Plus d'information

Raspberry Pi 3

Acheter

Q

Rechercher sur MCHobby - Wiki Page Discussion Lire Voir le texte source Historique

ARDUINO RASPBERRY-PI [:: PI Hacking ::] Particle.IO Micro Python Imprimante 3D

Actualités Modifications récentes Page au hasard Aide Outils Pages liées Suivi des pages liées Pages spéciales

Version imprimable

Adresse permanente

Information sur la

MCHobby investit du temps et de l'argent dans la réalisation de traduction et/ou documentation. C'est un travail long et fastidieux réalisé dans l'esprit Open-Source... donc gratuit et librement accessible. SI vous aimez nos traductions et documentations ALORS aidez nous à en produire plus en achetant vos produits chez MCHobby ... Sommaire [masquer] • Qu'est ce que Node-Red? 1 Intro Installer Node.js 2 Les noeuds d'entrée • Installer Node-Red 3 Les noeuds de sortie • Gestion de Node-Red 4 Les noeuds de fonction • Réaliser un premier diagramme 5 Les noeuds sociaux Noeuds de base 6 Noeud de stockage • Ajouter de nouveaux noeuds 7 Noeud d'analyse Exemples pratiques 8 Noeud avancés • PiCamRed: Email de PiCam

Intro

Voici une petite vue des nœuds déjà disponibles dans l'installation de base.

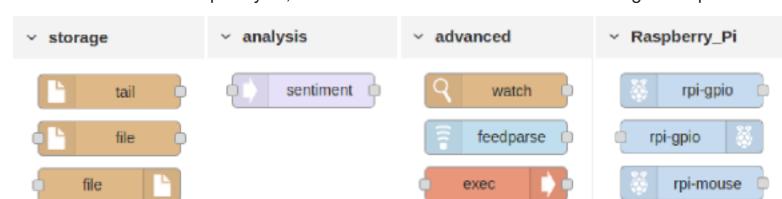
9 Les noeuds Raspberry-Pi

Rasp-Node-Red-Basic-Nodes

## Nodes



#### Sur une installation Raspberry-Pi, l'ensemble des noeuds dévolu au stockage sont plus limités que sur une machine normale.



### Les noeuds d'entrée

Vous les trouverez dans la collection de noeud "Input".

Il y a 7 noeuds d'entrée de base dans l'installation par défaut. Ces noeuds couvrent les mécanismes de communication de base utilisé principalement par les applications IoT. Cela s'étale des protocoles Internet fondamentaux tels que UDP et TCP jusqu'à un niveau plus élevé tels que HTTP et les mécanismes MQTT de publication/souscription (publish/subscribe).

Nom de Noeud	Description
inject	Injectez un timestamp (nombre de millisecondes écoulées depuis 1970) ou un texte utilisateur dans un message. Peut-être configuré pour injecter le texte manuellement, a intervalle régulier ou horaires spécifiques (en utilisant Cron).
catch	Capture les erreurs générée par les noeuds dans le même volet (la même feuille). Si un noeud lance une erreur pendant le traitement d'un message alors le flux s'arrête (fonctionnement typique). Ce noeud peut être utilisé pour capture ces erreurs retournant un message avec une propriété détaillant l'erreur, le type et le noeud ayant provoqué l'erreur.
mqtt	Faire une souscription sur un broker MQTT et "écoutant" un sujet particulier (topic), retourne toutes les données publiées sur le sujet (topic) comme un nouveau message. Supporte les niveaux de "Quality of Service" et rétention de la dernière donnée.
http	Réceptionne des requêtes HTTP permettant à Node-RED d'agir comme un serveur web rudimentaire. Le corps de la requête HTTP est délivrée comme message de sortie (ainsi qu'avec la réponse correspondante). Le message peut contenir une URL standard ou JSON.
websocket	Offre un point contact (endpoint) permettant d'établir une connexion websocket avec Node-RED. Offre une connexion duplex pour établir un canal de communication de type navigateur/serveur.
tcp	Utiliser pour accepter les requêtes TCP entrante sur le port spécifié (ou pour se connecter sur un port TCP distant). Génère des messages contenant les données TCP comme un simple – ou un flux/stream de – buffer, string ou string encodée en base64.
udp	Utilisé pour accepter les paquet UDP entrant (ou paquets multicast) sur le port spécifié. Génère un message contenant un les données UDP sous forme de BUFFER, string ou string encodée en base64.
serial in	Lit les données provenant d'un port série. Peut être configuré pour lire des buffers, durant une période de temps spécifique ou attendre un "line breaks".

## Les noeuds de sortie

Vous les trouverez dans la collection de noeud "Output".

Les noeuds de sorties (output en anglais) sont les pendants des noeuds d'entrée. Les noeuds de sortie offrent des moyens d'envoyer des données sur le même ensemble de protocoles (ex: mqtt, http, udp, etc).

Nom de Noeud	Description
debug	Offre une façon simple de voir des message qui sont affichés dans le volet "debug" (de débogage). Peut être configuré pour afficher le seul msg.payload ou l'objet msg en entier.
mqtt	Souscrit un broker MQTT et poste des données (msg.payload) qu'il reçoit en message entrant vers un sujet/topic MQTT. Supporte les niveaux de "Quality de Service" et la rétention de la dernière donnée.
http	Renvoi une réponse à une requête HTTP reçue sur le noeud d'entrée HTTP. Le corps de la réponse (body) est déterminé par msg.payload et peut définir le code du header et du statut.
websocket	Renvoi msg.payload sur le Websocket configuré. Si msgsession est définit alors renvoi l'information vers le client d'origine sinon fait un boardcast vers tous les clients connectés.
tcp	Répond sur le port TCP configuré. Peut également être utilisé pour envoyer des données sur un port spécifique.
udp	Envoyer un message UDP vers l'hôte et le port configuré (adresse ip address). Supporte le broadcast. Comme la plupart des noeuds, configuré par l'intermédiaire de l'interface utilisateur ou propriétés du message.
serial out	Envoi sur le port série configuré. Peut être configuré pour envoyer un retour à la ligne optionnel après les messages (message payload).
-	

# Les noeuds de fonction

Vous les trouverez dans la collection de noeuds "Function".

La catégorie "fonction" contient une une série de noeud supportant des fonctions de traitement. Cela va d'un simple noeud délai (delay), à un noeud d'aiguillage (switch), à noeud de fonction programmable qui peut être adapté à presque tous les besoins de programmation.

Nom de Noeud	Description
function	Noeud générique pour programmation de fonction. En utilisant du JavaScript standard, le noeud peut être personnalisé pour réaliser des tâches de traitement complexe sur les messages d'entrées et générant un ou plusieurs messages en sortie.
template	Configuré avec un modèle/template (en utilisant le format moustache) de complexité arbitraire. Ce noeud accepte un message d'entrée contenant des paires nom:valeur a insérer dans le modèle/template . C'est très pratique pour construire des messages, HTML, fichier config, etc.
delay	Un noeud générique pour retarder un message d'un temps donné (ou temps aléatoire). Peut également être configuré pour ralentir le flux des messages (ex: 10 ms par sec).
trigger	Créer deux messages de sortie séparé par intervalle de temps configurable lorsqu'un message d'entrée est reçu. Peut également être utilisé comme une horloge "watchdog".
comment	Un simple commentaire visuel configuré avec un titre et un corps.
http request	Permet de construire et envoyer une requête HTTP vers une URL spécifique. La méthode (PUT, GET, etc), "headers" et "payload" son tous configurable via l'interface ou par programmation.
tcp request	Un noeud pour effectuer une simple requête TCP. Il envoi le contenu msg.payload vers le port tcp et attend la réponse. Peut être configuré pour attendre la réponse (les données), attendre après un caractère spécifique ou termine immédiatement son exécution.
switch	Ce noeud aiguille (fait un routage) des messages sur base des propriétés du message. Les propriétés sont configurées en utilisant l'interface utilisateur et une variété de tests logiques (>, <, >= etc) appliqués aux propriétés du message.
change	Le noeud "change" peut être utilisé pour fixer, changer ou effacer (set, change, delete) les propriétés du message entrant. Une série de règles complexes autorise des modifications avancées incluant des recherches et remplacement dans le msg.payload
range	Noeud permettant d'échantillonner une valeur numérique en entrée pour la mapper vers une nouvelle sortie. Utilisé pour convertir ou borner les valeur (ex: la température). Non défini (Undefined) pour les données non numérique.

Nom de Noeud	Description
csv	Ce noeud fait un parsing de "msg.payload" et essaye de le convertis vers/depuis CSV. S'il reçoit une string CSV, il sort un objet JavaScript et s'il reçoit un objet JavaScript, il sort une string CSV.
html	Extrait les éléments d'un document html dans msg.payload en utilisant un sélecteur configurable (syntaxe de sélection CSS). Principalement utilisé pour parser un contenu HTML et retourner un tableau d'élément correspondant au filtre.
json	Ce noeud converti les informations depuis/vers un objet JSON. S'il reçoit un objet Javascript, il sort un contenu JSON. Et s'il reçoit un contenu JSON, il sort un objet JavaScript.
xml	Ce noeud converti les informations depuis/vers un format XML. S'il reçoit un objet JavaScript, il sort une string XML. S'il reçoit une string XML, il sort un objet JavaScript.
rbe	Noeud "Report By Exception". Génère un message seulement lorsque le message sur son entrée est différent de l'entrée précédente (string ou nombre) ou si l'entrée est modifiée d'une quantité configurable (mode "deadband", uniquement pour les valeurs numériques).

# Les noeuds sociaux

Vous les trouverez dans la collection de noeuds "Social".

Les noeuds "media sociaux" supporte les interactions avec les email et avec Twitter. Ces noeuds permettent d'envoyer ou recevoir des email, ou d'envoyer et recevoir des tweets.

Nom de Noeuc	Description
email in	Peut être pour se connecter répétitivement sur un serveur IMAP et retourner les nouveaux emails lorsqu'ils arrivent. Initialiser le sujet avec msg.topic et le corps du mail avec soit msg.payload ou msg.html s'il s'agit d'un email en HTML.
twitter in	Retourne les Tweets comme des messages. Peut être utilisé pour faire des recherches publiques ou sur des fils utilisateur contenant un terme de recherche configuré OU tous les tweets d'un utilisateur spécifique OU les messages reçu par un utilisateur authentifié.
email out	Envoi les messages entrants (dans le noeuds) comme un email via le serveur IMAP configuré. Le sujet et le destinataire sont configurable. Convertira des données binaires en fichier attaché.
twitter out	Tweets le msg.payload sur le compte configuré. Peut envoyé des messages directs et enverra des données binaires comme une image.

## Noeud de stockage Vous les trouverez dans la collection de noeuds "Storage".

Etant donné que nous utilisons une version pour Raspberry-Pi, le contenu de la section "stockage" est plus restreint que sur une installation normale.

Nom de	Description
Noeud	Description
tail	Effectue une opération "Tails" (ex: surveille les choses ajouté) au fichier configuré dans le noeud. (UNIQUEMENT Linux/Mac ONLY).  Ce noeud ne fonctionne pas sous Windows car il s'appuie sur la commande tail -F.
file in	Lit le fichier et envoi son contenu dans msg.payload et le nom de fichier dans msg.filename.  Le nom de fichier peut être configuré dans le noeud. S'il est laissé blanc, msg.filename doit être fixé par le message entrant
file	Ecrit le contenu de msg.payload dans le fichier spécifié, ex: créer un fichier de log. Le nom du fichier peut être configuré dans le noeud. Si laissé blanc, msg.filename sera fixé par le message entrant. Le contenu est ajouté en fin de fichier par défaut, il est possible de modifier cette fonctionnalité pour écraser le fichier à chaque fois; par exemple, pour générer une page web "statique"ou un rapport

## Noeud d'analyse Les noeuds d'analyse effecture des analyses standard sur les messages entrants. Seul le noeud "sentiment" est disponible dans l'installation par défaut. Le noeud sentiment peut être utilisé pour essayer de déterminer le sentiment en fonction des mots contenu

dans le message entrant, par exemple, un email ou tweet. Nom de

Noeud	Description
sentiment	Le noeud sentiment analyse le contenu de msg.payload et établit le niveau de sentiment du message sur base des mots qu'il contient. Il ajoute un objet msg.sentiment qui contient le résultat du "AFINN-111 sentiment score" dans la propriété msg.sentiment.score. Ce score évolue entre -5 et +5. A noter que ce test se base probablement sur un dictionnaire anglophone.
Noeud a	vancés

# Il existe également des noeuds offrant des fonctionnalités avancées.

Nom de Noeud	Description
watch	Surveille un répertoire ou fichier pour changement. Vous pouvez mentionner une liste de répertoire et/ou fichiers (séparé par des virgule). Vous aurez besoin de placer des guillemets "" autour des noms de fichier contenant des espaces. Sous Windows, vous devez utiliser des doubles \\ dans les noms de répertoire.  Le nom du fichier (nom complet) du fichier modifié est chargé dans msg.payload, tandis qu'une version "transformée en string" de la liste est retourné dans msg.topic.  msg.file contient le nom du fichier (nom cours) du fichier qui a changé. msg.type contient le type de changement, habituellement file ou directory (fichier ou répertoire) et msg.size contient la taille du fichier en octets/bytes.
feedparse	Ce noeud surveille les nouvelles entrées d'un fil RSS/atom et délivre les nouvelles entrées en tant que message. Il peut être configuré pour faire des relevés à intervale fix.
exec	Fait un appel à une commande système et fournit 3 sortie: stdout, stderr et le code de retour (return code). Utilise exec() par défaut, ce qui appelle la commande et bloque le processus en attendant la fin d'exécution du processus. Ensuite les résultats sont renvoyés en une fois, avec les erreurs.
Les noeuc	ds Raspberry-Pi

Nom de Noeud	Description
rpi_gpio in	Noeud d'entrée Raspberry Pi. Génère un 1 ou 0 dans msg.payload en fonction de l'état de broche d'entrée. Vous pourriez activer la résistance pullup ou pulldown sur l'entrée.  Le msg.topic est fixé sur pi/{the pin number}  Nécessite que la bibliothèque Python RPi.GPIO version 0.5.8 (ou plus récent) pour fonctionner.  Note: Nous utilisons les numéro de broche physique du connecteur P1 puisqu'elles sont plus facile à localiser.
rpi_gpio out	Noeud de sortie Raspberry Pi. Attend une valeur 0 ou 1 dans msg.payload (ou <i>true</i> ou <i>false</i> ). Fixera la broche sélectionnée au niveau logique "haut" ou "bas" en fonction de la valeur communiquée. La valeur initiale de la broche au moment du déploiement sera fixé à 0 ou 1. Pour une broche en mode PWM attend une valeur entre 0 et 100.  Nécessite que la bibliothèque Python RPi.GPIO version 0.5.8 (ou plus récent) pour fonctionner.
rpi_mouse	Noeud "souris Raspberry Pi". Génère un 1 ou 0 dans msg.payload lorsque le bouton de la souris est pressé et relâché. Fixe la valeur de msg.button sur la valeur: 1 = gauche/left, 2 = droite/right, 4 = milieu/middle, il est donc possible de déduire quel bouton (ou combinaison de bouton) ont été pressé.

Créé par C. Mobberley 

pour AdaFruit Industries. Traduction réalisée *et augmentée* par Meurisse D. pour MCHobby.be ☑.

Source: Raspberry Pi Hosting Node-Red ☑

Toute référence, mention ou extrait de cette traduction doit être explicitement accompagné du texte suivant : « Traduction par MCHobby.be 🗗) - Vente de kit et composants » avec un lien vers la source (donc cette page) et ce quelque soit le média utilisé. L'utilisation commercial de la traduction (texte) et/ou réalisation, même partielle, pourrait être soumis à redevance. Dans tous les cas de figures, vous devez également obtenir l'accord du(des) détenteur initial des droits. Celui de MC Hobby s'arrêtant au travail de traduction proprement dit.

Traduit avec l'autorisation d'AdaFruit Industries - Translated with the permission from Adafruit Industries - www.adafruit.com