESTION DE L’ALIMENTATION DES VOIES

- Il y a **deux objectifs** à atteindre :
- 1. Une machine doit pouvoir entrer et sortir du pont mais aussi s’arrêter dessus pour permettre le mouvement de rotation du pont.
 - 2. La voie du pont V0 ne doit pas être en court-circuit avec les autres voies (V1 à V5).

Dans le premier cas, il faut un relais 2RT pour inverser le courant sur la voie V0 et un relais 1RT (R0) pour couper l’alimentation de la voie V0, soit deux sorties de commande des relais. Pour le deuxième cas, seule la voie située en face de l’index (voir **figure 8-1**) **doit être alimentée temporairement et de la même façon que la voie V0 pour permettre l’entrée ou la sortie d’une machine**. Ceci nécessite donc d’autres relais de type 1RT (Rx) et d’autres sorties pour les commander. La **figure 8-2** montre l’alimentation des voies de sortie en fonction de l’alimentation du feeder de la voie V0.

Le programme gère le déplacement d’une machine sur le pont avec une variable **sensMarche** qui vaut 0 pour arrêt, 1 pour marche avant et 2 pour marche arrière. Ses trois cas sont commandés par les touches ‘A’, ‘B’ et ‘C’.

L’index bleu ciel (**figure 8-1**) représentant l’arrière d’une locomotive, il faut aller chercher **la cible C8 (voie V3 pour ce projet) si on veut sortir du pont en marche avant**. Et pour sortir, il faut que C0 (voie V1) soit aussi alimentée avec la bonne polarité, et pour l’instant elle n’est pas alimentée puisque seule la voie en face de l’index l’est. La solution la plus simple consisterait à prévoir une alimentation à part pour le pont et ses voies de garage (V0 et V2 à V5) et à alimenter la voie V1

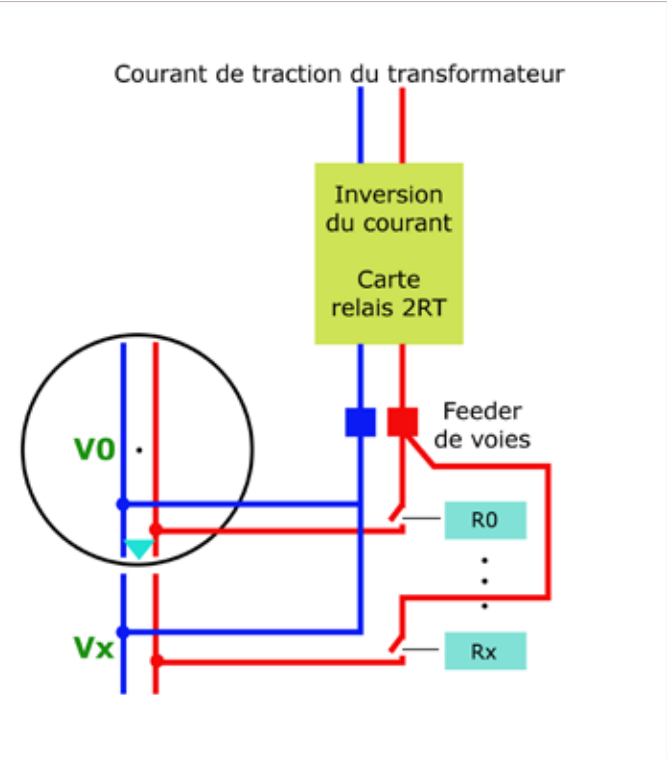


Figure 8-2

par l’alimentation du réseau en veillant à ce que ces deux alimentations soient polarisées comme il convient pour les entrées ou sorties du pont. **Pour une automatisation totale**, on peut aussi prévoir un **deuxième relais 2RT** pour inverser le courant sur V1 chaque fois que le pont est en face de la voie V3 (C8) et qu’on veut sortir en marche avant. **Dans ce cas, le pont tournant alimente une portion du réseau : la voie qui arrive jusqu’à lui.**

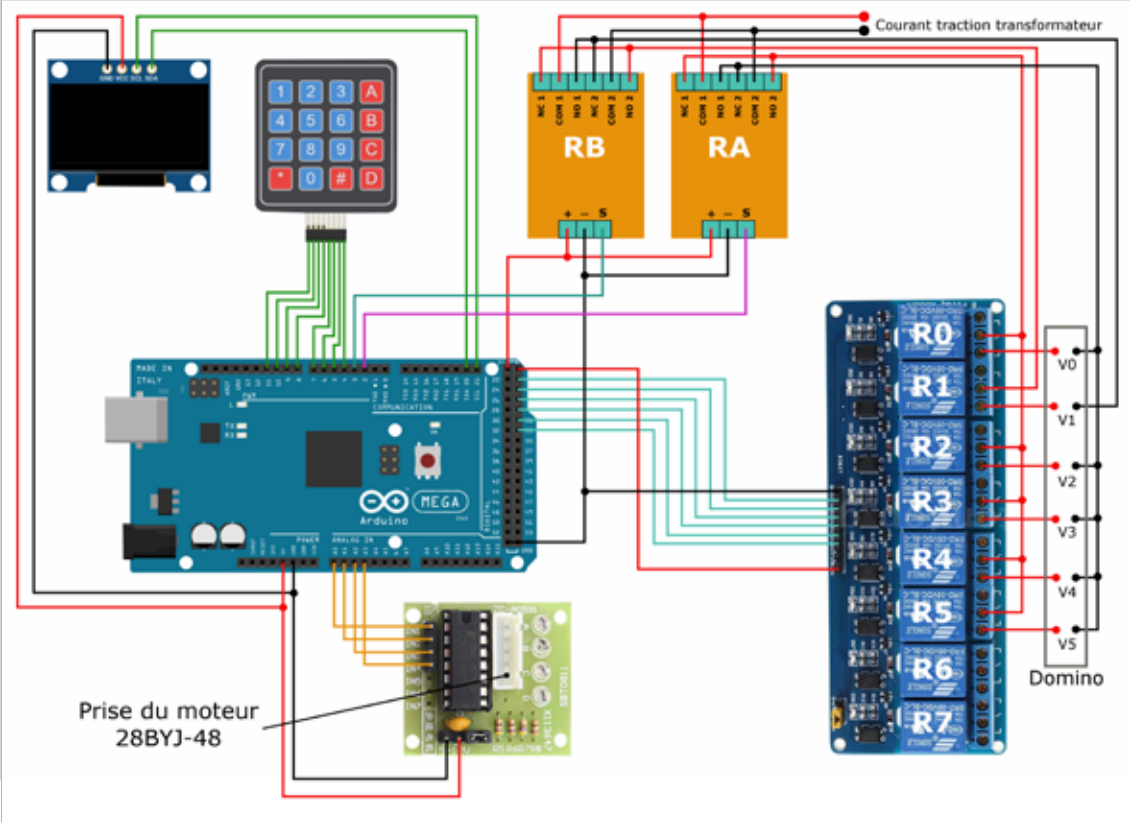


Figure 8-3

8.8 / PLAN DU MONTAGE

La **figure 8-3** montre comment relier tous les composants à la carte Mega 2560, **pour cette deuxième solution**. Outre la carte, nous trouvons l’interface de commande du moteur, le clavier à 16 touches, une carte 8 relais 1RT (R0 à R7), deux cartes à un relais 2RT (RA et RB), et un écran OLED relié en I2C. Le clavier est raccordé aux broches 4 à 11 et l’interface de commande du moteur est reliée aux broches A0 à A3. La carte relais 1RT utilise 6 relais (R0 à R5) pour alimenter ou non les voies V0 à V5 ; ces relais sont commandés par un état LOW sur la sortie concernée. Le relais **RA permet d’inverser le courant sur le feeder voie V0** et le relais **RB sur la voie V1** ; ces relais sont commandés par un état HIGH sur la broche 2 ou 3.

Les voies de garage du pont (V2 à V5) sont reliées au feeder de la voie 0 (un simple domino) et un rail passe par les relais R2 à R5, conformément à la **figure 8-2**. La sortie du relais RB alimente la voie V1 et un rail passe par le relais R1. Il est **conseillé de prévoir une alimentation de 5 V DC pour alimenter les cartes relais** (fil marron (+) et bleu (GND)). C’est le programme qui commande le relais RB en fonction du sens de marche voulu lorsque la voie 3 est sélectionnée.

Si vous préférez alimenter la voie 1 indépendamment du pont, n’utilisez pas le relais R1 et ne connectez pas la carte relais RB.

8.9 / INCORPORATION AU PONT TOURNANT PECO

Soudez deux fils souples aux extrémités des rails du tablier du pont, puis passez ces fils sous le tablier en pratiquant deux petites encoches dans le but de rejoindre l'axe de rotation (Figure 8-4). Fixez les fils sous le tablier et passez-les le long de l'axe de rotation. Ils ne doivent pas frotter avec le trou dans la fosse du pont ; pour cela, utilisez un petit bout de tube en cuivre (de plomberie, diamètre 12 mm) d'une part pour que les fils soient à l'intérieur, d'autre part pour centrer le mouvement dans la fosse (figure 8-4). Laissez tout de même un peu de mou entre ces fils et l'arrivée du courant (feeder de voies) pour permettre le mouvement. **Pour le reste du montage, suivez la notice sans monter les pièces de cuivre faisant contact.** L'axe du moteur **doit être confondu avec l'axe du pont** ; pour cela, fabriquez une pièce en bois, résine ou métal, réunissant l'axe du tablier et l'axe du moteur. Sans tour ni imprimante 3D, vous pouvez récupérer la partie filetée d'un boulon M4 x 0,70 pour constituer un axe collé à l'époxy dans le trou du tablier ; cet axe est vissé dans un **manchon (tube fileté intérieurement au pas de 0,70)**, ce qui

permet de régler précisément la hauteur du tablier tournant si nécessaire. Une gorge de 3 mm de large est fraisée à la lime aiguille pour recevoir l'arbre moteur comme le montre la figure 8-4.

Toutes les pièces se trouvent facilement sur internet. Il est nécessaire d'agrandir le trou de la fosse du pont en raison de la présence du tube de cuivre. Il faut réduire les sources possibles de frottement et pour cela, **on peut envisager un petit roulement à billes** monté sur le tube en cuivre.

Les voies qui arrivent au pont doivent être posées **en respectant bien l'angle de 22,5° entre deux voies**, être au même niveau que les voies du tablier et leurs rails doivent être légèrement chanfreinés (voir paragraphe 7 de la notice de montage). Tout cela demande un travail soigné mais qui est tout à fait à portée d'un modéliste. Le programme peut aussi prévoir **une correction de position** pour chaque voie (en nombre de demi-pas en plus ou en moins) pour tenir compte du fait qu'une voie n'est pas tout à fait en face de la position cible.



Figure 8-4

8.10 / UTILISATION DU PONT

Positionnez l'index du pont en face de la voie 1 pour permettre l'entrée d'une locomotive avec les touches 'D', 'O', '*' et '#'. Lorsque l'alignement est précis, resetez la carte : **l'initialisation du pont est faite**. Faites avancer la locomotive à vitesse réduite avec la touche 'A' **pour bien la centrer sur le pont**, puis arrêtez la avec la touche 'C'. Sélectionnez la **voie à atteindre** avec la touche adéquate (2 à 5 dans ce projet) : le pont tourne jusqu'à aller en face de cette voie. Faites ensuite **reculer la locomotive** à vitesse réduite avec la touche 'B' pour la faire pénétrer dans la voie de sortie. Arrêtez la avec la touche 'C'. Sélectionnez la voie 1 pour que le pont effectue une rotation vers celle-ci et **soit prêt à recevoir une autre machine**. Vous pouvez aussi sélectionner une autre voie pour sortir une autre machine de la remise. **Pour sortir une locomotive du pont, il faut sélectionner la voie V3** (cible C8), puis agir sur la touche 'A' pour faire avancer la machine.

Le courant dans les voies est automatiquement coupé quand le pont tourne. Le programme affiche des messages sur l'écran OLED et dans le moniteur série, et **il faut bien attendre l'affichage de ces messages avant d'appuyer sur une touche du clavier**.

Il n'est pas possible de faire un tel montage sur les simulateurs d'Arduino. Cependant, vous pouvez récupérer le programme sur Wokwi : <https://wokwi.com/projects/383924965574672385>.

Seuls carte Arduino Mega, clavier et écran sont visibles côté composants. Une barre de LED a été ajoutée pour **simuler tous les relais** : les deux LED les plus à droite représentent les relais RA et RB. Il est donc possible de simuler le fonctionnement, mais sans voir la rotation du moteur. Lorsque celle-ci est terminée, un message est affiché sur l'écran « **Voie X OK** ».

8.11 / AMÉLIORATION POSSIBLE

Les ponts du commerce sont généralement construits avec un espacement angulaire fixe entre les différentes voies possibles ; le principe de ce pont est donc parfait pour le commander. Cependant, si vous construisez vous-même le pont, il n'est pas toujours possible de bien espacer les voies au demi-degré près. Grâce au montage précédent, vous pouvez aligner le pont rigoureusement en face de chacune des voies et noter sa position en nombre

de demi-pas. Il est alors possible de rentrer dans le programme ces positions pour que le nombre de demi-pas à parcourir soit calculé en fonction de la position cible et de la position actuelle. Vous serez ainsi certain de toujours être en face des voies. Ceci ne dispense pas de l'initialisation qui doit être faite de façon rigoureuse en début d'utilisation, afin d'avoir la bonne référence de départ (position C0).

RÉSUMÉ DU CHAPÎTRE 8

Automatiser un pont tournant est un projet complexe qui demande de la réflexion : choix du moteur, choix de l'interface homme-machine, choix des alimentations nécessaires. Ce chapitre a permis de voir comment résoudre les problèmes pour éviter les courts-circuits et faire en sorte que les voies s'alignent parfaitement pour éviter les déraillements. La méthode de conception est à appliquer quel que soit le projet sur votre réseau.