

Génie informatique

Session S5

Unité APP 3

Semaines 5 et 6

SYSTÈMES EMBARQUÉS RÉSEAUTÉS

Département de génie électrique et de génie informatique

Faculté de génie

Université de Sherbrooke

Automne 2017

Note : En vue d'alléger le texte, le masculin est utilisé pour désigner les femmes et les hommes.

Document Guide-Etudiant.docx

Version 2.1, 6 septembre, 2017

Révisé par Jean Lavoie

Rédigé par Philippe Mabillean

Révisé par Arnaud Marchese

Copyright © 2008-2017 Département de génie électrique et de génie informatique,
Université de Sherbrooke.

Table des matières

1. Éléments de compétences visés	5
GIF-550 Systèmes embarqués réseautés.....	5
2. ÉNONCÉ DE LA PROBLÉMATIQUE.....	6
3. Connaissances nouvelles à acquérir par la résolution de cette problématique	9
<i>Connaissances déclaratives : QUOI</i>	9
<i>Connaissances procédurales: COMMENT</i>	9
<i>Connaissances conditionnelles: QUAND</i>	9
4. Références.....	10
Références essentielles à consulter	10
Ressources sur le WEB.....	10
Logiciels utilisés	10
Autres références	10
Notes de lecture.....	10
5. Sommaire des activités liées à la problématique	11
Activités de la semaine 1	11
Activités de la semaine 2	11
6. Productions à remettre	12
Validation du fonctionnement de la problématique.....	12
Le fonctionnement du code sera évalué uniquement pendant la période de validation. Toute modification portée ultérieurement au code ne sera pas évaluée (sauf l'ajout de commentaires). Référez-vous à la section portant sur la validation pratique pour plus de détail.....	12
Fichier contenant le code de la solution développée à remettre lors de la séance de validation de la solution.....	12
Rapport d'APP (à remettre à la fin de la deuxième rencontre de tutorat).....	13
7. Semaine 1 - Formation à la pratique procédurale – première séance	13
But de l'activité.....	13
Activités	13
8. Semaine 1 - Formation à la pratique en laboratoire.....	15
But de l'activité.....	15
Montage et test de la communication avec le module X-Bee.....	16
9. Semaine 1 - Formation à la pratique procédurale – seconde séance.....	19
But de l'activité.....	19

Activités	19
10. Semaine 2 - Validation pratique de la solution en laboratoire	20
L'activité de laboratoire	20
Validation de la solution pratique	20
11. Semaine 2 - Travail personnel de rédaction des productions exigées	21
12. Semaine 2 - Seconde rencontre de tutorat: validation des connaissances acquises	21
13. Rapport d'APP	22
Rapport d'APP à remettre avant 08 :30 du matin le 11 février (1 par équipe)	22
Code de votre solution à remettre avant 08 :30 du matin le 11 février (1 par équipe)	22
Modalités d'évaluation du rapport d'APP	22
14. Évaluation sommative	24
15. Évaluation de l'unité	24

1. Éléments de compétences visés

GIF-550 Systèmes embarqués réseautés

1. Concevoir, développer et mettre au point une application sur un système embarqué exploitant la connectivité à un réseau informatique.

2. ÉNONCÉ DE LA PROBLÉMATIQUE

Vous avez obtenu un contrat de l'équipe de recherche du laboratoire DOMUS de l'Université de Sherbrooke pour participer au développement d'une application d'informatique diffuse construite autour d'un réseau de capteurs disposés dans un appartement expérimental. Les capteurs seront utilisés pour reconnaître les différentes activités qui se déroulent dans l'appartement : préparation des repas, soins personnels, loisirs... Cette reconnaissance d'activités sera ultérieurement intégrée à des systèmes d'aide aux occupants, en particulier dans le cas de personnes âgées ou de personnes handicapées.

Parmi les capteurs on va retrouver des détecteurs de mouvement à infrarouge, des contacts de portes, des tapis tactiles... La plupart de ces capteurs se traduisent par un contact qui est ouvert ou fermé (généralement appelé « contact sec »). D'autres capteurs, comme des détecteurs de température par exemple, vont être interfacés en utilisant un bus comme l'I2C ou le SPI.

L'objectif du système à développer est la collecte de l'information d'état de ces capteurs et son acheminement vers le logiciel de reconnaissance d'activités qui s'exécute sur un serveur. La plateforme utilisée pour réaliser un premier prototype des nœuds qui vont constituer le réseau de capteurs intelligents est une carte de développement Mbed. Chaque nœud est donc composé d'un microcontrôleur Mbed associé à un module de communication sans fil XBee.

Votre mandat dans le cadre de ce premier contrat est le suivant :

Développer un réseau ZigBee autour d'un nœud coordinateur de réseau de type Personal Area Network (PAN) associé à un ou plusieurs nœuds capteurs. Les modules Xbee seront utilisés en mode API.

Le nœud coordinateur devra également convertir les données reçues dans une chaîne de caractères qui devra en permettre une lecture et une identification aisée. Ensuite, la chaîne sera envoyée à un serveur d'application en utilisant le protocole WebSocket via un réseau Ethernet IP. Un fichier de configuration du nœud coordinateur devra comporter des informations comme l'identification du réseau PAN formé et l'URL du serveur à rejoindre pour envoyer les informations collectées.

Les nœuds capteurs devront effectuer la lecture d'un groupe de capteurs dont la spécification devra être flexible (contacts secs, senseurs interfacés sur différents bus etc...). Cette flexibilité devra se traduire par une architecture du logiciel qui en permet une reconfiguration facile. Il est ainsi requis d'utiliser des pointeurs de fonction pour la lecture et d'appeler les différentes fonctions à travers un tableau de fonctions. On ne demande pas que cette configuration soit dynamique ; le code peut être recompilé pour ajouter une nouvelle fonctionnalité (nouveau type de capteur). Pour aider la maintenance, un fichier de configuration devra contenir des paramètres de l'application comme l'identification du réseau PAN à rejoindre ainsi que la période de lecture des capteurs (exprimée en secondes).

En plus des indications précédentes, votre système doit simplifier la configuration des nœuds. Ainsi, les adresses 64bits des nœuds ne doivent pas être codées dans vos logiciels (Coordinateur et Routeur). Vous devez implanter des mécanismes pour les découvrir dynamiquement.

Dans tout système de télécom, il faut implanter des routines pour détecter et agir lorsqu'il y a des situations d'erreurs. Le code des nœuds doit montrer cette détection et implanter un mécanisme de correction. De plus, il vous est demandé d'ajouter une DEL sur chaque nœud et l'allumer pendant environ une seconde lorsqu'une erreur est détectée. Pour permettre des extensions futures des systèmes, il vous est aussi demandé de réaliser des communications bidirectionnelles entre le coordonnateur et les nœuds capteurs. Pour ceci, on vous demande de faire allumer une DEL, reliée à un des ports du XBee (DIO) des nœuds capteurs, en utilisant une commande de configuration à distance provenant du Coordinateur (Remote AT Commands). Cette DEL devra « clignoter » à une fréquence de 1Hz et être sous le contrôle du nœud coordonnateur seulement (ne pas utiliser le mode clignotement automatique du XBee).

Vous ne pourrez pas développer des classes lors de la résolution de la problématique (tout doit être procédural). Vous pouvez toutefois utiliser les classes externes de Mbed (e.g. Serial, etc.). Il est interdit d'utiliser une librairie externe pour communiquer avec les modules XBee (vous devez développer votre propre librairie).

Pour valider votre développement, vous devrez présenter un montage comportant au moins deux nœuds :

- a) Un nœud coordinateur du réseau sans fil et interconnecté via Ethernet/IP au serveur d'application
- b) Des nœuds capteurs intégrés à ce réseau sans-fil et faisant l'acquisition de capteurs dont au moins :
 - a. Un accéléromètre 3 axes
 - b. Un contact sec (tout ou rien)
- c) Vous devez préparer un plan de « tests » de votre produit. Ce qui sous-entend de développer une stratégie pour bien montrer votre implantation de toutes les fonctionnalités de la problématique.

Les données collectées devront éventuellement s'afficher sur une page WEB associée au serveur d'application. La communication entre le nœud coordinateur et le serveur d'application devra utiliser les WebSockets. Pour votre démonstration et vos expérimentations, on vous suggère d'installer et d'utiliser le serveur (en Python) suivant : <https://github.com/dpallot/simple-websocket-server>.

3. Connaissances nouvelles à acquérir par la résolution de cette problématique

Connaissances déclaratives : QUOI

Sciences de l'ingénierie

- Interconnexion d'un microcontrôleur à un dispositif périphérique
- Modèle stratifié de l'architecture réseau
- Réseaux sans fil
- Topologie
- 802.11, 802.15.4 et ZigBee
- Association
- Noeuds coordinateurs, routeur et dispositif de fin
- Dispositifs d'interface réseau
- Interface de programmation avec une pile de protocoles (API)
- Mécanismes d'envoi et de réception de trames
- Adressage et mécanismes d'association d'adresses
- Entités et adressage au niveau application
- Architecture interne d'une pile de protocoles TCP/IP
- Intégration d'un système embarqué à un réseau
- Configuration, tests

Connaissances procédurales: COMMENT

- Programmer un sous-système d'entrée/sortie d'un microcontrôleur pour exploiter un circuit périphérique
- Exploiter un dispositif d'interface avec un réseau
- Développer une application embarquée utilisant une pile de protocoles pour accéder à un réseau informatique
- Intégrer une pile de protocoles à un système embarqué
- Configurer un système embarqué pour l'intégrer à un réseau

Connaissances conditionnelles: QUAND

- Définir la technologie d'interconnexion adéquate pour relier un système périphérique d'acquisition ou de commande à un système embarqué
- Choisir une pile de protocoles pour intégrer un système embarqué à un réseau informatique
- Définir l'architecture de communication permettant de relier un système embarqué à un réseau informatique

4. Références

Références essentielles à consulter

- XBee®/XBee-PRO® ZB RF Modules
- IEEE Standard 802.15.4

Ressources sur le WEB

- Site mbed.org
- Site de Digi <http://www.digi.com/>

Logiciels utilisés

- Logiciel X-CTU (Digi)
- Compilateur C++ en ligne sur le site mbed.org

Autres références

- Andrew Tanenbaum, "Réseaux", Pearson Education, 4e édition, 920 pages, 2003, ISBN 2-7440-7001-7
- "Real-Time Concepts for Embedded Systems" by Qing Li, Caroline Yao (Contributor), CMP Books; 1st edition (July 2003), ISBN: 1-57820-124-1
- "µC/TCP-IP: The Embedded Protocol Stack for the STMicroelectronics STM32F107", Christian Légaré, Micrium Press.
- "TCP/IP Lean", 2nd edition, Jeremy Bentham, CMP Books 2002, ISBN 1-57820-108-X

Notes de lecture

XBee®/XBee-PRO® ZB RF Modules Manual

Chapitres 2 à 4, 6 et 9 (82 pages)

IEEE Standard 802.15.4

Chapitre 5 et Chapitre 7 (en partie)

5. Sommaire des activités liées à la problématique

Activités de la semaine 1

- 1^{ère} rencontre de tutorat
- Étude personnelle et exercices: étude des sujets issus des objectifs d'étude du tutorat
- Formation à la pratique en laboratoire
- Formation à la pratique procédurale
- Étude personnelle
- Formation à la pratique procédurale

Activités de la semaine 2

- Étude personnelle et exercices: sujets issus des objectifs d'étude du tutorat
- Formation à la pratique en laboratoire: validation pratique des éléments de solution
- Rédaction du document d'APP
- 2^{ième} rencontre de tutorat: validation des connaissances acquises
- Remise du document d'APP et des schémas
- Évaluation formative
- Évaluation sommative

6. Productions à remettre

Validation du fonctionnement de la problématique

Le fonctionnement du code sera évalué uniquement pendant la période de validation. Toute modification portée ultérieurement au code ne sera pas évaluée. Réferez-vous à la section portant sur la validation pratique pour plus de détail.

Fichier contenant le code de la solution développée à remettre avant la séance de validation de la solution

Ce fichier doit être en format **zip** et contenir les fichiers nécessaires à la construction de l'application de démonstration demandée. Les sources de tous les fichiers développés ou modifiés par vous devront être fournies (donc les deux projets pour les modules Mbed du nœud coordinateur et du nœud capteur) et les fichiers de configuration créés. En résumé, seulement les fichiers requis pour compiler et utiliser votre code (exemples .c, .h, etc.) sans les bibliothèques disponibles via l'environnement Mbed. Le code doit être commenté et tous les fichiers doivent être identifiés dans leur entête (mais pas les fichiers de configuration). Par commentaires, on entend :

- 1- Un entête de fichier qui explique globalement le contenu du fichier et donne le nom des auteurs
- 2- Pour chaque fonction, une brève description de l'opération ou algorithme, la liste des entrées et sorties (incluant leurs fonctions, les limites ou valeurs permises/attendues) et les conditions/comportement en situation d'erreurs.
- 3- Tous commentaires dans le code permettant de mieux comprendre ce que vous avez codé

La bibliothèque développée pour les échanges de messages entre le Mbed et le XBee doit présenter une utilisation structurée de l'API et tirer parti de ses caractéristiques.

Les fichiers seront remis pour chaque groupe ayant travaillé sur le même réseau. Regroupez les fichiers dans un .ZIP avec vos CIP selon le format « aaaaxxxx-bbbbyyyy.zip ». Déposez-le avant le début de la période de validation en utilisant la fonctionnalité de dépôt des travaux sur le site WEB du département (dépôt > ... > s5info > APP3_Code). Tout retard sera pénalisé de 20% par jour.

Rapport d'APP (à remettre avant la deuxième rencontre de tutorat)

Le rapport doit comporter la description de votre solution avec en particulier une présentation de l'architecture complète du système, les spécifications des paramètres et des messages et la flexibilité dans la programmation du logiciel à l'ajout de nœuds capteurs. Celui-ci doit comporter un ou des schémas ou diagrammes. Ce document ne doit pas dépasser cinq pages de format 8 ½ par 11 en plus de la page titre. Déposez ce document en format PDF avant le tutorat 2 à 08:30 AM en utilisant la fonctionnalité de dépôt des travaux sur le site WEB du département (dépôt > ... > s5info > APP3_Rapport). Il doit être nommé aaaaxxxx-bbbbxxxx.pdf où aaaaxxxx et bbbbyyyy sont vos CIP. Tout retard sera pénalisé de 20% par jour.

 *Inscrire votre nom et votre CIP dans chaque document remis.*

7. Semaine 1 - Formation à la pratique procédurale – première séance

But de l'activité

La première séance de pratique procédurale a pour objectif une familiarisation avec les protocoles 802.15.4 et ZigBee ainsi que leur mise en œuvre sur les modules X-Bee.

Activités

À partir de la lecture du document IEEE Standard 802.15.4 et du manuel des modules X-Bee, répondre aux questions suivantes :

- 1- À quel protocole est-ce que les couches physiques et MAC utilisées par les modules X-Bee appartiennent?
- 2- Comment se déroulent les échanges d'information sur un réseau 802.15.4?
- 3- Mécanisme d'accès au média :
 - a. Comment fonctionne le mécanisme CSMA/CD ?
 - b. Comment fonctionne le mécanisme CSMA/CA ?
 - c. Pourquoi un mécanisme d'accès au médium comme le CSMA/CD ne peut pas être utilisé ici? Donner un exemple.

- d. Qu'est-ce que la segmentation temporelle (Slotted CSMA/CA) ? Quel est son avantage?
- e. Est-ce que le CSMA/CA est utilisé sur tous les types d'échange ?

4- Mécanisme d'adressage :

- a. Comment fonctionne le mécanisme d'adressage utilisé avec les modules X-Bee ?
- b. Comment est établi le parcours des paquets dans le réseau?
- c. Quels sont les algorithmes de routage présents dans les réseaux ZigBee ?
- d. Comment sont-ils mis en œuvre et activés avec les modules X-Bee ?
- e.

5- Dans le contexte de l'adressage au niveau de l'application (APS) :

- a. Préciser les notions suivantes : « EndPoint », « Cluster ID » et « Application Profile ».
- b. Donner un exemple de l'utilisation de ces différents champs pour une classe d'applications donnée.
- c. Quelle est la couche de protocole spécifique à ZigBee qui permet cette valeur ajoutée à l'adressage ?
- d. Existe-t-elle dans 802.15.4 ?

8. Semaine 1 - Formation à la pratique en laboratoire

Préparation

Pour accélérer le déroulement du laboratoire, veuillez vous assurer d'avoir fait les choses suivantes avant d'entrer en laboratoire:

- Installer le pilote qui permet d'émuler des ports série sur vos lignes USB :
<http://www.ftdichip.com/Drivers/VCP.htm>
- Installez le logiciel XCTU : <http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/xctu>
- Idéalement, effectuez le montage avant le laboratoire (ça prend au plus 15 min.)
- Il sera utile d'installer un logiciel du genre Tera Term
(<https://tssh2.osdn.jp/index.html.en>) pour effectuer les tests et le déverminage.

But de l'activité

Cette séance de pratique en laboratoire vise l'exploration de l'utilisation du module X-Bee proposé pour la mise en œuvre de la solution.

Attention : Les modules utilisés sont fragiles et vous devez les manipuler avec précaution. En particulier lorsque vous enfichez ou retirez les modules du breadboard l'alimentation doit être coupée. Ne pas oublier qu'avec les modules Mbed l'alimentation arrive à votre montage via le bus USB; elle est donc présente lorsque votre montage est connecté au PC.

Pour le laboratoire et pour la résolution de la problématique, vous allez travailler par équipe de 2 étudiants avec 1 coffre.

Montage et test de la communication avec le module X-Bee

Faire un montage sur un premier breadboard interconnectant un module X-Bee et un module Mbed en utilisant les signaux suivants (les numéros des broches sont entre parenthèses) :

Mbed	X-Bee
3,3 v (40)	Vcc (1)
GND (1)	GND (10)
P13 (13)	DIN (3)
P14 (14)	DOUT (2)
P8 (8)	RESET (5)

Tableau 1: Branchements entre Mbed et Xbee

Note : Vous pouvez ajouter une DEL avec la résistance adéquate sur la sortie AD5/DIO5 (15) du X-Bee pour faciliter la compréhension de l'état du module.

Écrire un logiciel qui effectue les opérations suivantes :

- Initialiser la broche P8 comme une sortie numérique
- Initialiser le port de communication série avec le PC
- Initialiser un port de communication série associée aux broches p13 et p14
- Mettre à 0 la broche RESET du module X-Bee pour 400 mS
- Entrer dans une boucle sans fin qui effectue les opérations suivantes :
 - Scrute le port COM associé au PC et renvoie le caractère reçu directement au PC

Toutes les communications série s'effectuent à 9600 bauds, 8 bits de donnée, pas de parité et un bit d'arrêt (8,N,1). Avec ce programme, vous devriez pouvoir ouvrir une fenêtre de terminal COM (dans TeraTerm par exemple) et lorsque vous entrez des caractères au clavier ceux-ci sont retranscrits dans la fenêtre de terminal. S'il n'y a pas de retranscription, cela veut dire que vous avez mal redirigé le flux d'information. Si cette étape fonctionne ou que vous êtes confiant par rapport à votre montage et votre compréhension de ce que vous êtes en train de faire, procéder à modifier votre programme ainsi :

- Initialiser la broche P8 comme une sortie numérique
- Initialiser le port de communication série avec le PC
- Initialiser un port de communication série associée aux broches p13 et p14
- Mettre à 0 la broche RESET du module X-Bee pour 400 mS

- Entrer dans une boucle sans fin qui effectue les opérations suivantes :
 - scrute le port UART associé aux broches p13 et p14 et si un caractère y est reçu l'envoyer en écho vers le port série attaché au PC
 - Scrute le port attaché au PC et si un caractère y est reçu l'envoyer en écho vers le port série associé aux broches p13 et p14.

Toutes les communications série s'effectuent à 9600 bauds, 8 bits de donnée, pas de parité et un bit d'arrêt (8,N,1). On peut allumer une D.E.L. du module Mbed pendant la transmission des caractères afin d'en avoir une trace visuelle.

Si ce n'est pas déjà fait, allez chercher le logiciel de configuration XCTU distribué par Digi:

<http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/xctu>

En utilisant X-CTU, vous pourrez communiquer avec le module X-Bee **via** le module Mbed et le logiciel que vous venez de développer (autrement dit, le Mbed sert de pont entre le Xbee et votre ordinateur). Ajoutez un module X-Bee à l'aide du bouton «Add devices ».

Ensuite, utilisez l'onglet « Configuration » pour lire la configuration et la version du progiciel (« Firmware ») installé sur le module X-Bee.

Pour le nœud « End Device », noté avec un R sur le module X-Bee, le « Function set » doit être Zigbee Router API. Ce module est configuré en mode Router pour faciliter la communication durant l'APP. Les modules en mode « End Device » peuvent tomber en veille, ce qui rend la tâche de communication plus difficile.

Pour le nœud Coordinateur identifié par la lettre C, le « Function Set » doit être Zigbee Coordinator API. **XB24-ZB 23A7**

Réalisez un second montage sur un autre breadboard interconnectant un autre module Mbed avec le second module X-Bee qui vous a été fourni.

Prévoir la disposition des deux breadboards que vous allez utiliser en fonction du fait que, pour la solution de la problématique, l'un va devenir le nœud capteur donc va accueillir l'accéléromètre ainsi qu'un bouton simulant un contact sec et que l'autre va devenir le nœud coordinateur et va accueillir l'interface Ethernet.

Répéter les mêmes opérations jusqu'à obtention de la configuration et de la version du progiciel installé sur le second module X-Bee.

Une fois ces étapes réalisées pour les deux modules X-Bee, configurer ceux-ci afin qu'ils puissent communiquer :

Choisir un identificateur de réseau PAN unique (l'inscrire au tableau) et programmer le dans les deux modules X-Bee en utilisant la commande « Write » de l'onglet « Configuration ».

Ensuite, pratiquer l'envoi et la réception des trames API avec un module X-Bee avec l'onglet « Terminal » du logiciel X-CTU. Les trames peuvent être construites à l'aide de la section « Send frames » de l'onglet « Terminal ».

En particulier, on validera l'association d'un nœud à un réseau, le changement de PAN ID, l'envoi et la réception de données.

Montage du connecteur Ethernet

Effectuer le montage du connecteur Ethernet sur votre breadboard. Utiliser le forum sur Mbed ou la documentation sur Sparkfun pour déterminer les bonnes connexions. Voici les numéros de série du connecteur et du PCB :

- Magjack de Sparkfun PRT-08534
- PCB de Sparkfun PRT-08790

9. Semaine 1 - Formation à la pratique procédurale – seconde séance

But de l'activité

La seconde séance de pratique procédurale a pour objectif de poursuivre la compréhension de différents protocoles de communication et de perfectionner les connaissances en programmation de systèmes embarqués.

Activités

- 1- Modèle stratifié de la pile de protocoles ZigBee.
 - a. Représenter les différentes couches présentes dans les modules X-Bee.
 - b. Expliquer le rôle et les fonctions des différentes couches de la pile de protocoles ZigBee.
 - c. Mettre en relation avec celles du modèle OSI.
 - d. Retrouve-t-on les mêmes couches dans tous les types de nœuds ?
- 2- Décrire sommairement le fonctionnement du protocole TCP et UDP, en résumant les différences entre chacun.
- 3- Décrire sommairement le fonctionnement du protocole WebSocket.
 - a. Quelle est son utilité?
 - b. Placer ces protocoles (HTTP, WebSocket) sur le diagramme stratifié de la pile de protocoles.
- 4- Expliquer le fonctionnement d'un pointeur de fonction.
 - a. Comment le déclarer, l'initialiser et l'appeler?
 - b. À quel endroit pourraient être utilisés les pointeurs de fonction pour résoudre la problématique?
 - c. Écrire le code d'un exemple d'utilisation.

10. Semaine 2 - Validation pratique de la solution en laboratoire

L'activité en laboratoire

Installer et tester la solution que vous avez développée. Afin de démontrer un réseau avec plus que deux nœuds. Le partage de nœuds capteurs entre plusieurs équipes est encouragé. Ceci sera encore plus valorisé si le code du nœud capteur n'a pas à être modifié (mise en évidence de la flexibilité du nœud capteur). L'évaluation de la solution se fait durant cette période.

À la correction du rapport, seulement les points suivants seront évalués à l'aide des fichiers du code:

- les commentaires,
- la conception d'un format de message structuré,
- la détection de plagiat.

Validation de la solution pratique

Vous devez développer une stratégie de test pour démontrer à l'évaluateur l'accomplissement de tous les requis de départ.

Évalué par le formateur (les activités suivantes devraient être démontrées dans le plan/stratégie de tests) :

- Qualité du montage
- Qualité de la démonstration (plan de tests)
- L'état de développement de votre solution.
 - Présentation sommaire de la solution, de son architecture et de ses composantes.
 - Flexibilité de la solution (ajout/retrait de capteurs)
- Les résultats de tests montrant la fonctionnalité de votre solution.
 - Fichiers de configuration (démontrer à l'aide de changements des valeurs)
 - PAN ID
 - URL
 - Période
 - Transmission WebSocket

- Valeurs de l'accéléromètre
- État du bouton
- Conversion en caractères ASCII
- Réception WebSocket
- Perte de connectivité
- Del sur xBee routeur
- Génération d'une Condition d'erreur
 - Del d'erreurs sur Coordinateur
 - Del d'erreurs sur Routeur

Retour sur les difficultés rencontrées et les apprentissages réalisés.

Les étudiants absents auront la note de 0/40 pour l'évaluation de la fonctionnalité du code.

11. Semaine 2 - Travail personnel de rédaction des productions exigées

Rédaction du rapport de l'APP suivant les directives données à la section 13. Finalisation du schéma de concept.

12. Semaine 2 - Seconde rencontre de tutorat: validation des connaissances acquises

Validation des connaissances - Bilan de groupe - Travail personnel de synthèse et d'études correctives.


13. Rapport d'APP

Rapport d'APP à remettre avant 08 :30 le jour du tutorat 2 (1 par équipe) en format PDF.

Maximum de 5 pages.

Code « commenté » de votre solution à remettre avant la période de validation (1 par équipe).

Seulement les fichiers requis pour compiler votre code (exemples .c, .h, etc.) sans les librairies disponibles via l'environnement. Le tout dans un fichier de type .zip.

 Inscrivez votre nom et votre CIP sur chaque document remis (y-compris les fichiers sur le serveur).

Modalités d'évaluation du rapport d'APP

L'évaluation du rapport d'APP contribue à l'évaluation des éléments de compétences de l'unité (voir section 15). On évalue l'exactitude, la précision, la complétude, la valeur de chaque élément de la solution. Il faut démontrer à l'évaluateur que vous avez assimilé tous les concepts durant l'APP. Après la lecture du rapport, l'évaluateur doit être en mesure d'effectuer la maintenance du système sans problème.

- Schémas détaillés de l'architecture des sous-systèmes et de l'ensemble du système
- Schéma détaillé de la topologie du réseau PAN avec paramètres pour chaque nœud
- Tableau décrivant les types de messages (trames) utilisés

Items évalués	GIF550-1
Rapport	
Description de l'architecture du système (branchement, adressage des composants, ...)	/10
Spécification de la topologie et des paramètres	/10
Spécification des messages et des communications entre les noeuds	/10
Gestion des erreurs	/10
Code	
Ensemble des fonctionnalités demandées	/15
Bonne pratique en développement C	/5
Respect des consignes (pointeurs, C, contenu du ZIP, ...)	/10
Qualité et pertinences des commentaires	/10
Total	/80

Tableau 2: Distribution des points

Le rapport sera corrigé une seule fois et une seule note sera attribuée aux aspects de l'ingénierie, en fonction du Tableau 2. Cependant, la qualité de la communication influence grandement la note attribuée aux aspects de l'ingénierie. Ainsi, des phrases mal construites ou pouvant porter à interprétation, des nombres sans unités ou des graphiques incomplets sont autant d'exemples qui nuisent à la compréhension technique et font perdre des points sur cet aspect. Les correcteurs n'ont pas à interpréter des phrases ou des paragraphes. Soyez donc précis et concis. La qualité de la communication sera évaluée selon le Tableau 3 (Évaluation de la communication). 15% pourront être retranchés à la note du rapport en fonction de la grille de correction. L'atteinte du niveau *cible* dans la grille assure de ne pas perdre de points, mais l'atteinte d'un niveau inférieur au niveau *cible* fait automatiquement perdre des points.

Critères	0	Non-initié (1)	Seuil (2)	Cible (3)	Excellent (4)
Organiser l'information	L'étudiant n'est pas en mesure d'organiser l'information pour en permettre la compréhension. Il n'y a aucune transition entre les sections.	L'étudiant organise minimalement l'information pour en permettre la compréhension. Les transitions entre les sections sont soit abruptes, soit inefficaces.	L'étudiant organise adéquatement l'information pour en permettre la compréhension chez son auditoire. La qualité des transitions entre les sections pourrait toutefois être améliorée.	L'étudiant organise efficacement l'information pour en faciliter la compréhension chez son auditoire. Les transitions entre les sections témoignent d'une compréhension approfondie du sujet.	
Présenter de l'information pertinente	L'étudiant éprouve de la difficulté à sélectionner l'information pertinente à présenter au regard des objectifs de la communication.	L'étudiant présente de l'information pertinente au regard des objectifs de la communication, mais des éléments importants sont soit mal présentés, soit absents.	L'étudiant présente de l'information pertinente et complète au regard des objectifs de la communication.	L'étudiant présente de l'information pertinente et complète au regard des objectifs de la communication. Le sujet est bien délimité et est présenté avec concision.	
Recourir à des références	L'étudiant recourt à trop peu de références et/ou abuse de citations et de paraphrases.	L'étudiant recourt à des références pertinentes, mais peu variées. Les citations et paraphrases présentées sont soit trop longues, soit déformées/s'inexactes.	L'étudiant recourt à des références pour appuyer ses propos. Les citations et paraphrases présentées sont généralement pertinentes.	L'étudiant recourt à des références pertinentes et variées pour appuyer ses propos. Les citations et paraphrases présentées sont toutes pertinentes.	
Présenter des communications graphiques de qualité	L'étudiant présente des communications graphiques (représentations 3D, schémas, courbes 2D, etc.), mais celles-ci n'appuient pas le texte.	L'étudiant présente des communications graphiques (représentations 3D, schémas, courbes 2D, etc.) qui appuient le texte, mais celles-ci ne sont pas toujours faites selon l'état de l'art.	L'étudiant présente des communications graphiques (représentations 3D, schémas, courbes 2D, etc.) qui appuient le texte et généralement faites selon l'état de l'art.	L'étudiant présente des communications graphiques (représentations 3D, schémas, courbes 2D, etc.) qui appuient le texte, sont faites selon l'état de l'art et sont accompagnées d'un titre évocateur. Elles ajoutent ainsi une plus-value au document produit.	
Rédiger dans une langue de qualité	L'étudiant éprouve de la difficulté à rédiger des phrases complètes et bien structurées, de même qu'à respecter les règles grammaticales élémentaires, ce qui nuit à la compréhension de son texte. Il n'utilise pas une terminologie et un vocabulaire appropriés.	L'étudiant fait encore des erreurs d'orthographe, de syntaxe et de grammaire, mais celles-ci ne nuisent pas de façon importante à la compréhension de son texte. Il utilise une terminologie et un vocabulaire minimalement appropriés.	L'étudiant rédige généralement des phrases complètes et bien structurées. Il respecte les règles grammaticales élémentaires, en plus d'utiliser une terminologie et un vocabulaire généralement appropriés.	L'étudiant rédige des phrases complètes et bien structurées. Il respecte les règles grammaticales élémentaires, en plus d'utiliser une terminologie et un vocabulaire parfaitement appropriés.	
Respecter les consignes données	L'étudiant remet une communication écrite respectant trop peu les consignes données, tant pour le fond que pour la forme.	L'étudiant remet une communication écrite respectant quelques-unes des consignes données, tant pour le fond que pour la forme.	L'étudiant remet une communication écrite respectant la majorité des consignes données, tant pour le fond que pour la forme.	L'étudiant remet une communication écrite respectant toutes les consignes données, tant pour le fond que pour la forme.	

Tableau 3: Évaluation de la communication

14. Évaluation sommative

L'évaluation sommative porte sur tous les objectifs d'apprentissage de l'unité.

15. Évaluation de l'unité

La note attribuée aux activités pédagogiques de l'unité est une note **individuelle**. L'évaluation portera sur les compétences figurant dans la description des activités pédagogiques. Ces compétences, ainsi que la pondération de chacune d'entre elles dans l'évaluation de cette unité, sont :

<i>Activité et élément de compétence</i>		<i>Rapport d'APP</i>	<i>Évaluation formative</i>	<i>Examen Sommatif</i>
GIF-550 Systèmes embarqués réseautés				
1	Concevoir, développer et mettre au point une application sur un système embarqué exploitant la connectivité à un réseau informatique	80	--	120
<i>Total</i>		80		120