LabView

LabVIEW est un environnement de développement de programmes, développé par National Instruments (NI), similaire aux environnements de développement C et BASIC, mais la différence significative entre LabVIEW et les autres langages informatiques est que d'autres langages informatiques utilisent des langages textuels pour générer du code , Et LabVIEW utilise le langage d'édition graphique G pour écrire le programme, et le programme généré se présente sous la forme d'un diagramme.

1. Analyse

Comme C et BASIC, LabVIEW est également un système de programmation général, avec une énorme bibliothèque de fonctions pour accomplir n'importe quelle tâche de programmation. La bibliothèque de fonctions de LabVIEW comprend l'acquisition de données, GPIB, le contrôle de port série, l'analyse de données, l'affichage et le stockage de données, etc. LabVIEW dispose également d'outils de débogage de programme traditionnels, tels que la définition de points d'arrêt, l'affichage de données et les résultats de sous-routines (sous-VIs) de manière animée, pas à pas, etc., pour faciliter le débogage de programme.

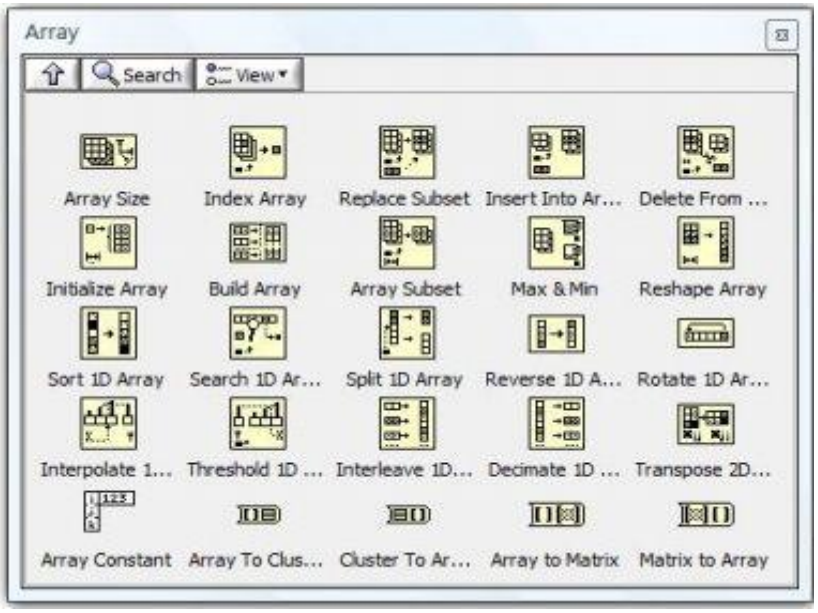
LabVIEW fournit de nombreuses commandes qui ressemblent à des instruments traditionnels (tels que des oscilloscopes et des multimètres), qui peuvent être utilisées pour créer facilement des interfaces utilisateur. L'interface utilisateur est appelée la face-avant dans LabVIEW. À l'aide d'icônes et de fils, les objets du panneau avant peuvent être contrôlés par programmation. Il s'agit du code source graphique, également appelé langage G. Le code source graphique de LabVIEW est similaire à un organigramme dans une certaine mesure, il est donc également appelé code de diagramme de blocs.

2. Caractéristiques

1. Le matériel universel est utilisé autant que possible
2. Il peut exploiter pleinement les capacités de l'ordinateur, possède de puissantes fonctions de traitement de données et peut créer des instruments plus puissants.
3. Les utilisateurs peuvent définir et fabriquer divers instruments en fonction de leurs propres besoins.
4. Utilisez LabVIEW, peut produire le fichier exécutable qui s'exécute indépendamment, c'est un véritable compilateur 32 bits/64 bits. Comme de nombreux logiciels importants, LabVIEW propose plusieurs versions de Windows, UNIX, Linux et Macintosh.
5. On peut se rendre compte des fonctions des différents instruments et vumètres en changeant de logiciel, ce qui est très pratique et équivaut à un logiciel qui est matériel .

3. Domaine d'application

1. ***Test et mesure*** : LABVIEW a été conçu à l'origine pour le test et la mesure, le test et la mesure sont donc le domaine d'application le plus étendu de LABVIEW. Après des années de développement, LABVIEW est largement reconnu dans le domaine du test et de la mesure. Jusqu'à présent, la plupart des instruments de test et des périphériques d'acquisition de données ont des pilotes LabVIEW spécialisés, et LabVIEW peut être utilisé pour contrôler ces périphériques matériels de manière très pratique. Dans le même temps, les utilisateurs peuvent facilement trouver divers toolkits LabVIEW adaptés aux domaines de test et de mesure.
2. ***Contrôler :*** LabVIEW dispose d'un module spécialement utilisé dans le domaine du contrôle-LabVIEWDSC. De plus, les équipements et câbles de données couramment utilisés dans le domaine du contrôle industriel ont généralement des pilotes LabVIEW correspondants. Il est très pratique d'utiliser LabVIEW pour compiler divers programmes de contrôle.
3. ***Simulation*** : LabVIEW contient une variété de fonctions d'opérations mathématiques, particulièrement adaptées à la simulation, à la simulation et au prototypage. Avant de concevoir un équipement électromécanique, vous pouvez utiliser LabVIEW pour construire un prototype de simulation sur ordinateur afin de vérifier la rationalité de la conception.



4. ***Multiplateforme*** : LabVIEW a une bonne cohérence de plate-forme. Le code LabVIEW peut s'exécuter sur les trois systèmes d'exploitation de bureau courants sans aucune modification : Windows, Mac OS et Linux. De plus, LabVIEW prend également en charge divers systèmes d'exploitation temps réel et périphériques embarqués, tels que les PDA, les FPGA et les périphériques RT courants exécutant les systèmes VxWorks et PharLap.

Communication entre arduino et labview

1. Pourquoi l'interfaçage d'Arduino avec LabVIEW est-il nécessaire ?

Ardunio est une carte matérielle open source et il est très facile d'effectuer des tâches compliquées avec. Il possède non seulement de nombreuses fonctionnalités intégrées, mais prend également en charge les périphériques externes. Mais cela nécessite d'écrire différentes lignes de code qui peuvent être lourdes. Voici le LabVIEW pour vous sauver. LabView utilise un langage graphique et ne nécessite pas l'écriture de lignes de code. Au lieu de cela, nous utilisons un diagramme graphique pour effectuer une tâche particulière. Vous pouvez consulter le tutoriel 1 pour plus de détails.

L'interfaçage de LabVIEW et d'Arduino combine à la fois la simplicité de la carte matérielle Aduino et le langage graphique de LabVIEW. Maintenant, nous n'avons pas besoin d'écrire des lignes de code et nous n'avons pas non plus besoin des autres cartes matérielles complexes.

2. Logiciel requis

- Arduino IDE https://www.arduino.cc/en/software
- LabVIEW http://www.ni.com/trylabview/
- Visual Package Manger http://jki.net/vipm/download
- LabVIEW Interface for Arduino(Présent dans Visual Package Manager. Téléchargez-le et installez-le)

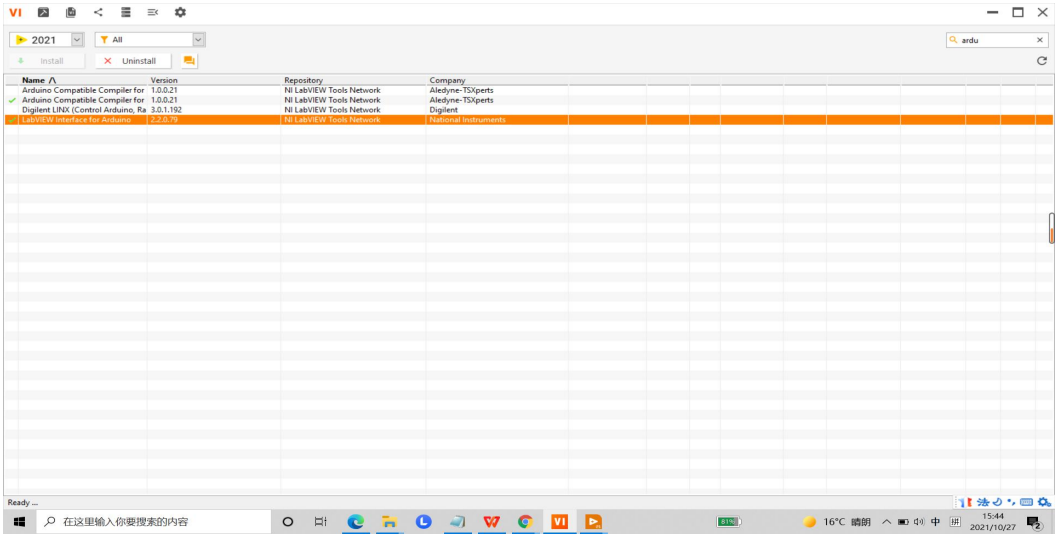
3. Connecter labview et arduino

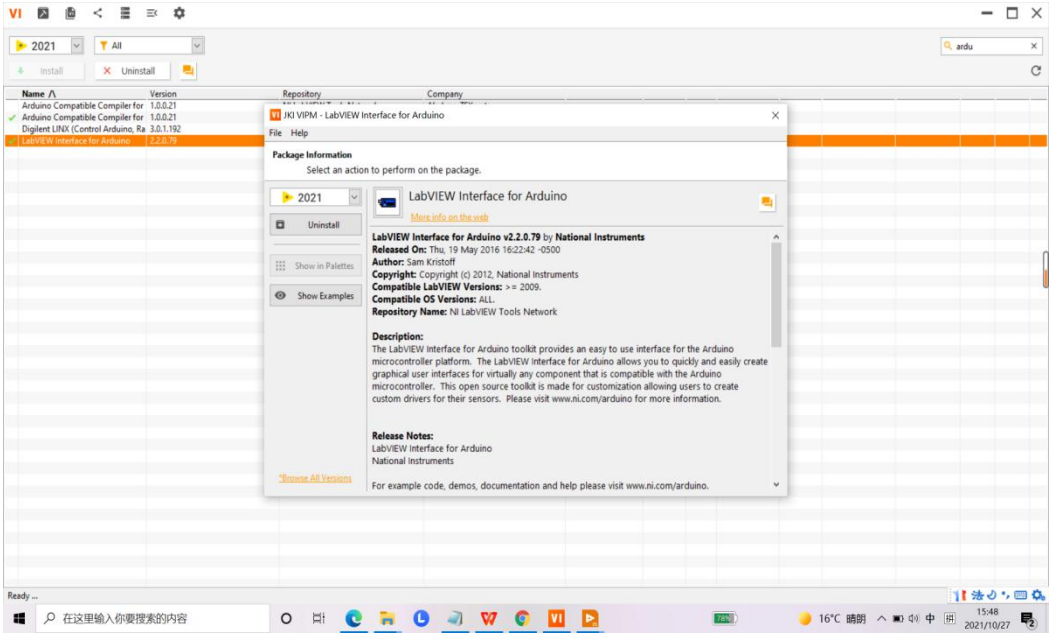
Au début, téléchargez le module sur arduino qui a été développé dans labview, le nom de ce module est **LabVIEW Interface for Arduino**.

Dans l'interface LabVIEW pour le mode Arduino, il vous suffit de programmer le programme Arduino donné dans le contrôleur Arduino, puis de terminer la partie LabVIEW de la programmation.Comme la programmation Arduino n'est pas impliquée, seuls la bibliothèque officielle de fonctions Arduino et le capteur Arduino sont Il y a de grandes limitations dans la sélection.

Nous devons donc télécharger le moteur d'exécution et le logiciel de gestion des packages d'outils (VIPM, VI Package Manager) avant l'installation.

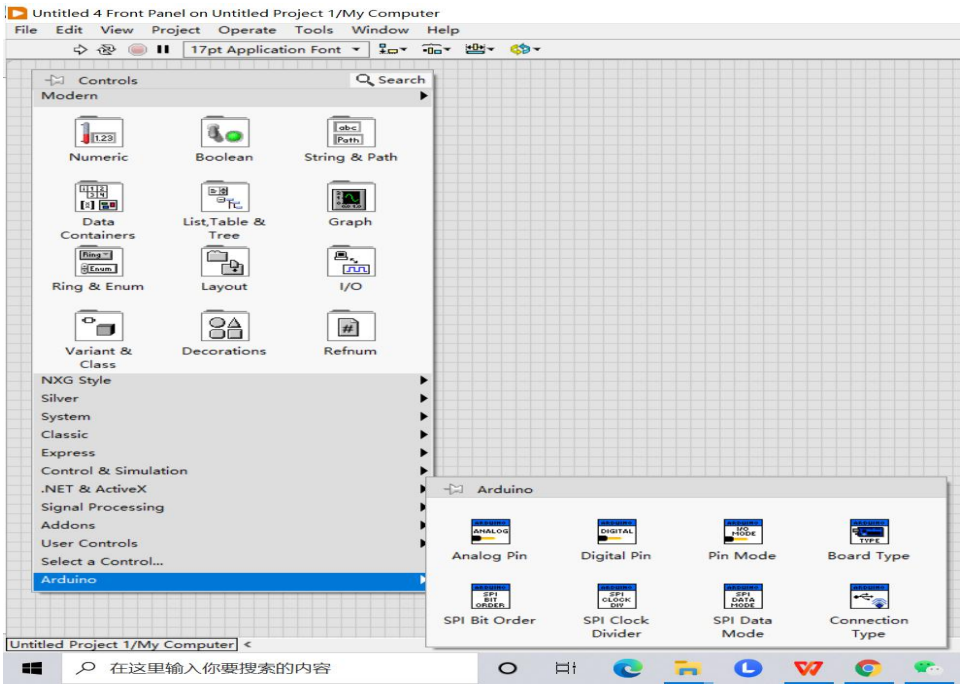
1. Ouvrez JKI VI Package Manager, sélectionnez “LabVIEW Interface for Arduino”





2. Après l'installation réussie, ouvrez LabVIEW, vous pouvez voir le module correspondant dans le schéma fonctionnel:

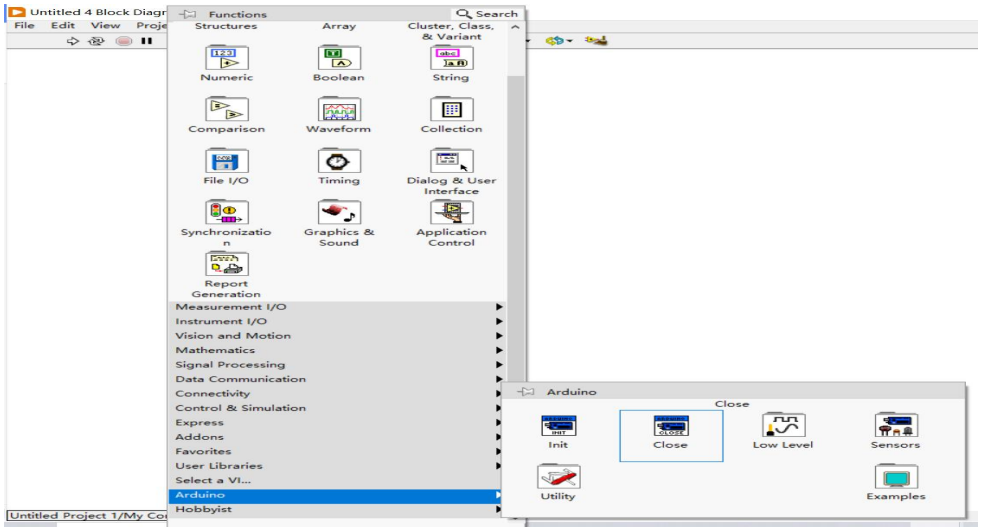
La bibliothèque de fonctions de contrôle et d'opération Arduino apparaîtra dans la colonne des fonctions de la face-avant et du diagramme. Les commandes Arduino incluent les I/S analogiques, les I/O, Pin Mode, Board Type,etc.



(Bibliothèque de contrôle)

La bibliothèque de fonctions d'opération comprend Arduino INIT, Arduino CLOSE, Low Level, Sensors, Exemples, etc.

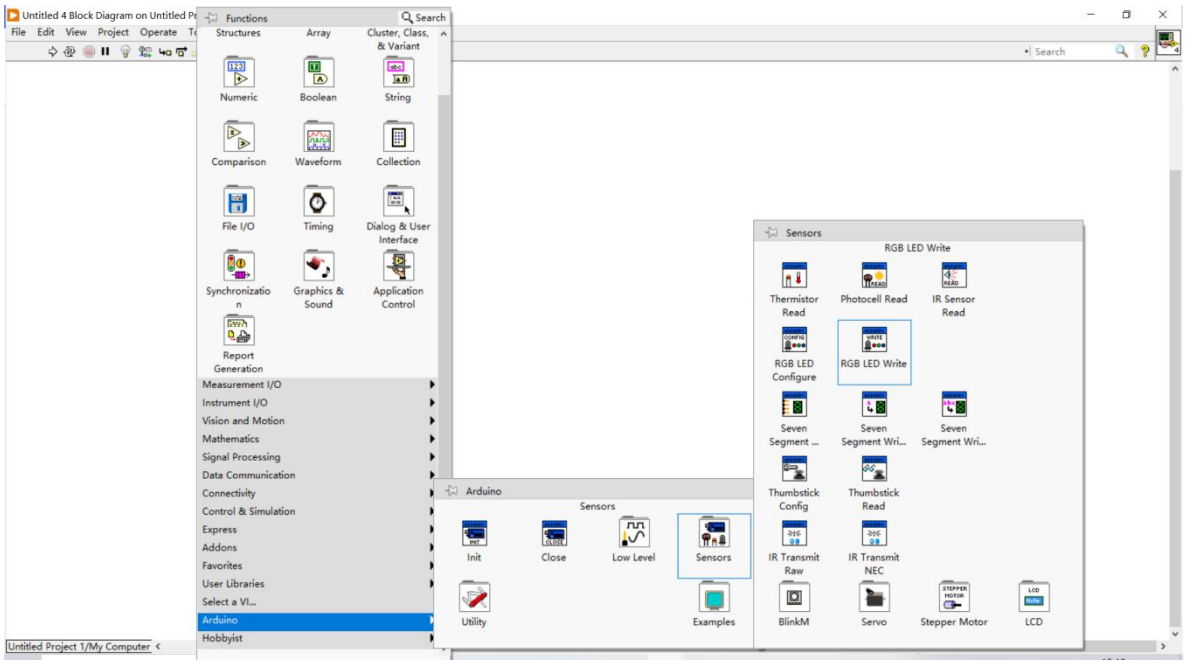
Parmi eux, **Arduino INIT** et **Arduino CLOSE** sont indispensables pour chaque programme, c'est-à-dire pour établir une connexion avec le contrôleur Arduino. Après avoir terminé l'opération prédéfinie sur l'Arduino, déconnectez la connexion avec le contrôleur Arduino pour libérer les ressources du port série du contrôleur Arduino.



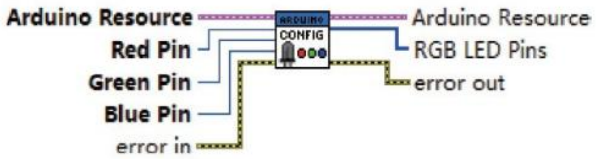
(Bibliothèque de fonctions)

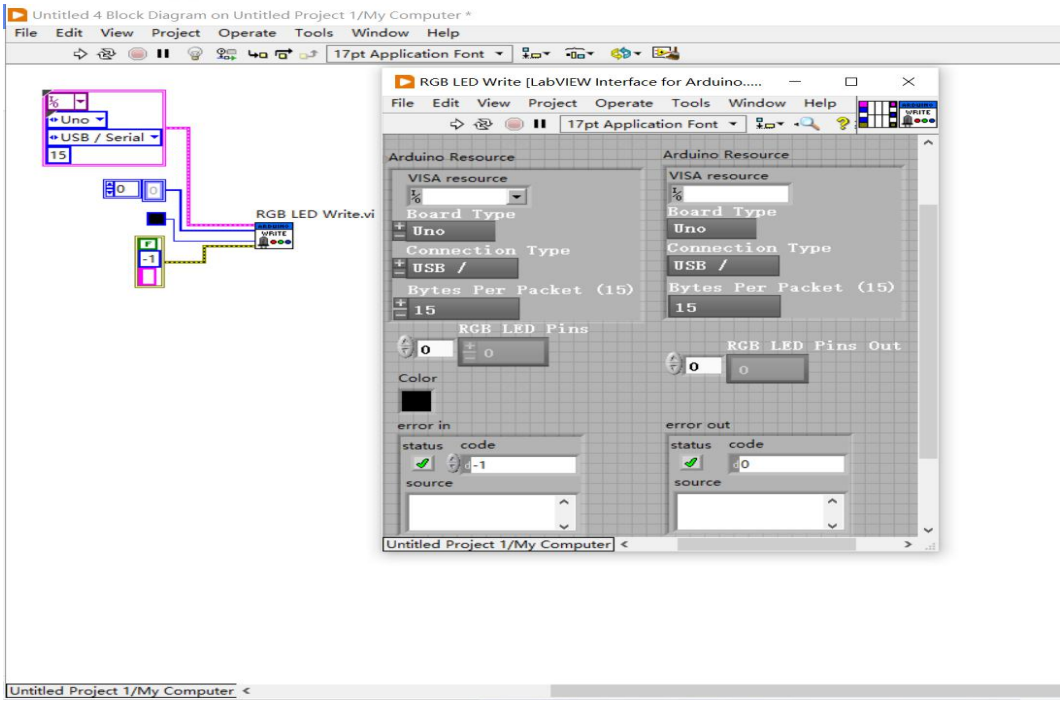
De plus, **Low Level** est l'application principale d'Arduino, **Sensors** est une bibliothèque de fonctions de capteur packagée. **Exemples** incluent un grand nombre d'exemples d'application. Étant donné que **Sensors** et **Exemples** sont des bibliothèques de fonctions construites en encapsulant **Low Level**, ils sont plus pratiques à utiliser, mais leur compatibilité et leur évolutivité ne sont pas aussi bonnes que **Low Level**.

Sensors:



Nous prenons la lampe à LED RVB comme exemple:





comme indiqué sur la figure, réalisons l'écriture de couleur de la lampe à LED RVB, le paramètre d'entrée est le tableau de broches RVB et la valeur de couleur, et le paramètre de sortie est le tableau de broches RVB. node peut être appelé plusieurs fois.Pour changer la couleur de la LED RVB.

Low Level

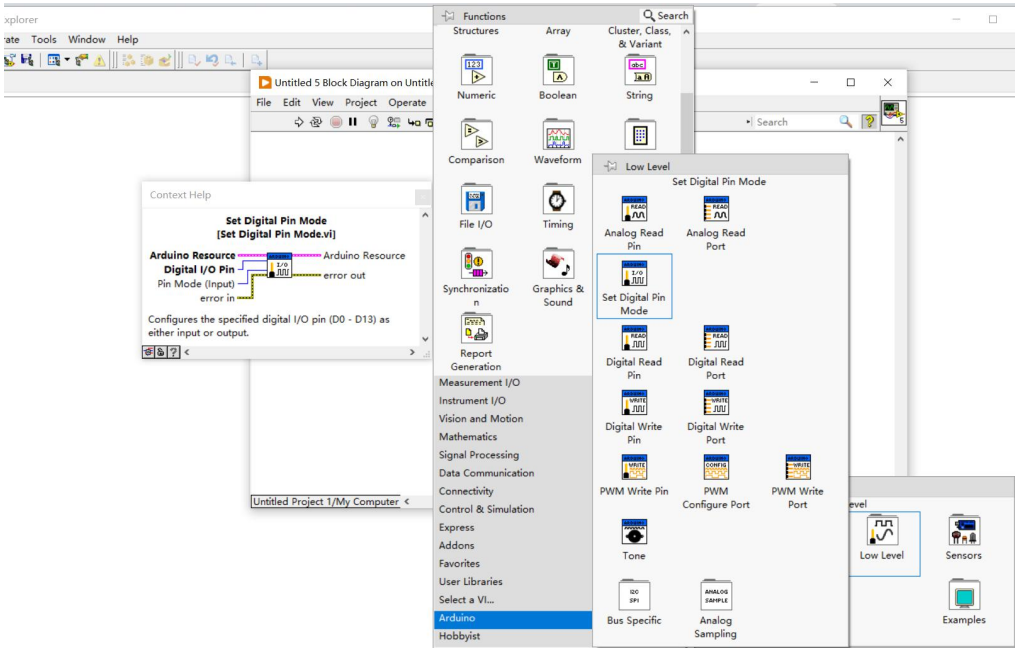
La bibliothèque de fonctions de bas niveau comprend une entrée analogique, une configuration d'I/O numériques, une entrée numérique, une sortie numérique, une sortie PWM (sortie analogique), Tone, I2C, SPI, un échantillonnage analogique, comme illustré à la Figure 3-10. Parmi eux, l'entrée analogique, la sortie numérique, la sortie numérique et la sortie PWM sont divisées en deux types : broche et port. Le fonctionnement du port convient aux situations qui nécessitent plusieurs broches d'I/Opour fonctionner.

La bibliothèque de fonctions d'I/O numériques dispose de 5 fonctions au total pour réaliser les opérations de base du port numérique, notamment définir Set Digital Pin Mode, Digital Read Pin、Digital Read Port、 Digital Write Port et Digital Write Port. Le nœud de fonction Set Digital Pin Mode est illustré à la Figure 3 11. Sa fonction est de configurer le port d'I/O sélectionné comme état d'entrée ou de sortie. Les paramètres d'entrée sont le numéro de broche d'I/O numérique (D0～D13)

et le mode de broche (entrée/sortie). Étant donné que l'entrée analogique et la sortie analogique (PWM) sont toutes deux des broches désignées, pour Arduino Uno, l'entrée analogique est A0 ~ A5 et la sortie analogique est 3, 5, 6, 9, 10 et 11. Ces fonctions spéciales peuvent déterminer la directionnalité, tandis que la directionnalité des I/O numériques ordinaires doit être configurée par l'utilisateur.

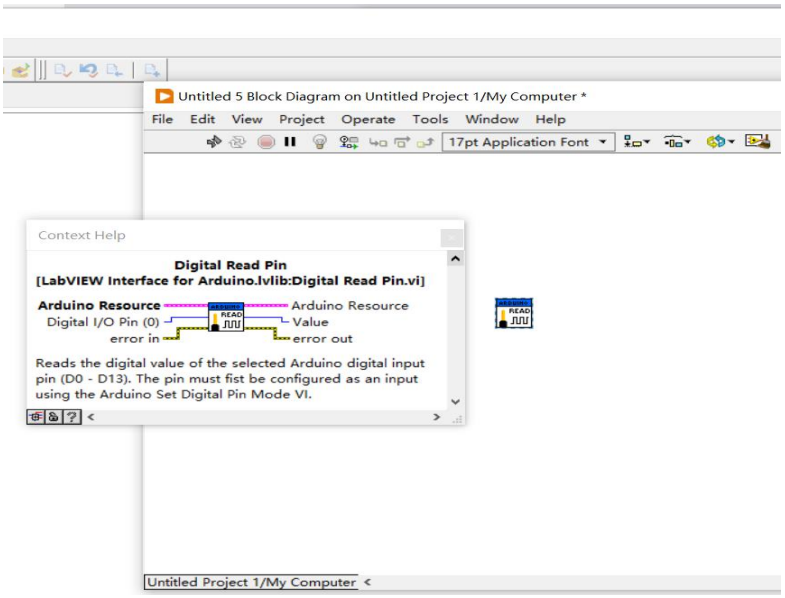
Par example:

Set Digital Pin Mode est de lire l'état logique (0 ou 1) de la broche spécifiée, le paramètre d'entrée est le numéro de broche numérique (D0 ~ D13) à lire, la valeur par défaut est D0 et la sortie est la valeur logique de la sélection épingler.



(Nœud de fonction de Set Digital Pin Mode)

Digital Read Pin consiste à lire l'état logique des broches D0 à D13 en séquence, sans spécifier le numéro de broche, la sortie est un tableau unidimensionnel et les éléments du tableau sont les valeurs logiques de D0 à D13 à leur tour.



(Nœud de fonction de Digital Read Pin)