Prototypage d'un capteur de température et d'humidité IoT WLAN solaire

Hans-Günter Kremser, Texas Instruments

Selon de nombreuses études, le futur marché des applications de l'Internet des objets (IoT), véritables produits d'avenir, pourrait peser des millions de dollars. De nombreux domaines tels que l'électronique portable, l'électroménager, la sécurité et la santé seront conquis par les appareils IoT.
Le présent article décrit comment mettre les données de capteurs sur le *cloud* pour qu'elles soient accessibles partout dans le monde. En outre, l'ensemble du système fonctionne à l'énergie solaire.
S'il est aisé de calculer la consommation énergétique de tout matériel, il est plus difficile de dimensionner la batterie et le panneau pour une application fonctionnant à l'énergie solaire. En outre, il convient de prendre en considération les temps de fonctionnement et d'arrêt de l'appareil. Les conditions d'ensoleillement à l'emplacement envisagé de l'équipement final étant difficiles, voire impossibles à prévoir, un capteur de température et d'humidité WLAN complet et autonome, avec contrôle des batteries, a été construit au moyen des seuls circuits d'évaluation TI.
Un panneau solaire 2,5 W polycristallin bon marché de 13 cm x 15 cm, une batterie de téléphone mobile Li-ion de 4,2 V et les circuits d'évaluation (EVM) suivants ont été utilisés pour réaliser ce prototype : CC3200EVM, TPL5110EVM, BQ25505EVM, HDC1010EVM et TPS82690EVM. En fonction du rapport cyclique requis et des conditions d'ensoleillement, il est possible de mesurer la consommation énergétique réelle, ce qui facilite le dimensionnement de la batterie et du panneau solaire. Grâce au temporisateur TPL5110, le courant de repos en période d'arrêt est inférieur à 400 nA, selon la fiche technique. Le temps de mise sous tension du convertisseur courant continu-courant continu, du CI du capteur et du module WLAN dépend de la connexion Internet, des capacités du routeur, de la charge du serveur, etc.

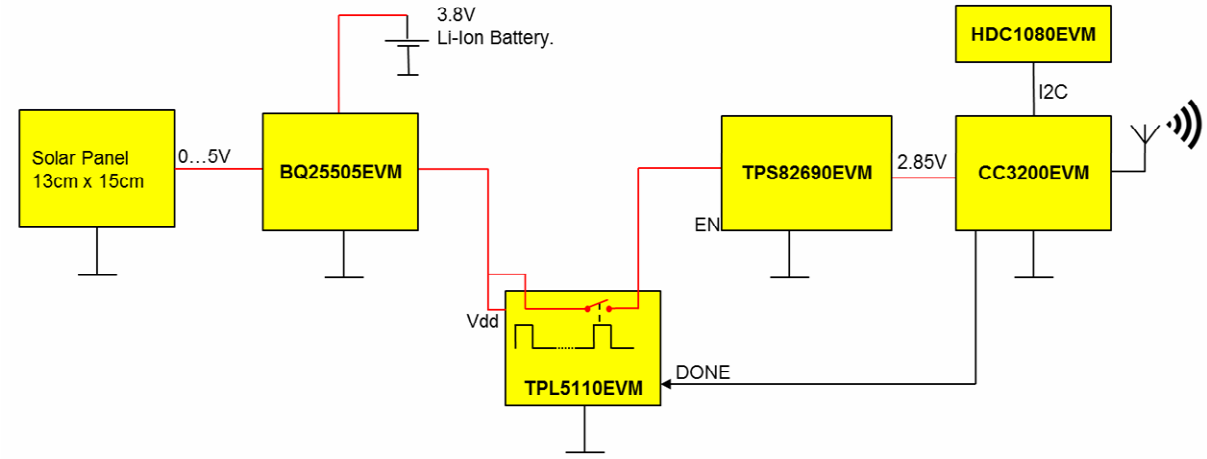


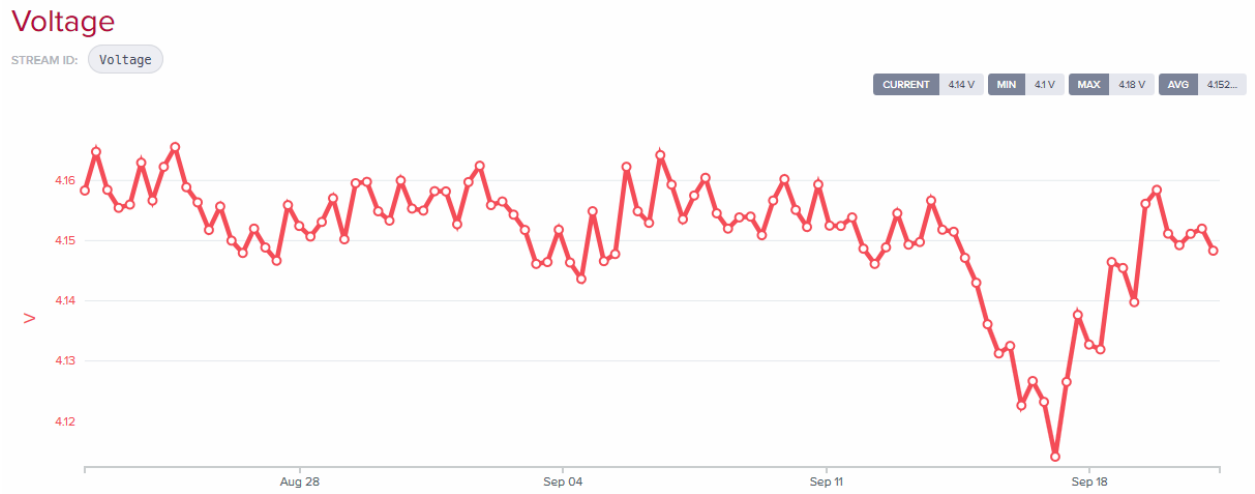
Schéma fonctionnel

Aperçu du système :

L'ensemble du contrôle de batterie, protection contre la surtension incluse, est commandé par le BQ25505EVM. Celui-ci reçoit la tension variable fournie par le panneau solaire et charge la batterie Li-ion avec le hacheur parallèle intégré en envoyant une tension d'environ 0,3 V si la batterie est complètement déchargée. Habituellement, la batterie est préchargée et la récupération d'énergie commence à environ 120 mV.
La pièce la plus importante du système quant à l'économie d'énergie est le TPL5110EVM qui est configuré pour fonctionner en mode temporisateur. Le potentiomètre embarqué permet des intervalles de temps compris entre 100 ms et 2 h. Le MOSFET de commutation d'alimentation a été remplacé par un transistor 1,3 A plus puissant en raison du module WLAN, qui consomme l'essentiel de l'énergie durant les transmissions. Le rapport cyclique a été défini à 10 min et la transmission prend environ 10 s. L'intégrale, sur une période de 10 s, entraîne une puissance requise de 2,13 Ws, tandis que la puissance de repos est évaluée sur la base des valeurs de la fiche technique à 2,06 mWs. Avec une batterie complètement chargée à 7,98 Wh, la durée de vie théorique est évaluée à 114 jours. Ceci indique que la batterie choisie est bien trop puissante, car la batterie est censée se charger durant la journée. Dans des conditions idéales, le système est alimenté par la batterie durant les périodes d'obscurité et elle est chargée dans la journée. Mais, à Munich par exemple, cela pourrait représenter moins de quatre heures en hiver.
En supposant un ensoleillement extérieur de 1 kW/m^2 (1) et un rendement estimé de 15 %, la puissance du panneau solaire peut être évaluée à environ 150 W x (0,15 m x 0,13 m) = 2,9 W, ce qui correspond presque

aux 2,5 W spécifiés pour le panneau utilisé.
Le TPS82690EVM est un simple module d'alimentation de convertisseur Buck destiné à générer les 2,85 V régulés à partir de la tension variable de la batterie pour le module WLAN CC3200EVM.
La température est mesurée avec une précision de ± 0,2 % et l'humidité avec une précision de 2 % avec le HDC1010EVM. Cet EVM est connecté au CC3200EVM via une interface I2C et également alimenté par le module Wi-Fi.
Un moniteur de batterie a été installé avec un diviseur de tension résistif à haute impédance directement connecté à la batterie, et la tension est mesurée avec le convertisseur analogique-numérique intégré du CC3200.
Le logiciel a été écrit avec la plateforme de développement Energia et tous les paramètres mesurés tels que la température, l'humidité et la tension de la batterie sont envoyés sur le *cloud* sur le site www.m2x.att.com une fois le CC3200 connecté à un réseau Wi-Fi. De nombreux autres services de *cloud* sont disponibles (6).
Un modèle de logiciel peut être trouvé aux adresses (2) et (3) et l'ensemble du code source est disponible à l'adresse (4).

Conclusion :
Une application IoT solaire complète peut être construite en connectant simplement des circuits d'évaluation et quelques composants externes.
Comme indiqué à la figure suivante, la tension de la batterie est relativement constante sur une période de 30 jours et la batterie est presque entièrement chargée. Une courbe descendante indique que la batterie va se décharger et que le rapport cyclique doit être réduit.



Tension de la batterie sur une période d'un mois

À propos de l'auteur :
Après des études d'ingénierie des communications à Cologne, Hans-Günter Kremser a travaillé pendant plusieurs années en tant qu'ingénieur développement et qu'ingénieur d'application terrain pour divers fabricants de semi-conducteurs. Aujourd'hui, il est ingénieur d'application terrain en chef pour les produits analogiques chez Texas Instruments.

Références :

(1) https://de.wikipedia.org/wiki/Sonnenenergie
(2) http://processors.wiki.ti.com/index.php/CC3200_DEMOS#_AT.26T_M2X
(3) http://energia.nu/pin-maps/guide_cc3200launchpad/
(4) https://e2e.ti.com/support/wireless_connectivity/simplelink_wifi_cc31xx_cc32xx/f/968/t/543241
(5) http://www.ti.com/ww/en/internet_of_things/iot-cloudsolution.html