



MATRIX | EBLOCKS 2

Bluetooth Communications



CP1795

MATRIX
www.matrixsl.com

Copyright © 2018 Matrix Technology Solutions Limited

CP1795

Communication Bluetooth

Notes de cours

Contenu

1 Introduction 6

1.1 Structure de ces notes 6

1.2 Résultats de l'apprentissage 6

1.3 Utilisation de ce manuel 7

1.3.1 Introduction aux sections relatives à la mise en œuvre pratique 7

2 Qu'entend-on par Bluetooth? 8

2.1 Le module Bluetooth RN46788

2.2 Mode de commande 8

2.3 Matériel et logiciels utilisés dans le cours 8

2.3.1 Blocs E2 parties 8

2.3.2 Code de flux 9 2.4

Autres dispositifs utiles 9

2.4.1 Dongle Bluetooth/USB embarqué 9

3 Garantie 10

3.1 Mise en place du

matériel 10 3.1.1 Mise en place

des E-blocks 210

3.2 Introduction à Flowcode 11

3.2.1 Début en Flowcode 11

3.3 Le composant Bluetooth 11

3.3.1 Propriétés et connexion des broches 12

3.3.2 Macros du composant Bluetooth 12

3.4 Tester le matériel 12 4 Théorie et

contexte de Bluetooth 13

4.1 Introduction à Bluetooth 13

4.1.1 Brève histoire 13

4.1.2 Concepts de Bluetooth 13

4.1.3 Avantages de Bluetooth 14

4.1.4 Inconvénients de

Bluetooth 14 4.2 Protocoles et

modèle OSI 14 4.2.1 Application 16

4.2.2 Couches Bluetooth 16

4.2.3 Détails du matériel 17

4.2.4 Profils 19

5 Découverte 20

5.1 Théorie : Trouver d'autres appareils Bluetooth 20

5.1.1 Les commandes d'enquête 20

5.1.2 Paramètres d'enquête supplémentaires 20

5.1.3 Exemple de découverte 21

5.2 Exercice 1 : découverte des appareils Bluetooth 22

5.2.1 Introduction 22

5.2.2 Objectifs 22

5.2.3 Pré-requis 22

5.2.4 Exigences en matière de matériel et de logiciel 22

5.2.5 Informations sur l'exercice 22

5.2.6 Résultat de l'apprentissage 23

5.2.7 Tâches supplémentaires 23

5.3 Mise en œuvre pratique : Découverte des appareils Bluetooth 23

5.3.1 À découvrir et découvrable 23

5.3.2 Planification du programme 24

5.3.3 Macros requises 24

5.3.4 Initialisation 25

5.3.5 Envoi d'une commande	25	6.3.1 Poursuite du développement	31
5.3.6 Vérification des réponses	26	6.3.2 Configuration du dispositif Bluetooth	31
5.3.7 Exécution du programme complet	27	6.3.3 Création du programme	32
6.1 Théorie : La découvrabilité	28	6.3.4 Exécution et démonstration du programme	33
6.1.1 Configuration pour le démarrage	28	7 Connexion des appareils Bluetooth	34
6.1.2 Utilisation des commandes Set	28	7.1 Théorie : Connexion - Adresses	34
6.1.3 Utilisation des commandes Get	28	7.2 Exercice 3 : connexion à un appareil	35
6.1.4 Utilisation des commandes d'action	28	7.2.1 Introduction	35
6.1.5 Découvrir un appareil	29	7.2.2 Objectifs	35
6.1.6 Entrer dans le mode commande	29	7.2.3 Pré-requis	35
6.1.7 Attendre l'exécution de la commande	29	7.2.4 Exigences en matière de matériel et de logiciel	35
6.1.8 Vitesse de transmission et réglages	29	7.2.5 Informations sur l'exercice	35
6.1.9 Mode d'authentification	29	7.2.6 Résultat de l'apprentissage	35
6.1.10 Mode esclave	29	7.2.7 Autres travaux	35
6.1.11 Activation de la configuration mise à jour	29	7.3 Mise en œuvre pratique : Connexion	36
6.1.12 Attendre le redémarrage et quitter le mode commande	29	7.3.1 Réinitialisation des systèmes	36
6.1.13 Résumé des commandes de configuration de base	29	8 Clés d'accès et connexion	37
6.2 Exercice 2 : découvrabilité	30	8.1 Théorie : Passkeys et Connecting	37
6.2.1 Introduction	30	8.1.1 Envoi de la commande Passkey	37
6.2.2 Objectifs	30	8.1.2 Initiation de l'appariement et de la connexion	37
6.2.3 Pré-requis	30	8.2 Exercice 4 : Passkeys et Connecting	38
6.2.4 Exigences en matière de matériel/logiciel	30	8.2.1 Introduction	38
6.2.5 Informations sur l'exercice	30	8.2.2 Objectifs	38
6.2.6 Résultats de l'apprentissage	30	8.2.3 Pré-requis	38
6.2.7 Tâches supplémentaires	30	8.2.4 Exigences en matière de matériel/logiciel	38
6.3 Mise en œuvre pratique : Découvrabilité	31	8.2.5 Informations sur l'exercice	38
		8.2.6 Résultat de l'apprentissage	38
		8.2.7 Autres travaux	38
		8.3 Mise en œuvre pratique : Passkeys et Connecting	39
		8.3.1 Le programme de base	39
		8.3.2 Fonctions avancées : Choisir l'appareil auquel se connecter	40
		8.3.3 Réinitialisation des systèmes	40
		9 Vérification des réponses	41

9.1 Théorie : Vérification des réponses	41	11.1.1 Sécurité en général	48
9.1.1 Réponses sollicitées et non sollicitées	41	11.1.2 Modes de sécurité	49
9.1.2 Macros de gestion des réponses	41	11.1.3 Authentification	49
9.2 Exercice 5 : Vérification des réponses	43	11.1.4 Confiance	49
9.2.1 Introduction	43	11.2 Exercice 7 : confiance et sécurité	50
9.2.2 Objectives	43	11.2.1 Introduction	50
9.2.3 Pre-requis	43	11.2.2 Objectifs	50
9.2.4 Exigences en matière de matériel/logiciel	43	11.2.3 Pré-requis	50
9.2.5 Informations sur l'exercice	43	11.2.4 Exigences en matière de matériel et de logiciel	50
9.2.6 Résultats de l'apprentissage	43	11.2.5 Informations sur l'exercice	50
9.3 Mise en œuvre pratique : Vérification des réponses	44	11.2.6 Résultat des apprentissages	50
9.3.1 Utilisation de la macro WaitForStringValue	44	11.3 Mise en œuvre pratique : Confiance et sécurité	51
9.3.2 Méthodologie de vérification des erreurs	44	11.3.1 Objectifs généraux	51
10 Introduction à la technologie Bluetooth Low Energy (BLE)	45	11.3.2 Caractéristiques supplémentaires	51
10.1 Modes de fonctionnement de Bluetooth Low Energy (BLE)	45	12 Principes de conception des projets	52
10.2 Exercice 6 : BLE GAP transparent UART serial data service	46	12.1 Projet - Télécommande simple	52
10.2.1 Introduction	46	12.1.1 Extension du projet	52
10.2.2 Objectifs	46	12.2 Projet - Enregistreur de données médicales	52
10.2.3 Pré-requis	46	12.2.1 Extension du projet	52
10.2.4 Exigences en matière de matériel/logiciel	46		
10.2.5 Informations sur l'exercice	46		
10.2.6 Résultats de l'apprentissage	46		
10.3 Mise en œuvre pratique : Service de données série BLE	47		
10.3.1 Mode BLE	47		
10.3.2 Secure Simple Pairing	47		
10.3.3 Activation des paramètres	47		
11 Confiance et sécurité	48		
11.1 Théorie :			
Confiance et sécurité	48		

Introduction

Structure de ces notes

Ces notes sont présentées comme suit :

Partie 1 : Introduction générale à Bluetooth et au kit de communication Bluetooth

- Mise en route - présentation du matériel et du logiciel

Historique de Bluetooth et aperçu général

Partie 2 : Le cours

Une série d'exercices progressifs permettant aux étudiants d'acquérir les concepts et la pratique nécessaires à l'établissement de communications en mode "dent bleue".

Les chapitres sont divisés en 3 sections pour chaque chapitre.

- Section théorique

- Description de

l'exercice Notes

pratiques

Partie 3 : Références et annexe

Des chapitres annexes fournissent des informations

de référence. Comprend :

- Section de référence

Une section de référence "Comment faire" pour la programmation

Résultats de l'apprentissage

Ces notes du professeur sont conçues pour présenter les concepts et les stratégies nécessaires à la pratique des communications Bluetooth. En réalisant les exercices de ce cours, les étudiants apprendront ce qui suit :

- Comment un dispositif Bluetooth découvre un autre dispositif Bluetooth et les options concernant la découvrabilité
- Comment les appareils Bluetooth s'appairent et établissent un canal de communication
- Comment les données de différents types sont transférées entre les appareils Bluetooth Comment la confiance et la sécurité sont gérées par Bluetooth.

Ces notes n'aborderont pas les caractéristiques des fréquences radio et les caractéristiques et protocoles de transmission de bas niveau de Bluetooth. Tous ces éléments sont gérés par un module Bluetooth disponible sur le marché, qui protège les utilisateurs de ces problèmes.

Ces notes sont structurées en un certain nombre de sections qui vous amènent d'abord à installer, configurer et tester le matériel et le logiciel dans le contexte de Bluetooth, puis à une série d'exercices et d'exemples qui amènent l'étudiant à comprendre le fonctionnement de Bluetooth.

Les exercices doivent être réalisés en utilisant Flowcode V10 ou une version ultérieure, un langage de programmation graphique. Le composant Flow- code Bluetooth est conçu pour permettre aux étudiants de se familiariser avec Bluetooth sans se heurter aux problèmes de la programmation en C ou dans un langage de niveau inférieur.

Utilisation de ce manuel

La partie principale de ce manuel est structurée autour d'une approche en trois parties :

- **Théorie.** La première partie d'un chapitre introduit le sujet en question et discute de la théorie qui le sous-tend, explique les commandes utilisées et les séquences et stratégies générales requises.
- **Exercice.** Le deuxième élément est un exercice. L'exercice est donné ici afin que les buts et les objectifs soient compris et gardés à l'esprit lors de la lecture de la section suivante.

Mise en œuvre pratique. La troisième section traite de la mise en œuvre pratique de l'exercice. Les éléments abordés ici comprennent les macros Flowcode utilisées, les stratégies Flowcode et tout autre élément d'information ou conseil nécessaire à la mise en œuvre des objectifs de l'exercice.

Il est prévu que la théorie soit d'abord expliquée aux étudiants sous la forme d'un cours magistral ou d'un polycopié. Les étudiants peuvent ensuite recevoir l'exercice. Les superviseurs ont le choix de distribuer ou non les notes de mise en œuvre pratique.

Nous suggérons que pour chaque exercice, les étudiants reçoivent la (les) feuille(s) d'exercice et les notes de mise en œuvre pratique. Les étudiants devraient créer les programmes en Flowcode. Les premières notes de mise en œuvre pratique sont assez détaillées et fournissent de nombreuses informations sur la manière dont le programme doit être construit. Les exercices ultérieurs ne sont pas accompagnés de notes de mise en œuvre pratique aussi détaillées : l'étudiant doit alors utiliser ses connaissances pour accomplir les tâches détaillées dans l'exercice.

La fiche technique du module Bluetooth est un élément clé de la solution Bluetooth : les étudiants devront rechercher la signification des commandes et des fonctions du module Bluetooth. En conséquence, les étudiants doivent recevoir une copie du guide de l'utilisateur du module Bluetooth qui est fourni dans le fichier zip des ressources CP1795.

Pour la plupart des exercices, un ensemble complet de programmes de solution est fourni. Ceux-ci se présentent sous la forme de deux programmes - un pour chaque nœud Bluetooth. Notez que ces exemples nécessiteront la modification de tous les identifiants de dispositifs utilisés dans le programme.

Une introduction aux sections de mise en œuvre pratique

Cette première section pratique se présente sous la forme d'un exemple travaillé avec toutes les informations pratiques fournies et un programme exemplaire construit. Les sections pratiques ultérieures contiendront les informations supplémentaires nécessaires pour atteindre les objectifs de l'exercice, et contiendront des extraits de code pour les nouvelles procédures et macros, mais n'incluront pas nécessairement des programmes d'exemple complets. A ce stade, les étudiants sont supposés être suffisamment compétents pour pouvoir créer les programmes à partir des informations fournies dans la section pratique.

Il est prévu que, dans la mesure du possible, les programmes créés par l'étudiant pour l'exercice précédent soient réutilisés pour l'exercice en cours. Cela permet d'illustrer à la fois l'évolution du programme au fur et à mesure que de nouvelles étapes sont ajoutées, et de montrer la complexité croissante et le déroulement du programme d'un système complet. Cette approche permet également aux étudiants de disposer d'un code de base qui leur est déjà familier. Cependant, les programmes des étudiants peuvent avoir besoin d'être évalués à différents moments pour s'assurer que le code est adéquat pour le projet en cours, et des modifications ou des commentaires peuvent être apportés si nécessaire pour diriger les programmes des étudiants à travers le cours dans son ensemble. Ces modifications peuvent prendre la forme d'une adaptation de certaines parties des programmes en macros afin d'alléger certaines parties du programme, ou de réécritures de code contrôlées et évaluées afin d'implémenter des éléments d'une manière plus organisée et plus efficace.

Qu'entend-on par Bluetooth ?

Bluetooth est accessible à plusieurs niveaux. Bluetooth est à la fois une technologie de communication et une stratégie de communication. En tant que technologie de communication, Bluetooth transforme les données d'une application en signaux radio et vice-versa. En tant que stratégie de communication, Bluetooth permet de découvrir et de se relier à d'autres appareils Bluetooth et d'accéder aux fonctions de ces appareils.

Ce cours se concentre sur les stratégies de communication de Bluetooth. C'est à ce moment que l'utilisateur final fera l'expérience des technologies Bluetooth, telles que l'envoi de données entre appareils.

La technologie de communication Bluetooth est généralement gérée automatiquement par les dispositifs Bluetooth. À moins de concevoir activement des puces Bluetooth, les couches de communication inférieures resteront cachées à la plupart des utilisateurs finaux. La théorie et le contexte ont été fournis pour montrer comment Bluetooth gère les communications.

Le module Bluetooth RN4678

Le dispositif utilisé sur la carte Matrix Bluetooth BL0170 est un dispositif Microchip RN4678. Il s'agit d'un module Bluetooth autonome avec lequel il est possible de communiquer à l'aide d'un ensemble de commandes via un UART série. Ce dispositif permet aux utilisateurs de se connecter à d'autres dispositifs Bluetooth et de les utiliser au niveau de l'application. L'utilisation de ce dispositif permet au cours de se concentrer sur les stratégies et les séquences de communication entre les dispositifs Bluetooth sans qu'il soit nécessaire d'approfondir les niveaux technologiques inférieurs.

Mode commande

Le RN4678 fonctionne selon deux modes : Le mode Données (par défaut) et le mode Commande. Lorsque le RN4678 est connecté à un autre dispositif et qu'il est en mode Données, le RN4678 agit comme un tuyau de données : tout ce qui est reçu de l'UART est transmis au dispositif pair connecté via SPP s'il est connecté à un dispositif Bluetooth Classic, ou via un service GATT privé s'il est connecté à un dispositif BLE. Lorsque des données sont reçues de l'appareil homologue par SPP pour Bluetooth Classic ou par UART Transparent pour BLE, ces données sortent directement sur UART.

Le RN4678 est configuré ou contrôlé, ou les deux, en le plaçant en mode commande et en exécutant des commandes ASCII sur UART.

..

[Matériel et logiciels utilisés dans le cours](#)

Le cours sur le Bluetooth utilise le matériel et les logiciels suivants :

E-blocks2 pièces

Le kit de formation E-Blocks2 contiendra deux kits complets comprenant les éléments suivants :

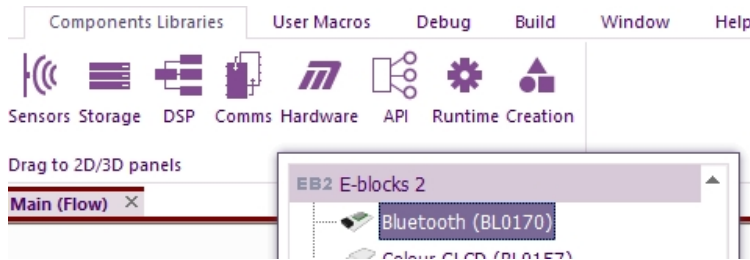
- BL0011 Programmeur PIC ou BL0055 Programmeur Arduino
- BL0170 Carte Bluetooth
- BL0169 Écran LCD

Et un de chaque

- BL0145 Carte de commutation
- BL0167 Carte LED

Les programmes d'exemple utilisés dans ce cours nécessitent Flowcode V10 ou une version ultérieure.

Pour Flowcode V10, le composant **Bluetooth BL0170** se trouve sous **E-blocks2** dans la section **Hardware** de la barre d'outils **Component Libraries** :



Tous les fichiers d'exemples et de programmes d'exercices fournis dans le fichier zip des ressources CP1795 ont été créés pour Flowcode V10 ou une version ultérieure.

Ce cours suppose une certaine familiarité avec Flowcode. Si nécessaire, l'étudiant doit passer du temps à utiliser les tutoriels et le cours Flowcode pour se familiariser avec l'utilisation de Flowcode.

Autres dispositifs utiles

Les dispositifs suivants ne sont pas fournis avec la solution Bluetooth, mais s'ils sont disponibles, ils peuvent être utilisés avec la solution Bluetooth

Clé Bluetooth/USB embarquée

Pour les communications vers et depuis le PC, un module Bluetooth est nécessaire. Certains ordinateurs portables et PC récents sont déjà câblés pour Bluetooth. La plupart des PC et des ordinateurs portables actuels nécessitent cependant un module Bluetooth qui se branche sur un port USB libre et permet à l'ordinateur de fonctionner comme un appareil Bluetooth.

Notez qu'il n'est pas nécessaire d'avoir un module Bluetooth sur votre PC pour réaliser tous les exercices.

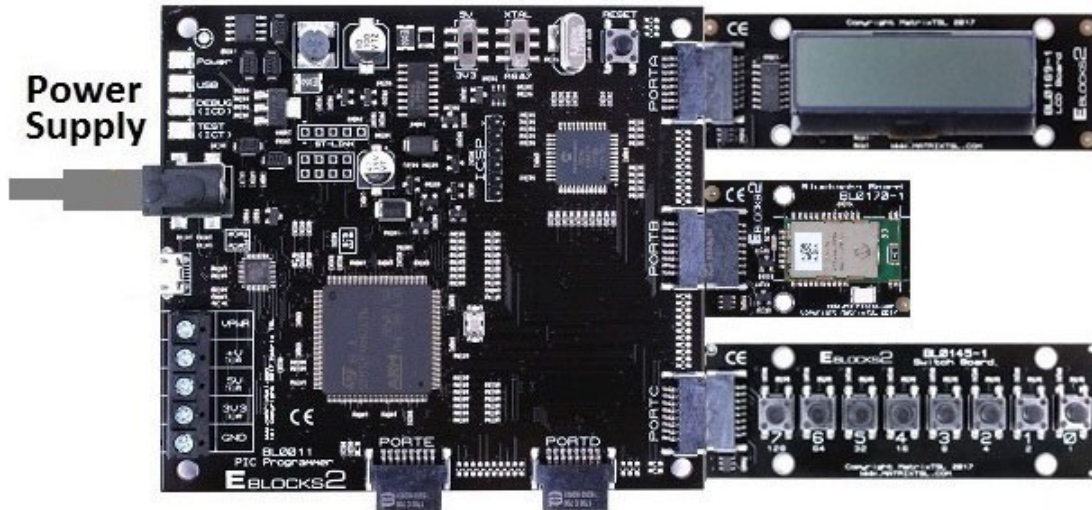
Pour commencer

Mise en place du matériel

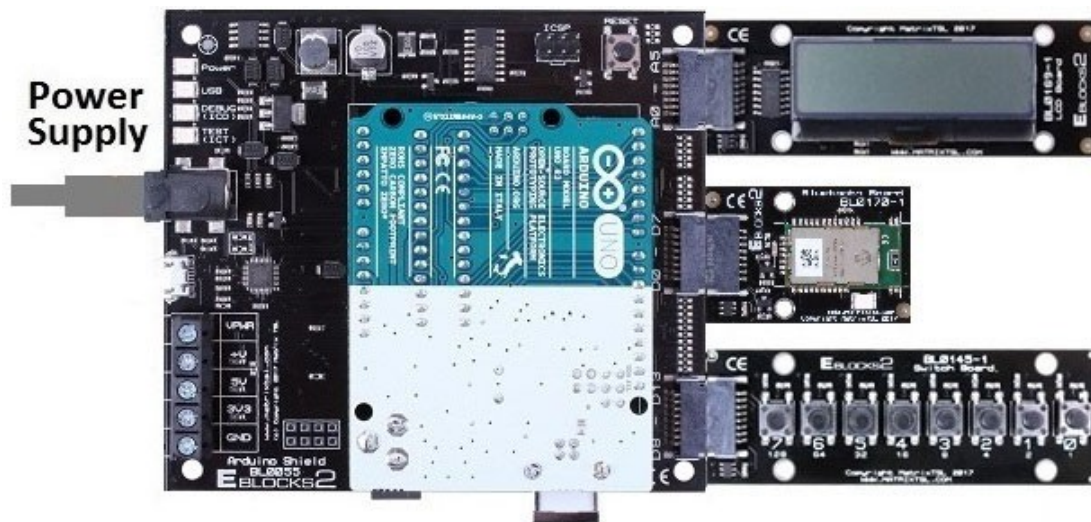
E-blocks2 setup

Les deux jeux de cartes E-blocks2 doivent être connectés comme indiqué ci-dessous.

Certains exercices utilisent un tableau de distribution sur le nœud B uniquement, le nœud actif, comme indiqué.



PORT	BL0011 Programmeur PIC
PORT A	BL0169 Carte LCD
PORT B	BL0170 Carte Bluetooth
PORT C	Certains projets utilisent la carte de commutation BL0145 ou la carte LED BL0167.



PORT	BL0055 Programmeur Arduino
PORT A0-5	BL0169 Carte LCD
PORT D0-7	BL0170 Carte Bluetooth
PORT D8-13	Certains projets utilisent la carte de commutation BL0145 ou la carte LED BL0167.

Flowcode est un système d'organigramme pour les microcontrôleurs. En général, les microcontrôleurs sont programmés en C (un langage difficile à apprendre) ou en langage assembleur (un langage encore plus difficile à programmer). L'enseignement des microcontrôleurs a généralement nécessité un investissement important en temps afin de permettre aux étudiants d'acquérir les bases des langages requis avant qu'ils ne puissent s'attaquer à des systèmes tels que Bluetooth.

Flowcode est une méthode de création de programmes beaucoup plus simple et intuitive. Basé sur des symboles d'organigramme standard et utilisant des icônes par glisser-déposer, un programme fonctionnel peut être construit en quelques minutes. Les icônes peuvent être configurées à l'aide de boîtes de dialogue qui éliminent le risque d'erreurs de syntaxe ou de sélection d'options non valides. Les organigrammes peuvent être suivis visuellement, ce qui permet aux utilisateurs d'avoir une vue d'ensemble et de ne pas se limiter à un seul élément. La plupart des cours d'introduction à la programmation utilisent, ou recommandent de créer, des organigrammes comme précurseurs de l'écriture du code final, en raison de leur capacité à décomposer le flux du programme d'une manière claire et compréhensible. Avec Flowcode, c'est l'écriture de votre programme.

Une gamme de composants peut être utilisée avec des programmes qui vont des LED et interrupteurs de base jusqu'aux systèmes de communication complets tels que CAN, TCP/IP et, bien sûr, Bluetooth. Une fois de plus, les dialogues sont utilisés pour éliminer les possibilités d'erreurs de syntaxe ou d'appels de fonctions mal assemblés. Les icônes de macro permettent à l'utilisateur d'accéder à des fonctions complètes, ce qui lui permet d'effectuer des tâches aussi variées que l'allumage d'une simple LED, l'envoi de texte à un écran LCD ou la vérification des messages entrants sur un système Bluetooth.

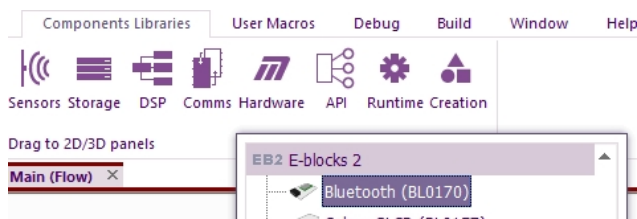
Le système est conçu pour être facile à comprendre et à utiliser, tout en offrant la profondeur et la flexibilité requises par les marchés techniques, éducatifs et industriels d'aujourd'hui.

Débuter avec Flowcode

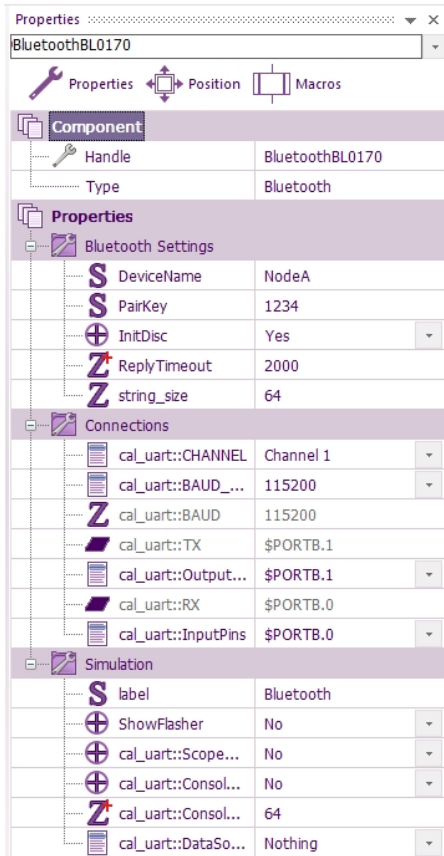
Ce cours suppose un certain degré de familiarité avec Flowcode. Les nouveaux utilisateurs de Flowcode devraient consulter les didacticiels et le cours d'introduction de Flowcode avant d'utiliser Flowcode dans le cadre de ce cours. Ces didacticiels et ce cours sont disponibles sur le site Web www.flowcode.co.uk. Consultez le site web de Matrix TSL www.matrixtsl.com et le Matrix Learning Centre pour obtenir des détails sur les cours et des liens.

La composante Bluetooth

Pour Flowcode V10, le composant **Bluetooth BL0170** se trouve sous **E-blocks2** dans la section **Hardware** de la barre d'outils **Component Libraries** :



Ajouter ce composant au panneau système



Le composant Bluetooth BL0170 est basé sur le dispositif RN4678 et possède un certain nombre de propriétés et de connexions qui peuvent être définies par l'utilisateur.

Il s'agit notamment du nom de l'appareil, de la clé d'appariement et de la possibilité ou non de découvrir l'appareil. Ces paramètres sont appliqués lorsque l'appareil est initialisé.

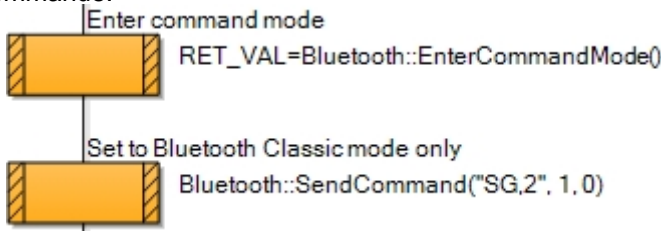
Les détails de la connexion UART série à l'appareil sont également configurés ici.

Macros de composants Bluetooth

Le composant Bluetooth utilise un certain nombre de macros (ou fonctions) pour aider à programmer le périphérique Bluetooth. Les détails complets des macros, de leurs paramètres et de leurs utilisations sont disponibles dans le fichier d'aide du composant Bluetooth.

La section Exercices d'enseignement du cours présentera et expliquera l'utilisation des macros au fur et à mesure qu'elles sont nécessaires pendant les exercices.

L'exemple ci-dessous, par exemple, force l'appareil Bluetooth à passer en mode commande, puis émet une commande.



Test du matériel

Les kits de formation Bluetooth peuvent être testés et leurs adresses de périphériques Bluetooth déterminées à l'aide du programme de test Flowcode inclus, nommé BT_TEST,fcfx.

Elle se trouve dans le fichier zip des ressources CP1795.

Connectez les cartes comme indiqué à la section 3.1 et mettez-les sous tension. Connectez chaque kit à tour de rôle au PC hôte exécutant Flowcode via le câble USB et téléchargez le programme.

Lorsque le test est réussi, l'écran LCD affiche l'adresse de l'appareil du module Bluetooth connecté. Cette adresse peut être notée et utilisée comme adresse de connexion dans les exercices suivants.

Introduction à Bluetooth

Bref historique

"Le roi Harald a fait construire ce monument pour Gorm, son père, et Thyri, sa mère. C'est ce même Harald qui a conquis tout le Danemark et la Norvège, et qui a fait des Danois des chrétiens". C'est ce que dit la pierre runique de Jelling. Ce Harald n'est autre que le légendaire roi viking Harald Bluetooth, qui a donné son nom à la technologie Bluetooth. Au départ, Bluetooth n'était rien d'autre qu'un nom de projet, un nom pour désigner la technologie en cours de développement. Mais le nom est resté.

Et c'est un nom tout à fait approprié. Tout comme le Bluetooth de l'époque des Vikings a rassemblé des nations disparates, leur a donné un nouveau credo et les a conduites à la grandeur, le Bluetooth des temps modernes rassemble des dispositifs matériels disparates, leur donne un nouveau protocole et étend leurs capacités.

Aujourd'hui, le monument de Bluetooth ne serait pas des runes gravées sur la pierre, mais le flot de signaux électroniques que le monde communique via Bluetooth.

N.B. Le logo Bluetooth contient les runes H et B pour Harald Bluetooth :



Bluetooth a été introduit en 1998 par le Bluetooth Special Interest Group, une fédération d'entreprises impliquées dans les communications, l'industrie et les technologies commerciales. La diversité et le nombre d'entreprises impliquées ont contribué au succès de Bluetooth et à son adoption par un grand nombre de systèmes différents.

La version 1.1 de la spécification Bluetooth (les versions 1.0 et 1.0B étant trop limitées d'un point de vue technologique pour permettre un développement majeur) a connu un succès immédiat. La spécification a été mise à jour à la version 1.2 en 2003 avec des caractéristiques supplémentaires telles que des vitesses plus élevées et des sauts de fréquence adaptatifs. En 2004, la version 2.0 + la spécification EDR (Enhanced Data rate) a été publiée. La version 2.0+EDR est plus rapide, consomme moins d'énergie et permet d'améliorer les communications entre des dispositifs multiples. Bluetooth continue de se développer et de s'adapter à l'évolution rapide des besoins en télécommunications du marché moderne.

Concepts Bluetooth

- Bluetooth est conçu pour permettre un transfert de données sans fil sécurisé entre des appareils matériels.
- Grâce à l'utilisation d'un protocole, les données peuvent être transmises entre les dispositifs de manière standard. Cela simplifie le codage et les tests car il n'est pas nécessaire de créer un code propriétaire.
- Bluetooth est entièrement spécifié, ce qui permet aux fabricants de créer du matériel compatible Bluetooth sans avoir à concevoir et à mettre en œuvre des protocoles propriétaires de haut niveau. Cela permet également de simplifier les réseaux, car il n'y a pas de problèmes avec des protocoles ou des spécifications propriétaires concurrents.
- La communication est automatique dès que deux appareils Bluetooth sont à portée. Cela signifie que les utilisateurs n'ont pas besoin de suivre des procédures de configuration pour commencer à communiquer.

Les appareils Bluetooth peuvent être sécurisés et ne pas être découverts, ce qui permet d'éliminer les menaces de sécurité liées à la communication automatique. C'est à vous de décider si vous voulez faire partie du réseau ou non.

Bluetooth est essentiellement un réseau de communication sans fil capable de découvrir et de communiquer automatiquement avec les nouveaux appareils qui se trouvent à sa portée. L'utilisateur bénéficie ainsi de deux avantages essentiels.

Avantages du Bluetooth

Les câbles ne sont pas nécessaires.

Pas de câblage spaghetti qui court partout. Pas de désordre inesthétique. Pas de risque de trébucher ou d'endommager. Et pour des utilisations telles que la technologie automobile, pas de faisceaux de câbles à installer et à entretenir/réparer/remplacer.

Pas de contraintes physiques dues à l'emplacement des câbles ou des points de réseau, d'où une plus grande liberté de mouvement pour les appareils.

La communication étant basée sur la distance, les appareils peuvent être mobiles.

Les appareils peuvent être déplacés d'un réseau à l'autre sans nécessiter d'infrastructure.

Détection automatique des autres appareils dans le rayon d'action.

Il n'est pas nécessaire de procéder à des changements physiques, tels que de nouveaux points de réseau, pour accueillir de nouveaux appareils.

Par conséquent, la facilité d'utilisation et la facilité de mise à niveau.

Les fonctions de sécurité Bluetooth vous permettent de configurer l'appareil pour qu'il réponde à d'autres appareils ou qu'il les ignore, ce qui vous permet d'être responsable du réseau.

De nouveaux appareils peuvent être ajoutés simplement en entrant dans la zone de couverture. Il n'est donc pas nécessaire d'éditer ou de modifier les paramètres ou quoi que ce soit pour ajouter de nouveaux appareils.

Inconvénients du Bluetooth

Complexité et fiabilité

Les câbles ne sont peut-être pas portables, mais ils sont simples et fiables. La complexité, comme dans les systèmes Bluetooth, augmente le risque de défaillance.

Plus un système est complexe, comme le Bluetooth, plus il peut être difficile de trouver et de réparer les défauts.

Durée de vie de la batterie

Les appareils mobiles sont limités par la durée de vie de la batterie et les possibilités de recharge. Des coupures de courant peuvent survenir pendant les communications.

Les câbles peuvent être normalisés, mais les prises de courant et les niveaux de tension ne le sont pas. En voyage, un adaptateur de prise de courant peut s'avérer nécessaire.

Coûts

Les dispositifs Bluetooth sont souvent plus coûteux que les solutions fixes conventionnelles telles qu'un câble. Un investissement initial peut être nécessaire pour passer d'un équipement ancien à des solutions Bluetooth.

Coûts de réparation plus élevés.

Vitesse

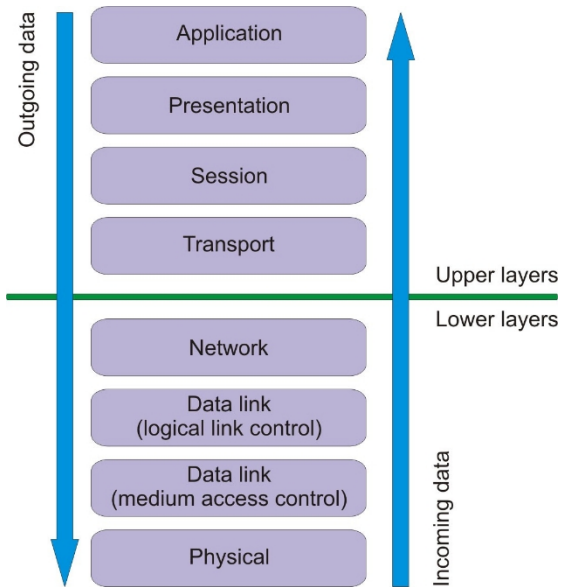
Bluetooth est plus lent que les solutions fixes équivalentes telles qu'un câble direct.

La vitesse des communications Bluetooth peut être limitée par des considérations de débit en bauds ou par les limites des vitesses de transfert de données du protocole Bluetooth.

Protocoles et modèle OSI

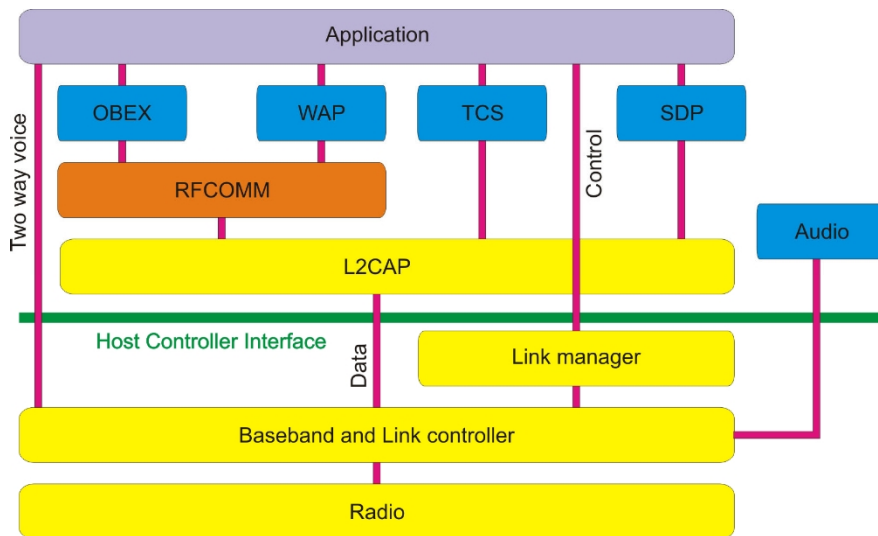
Bluetooth fonctionne par couches basées sur le modèle OSI. Le modèle OSI est une séquence de couches utilisées pour définir comment les données passent de l'application à l'envoi physique de signaux. Ce processus est répété en sens inverse lors de la réception de données pour passer des signaux aux données de l'application. Chaque couche aborde des aspects fondamentaux des processus de communication. En abstrayant les différentes parties du processus de communication en couches basées sur le modèle OSI, les développeurs peuvent mettre en œuvre leurs systèmes comme une série de tâches couvrant chacune une ou plusieurs couches OSI. Des standards peuvent être mis en œuvre pour des tâches ou des couches spécifiques, créant ainsi une base sur laquelle les développeurs peuvent travailler.

Le schéma de base du modèle OSI est présenté ci-dessous :



Le modèle OSI est utilisé pour aider à séparer les différents

éléments. Les couches Bluetooth sont présentées ci-dessous :



i

Le processus de base est simple : les données brutes au sommet doivent être converties en bits et en octets qui peuvent être transmis par signal radio et reconstitués en données brutes avec lesquelles l'application à l'autre extrémité peut travailler.

La transmission consiste à décomposer les données en morceaux qui peuvent être envoyés et à envelopper les morceaux de données avec des éléments d'information supplémentaires qui peuvent être utilisés par cette couche particulière pour vérifier des éléments tels que des erreurs ou des données manquantes. Un seul morceau de données provenant d'une application peut être décomposé en un certain nombre de morceaux différents, chacun d'eux étant entouré de plusieurs enveloppes différentes.

La réception implique un processus de passage du paquet, chaque couche déballant la couverture supérieure et transmettant les paquets à la couche spécifiée dans cette enveloppe. À chaque étape, d'autres fonctions peuvent intervenir, telles que le contrôle des erreurs et l'assemblage de groupes de paquets en un paquet plus important qui sera transmis à la couche supérieure une fois qu'il sera complet. Enfin, le paquet atteint l'application et les données peuvent être affichées à l'utilisateur (ou diffusées par le haut-parleur s'il s'agit de données audio, etc.)

Il n'est pas nécessaire d'utiliser toutes les couches. Certains paquets peuvent sauter plusieurs couches. Le processus dépend du type de données envoyées.

Application

L'application est l'élément qui crée ou utilise les données, qu'il s'agisse de données brutes, de signaux audio ou d'informations textuelles. C'est à ce niveau que les utilisateurs communiquent avec le système. Si nous utilisons le bureau de poste comme analogie pour Bluetooth, l'application serait les mots que nous écrivons ou lisons.

Les données que nous saisissons sous quelque forme que ce soit, orale ou dactylographiée, etc. sont converties en une forme adaptée à la transmission, qu'il s'agisse de données ASCII ou de données au format audio. Les données peuvent ensuite être envoyées dans le pipeline pour être transmises.

Lorsque les données sont reçues, elles sont dans un format que l'application peut comprendre. Tout le travail de regroupement des paquets de données et de réassemblage a été effectué. Il s'agit de données brutes prêtes à être utilisées.

Couches Bluetooth

OBEX/WAP/TCS/SDP/AUDIO

Il s'agit de protocoles pour les données. Plutôt que d'utiliser des modèles de données personnalisés, il est préférable d'utiliser les protocoles et les normes existants. Cela vous permet de communiquer avec d'autres applications via les mêmes protocoles standard. Les données sont converties au format du protocole, prêtes à être transmises, ou sont converties du protocole aux données brutes utilisées par l'application. L'utilisation de ces protocoles communs simplifie les tâches de programmation et facilite le transfert du code vers d'autres appareils.

Certains protocoles, tels que SDP et TCS, peuvent être utilisés pour aider à configurer les opérations Bluetooth. Le protocole SDP, par exemple, permet aux applications d'interroger le dispositif Bluetooth sur les services qu'il peut fournir.

Les protocoles couramment utilisés sont les suivants

OBEX - Échange d'objets

WAP - Wireless Access Protocol (Protocole d'accès sans fil)

TCS - Spécification de contrôle

téléphonique SDP - Protocole de

découverte de service

AUDIO - signal audio direct tel qu'il est utilisé pour les écouteurs

RFCOMM

RFCOMM, Radio Frequency **COMM**unication, est un protocole spécial. RFCOMM permet aux utilisateurs d'utiliser une connexion de port COM série standard plutôt qu'une connexion radio Bluetooth. Il s'agit de ports COM virtuels et non de ports COM physiques. RFCOMM fait croire à l'ordinateur que ses ports COM virtuels existent de la même manière que les ports physiques réels. Mais lorsqu'il reçoit des données, il les envoie discrètement à la liaison radio Bluetooth. L'application finale n'en sait rien ; pour elle, il s'agit simplement d'un port série vers lequel elle a été envoyée. Lorsqu'elle reçoit des données, il s'agit simplement de données provenant d'un port COM série, l'aspect Bluetooth lui étant totalement caché.

Le grand avantage est que la myriade de technologies de port série COM existantes peut être utilisée directement sans nécessiter d'adaptateurs spéciaux ou de nouveaux logiciels. Cela peut avoir un effet considérable sur le coût et l'effort de mise à niveau ou de développement des produits qui utilisent actuellement la technologie du port série COM. La meilleure analogie entre la poste et RFCOMM est le courrier électronique. Il s'agit d'une communication de type lettre, mais ce n'est pas une lettre. Les courriels ont le même format qu'une lettre, nous les écrivons et les recevons, mais le processus d'envoi n'est pas géré par la Poste. De même, le processus d'envoi de données dans les communications série RFCOMM n'est pas vraiment une communication série, même s'il semble l'être vers le contrôleur hôte ou le PC.

RFCOMM et le module RN4678

Le module RN4678 se situe entre la couche RFCOMM et la couche application. Le module gère l'aspect RFCOMM proprement dit, mais exige que les commandes décrites dans le jeu de commandes lui soient envoyées, de sorte qu'il ne se situe pas tout à fait au niveau de l'application. Cela permet aux programmeurs

pour se concentrer sur la stratégie de communication et laisser les spécificités RFCOMM au module Bluetooth RN4678.

L2CAP

Le contrôleur de liaison logique et le protocole d'adaptation contrôlent et coordonnent le flux de données en créant des canaux virtuels, des sessions et des transferts de fichiers. Le L2CAP est également la principale zone d'assemblage des paquets, qui décompose les données en paquets individuels prêts à être envoyés à la couche de bande de base pour la préparation de la transmission. Au retour, le L2CAP est la principale zone de réassemblage qui rassemble tous les paquets individuels et réassemble les données qu'ils contiennent. Dans notre bureau de poste à dent bleue, cette couche correspondrait aux pages de la lettre et à l'enveloppe dans laquelle elle est envoyée.

Interface du contrôleur hôte

L'interface contrôleur hôte relie le PC ou le système au matériel Bluetooth. Les éléments de l'interface contrôleur hôte comprennent les pilotes de périphériques nécessaires pour faire fonctionner le matériel Bluetooth et son interface. L'interface contrôleur hôte se situe au point de rencontre entre le matériel Bluetooth et le matériel de traitement des données.

Gestionnaire de liens

Le gestionnaire de liens est un langage de bas niveau permettant de configurer et de contrôler les liens.

Étant donné la sophistication des transmissions Bluetooth avec le saut de fréquence adaptatif et la large gamme de fréquences disponibles, le gestionnaire de liens est un outil très utile. Le gestionnaire de liens est utile au niveau du réseau et de la liaison de données. Le gestionnaire de liens est comparable au code postal que la Poste utilise pour automatiser une grande partie de son tri.

Contrôleur de bande de base et de liaison

Le contrôleur de bande de base et de liaison, qui correspond au niveau de liaison de données du modèle OSI, est un contrôleur de paquets. Il assemble les données qui lui sont envoyées en paquets de données prêts à être transmis par la couche radio. Le contenu des données et la personne qui les a envoyées n'ont pas d'importance. Tout ce qui compte, c'est que la couche bande de base puisse les assembler en paquets de données pour la transmission.

Les données qui lui sont transmises par la couche radio voient leurs étiquettes de paquet examinées et sont ensuite acheminées vers la couche appropriée. La couche bande de base ne se préoccupe pas du contenu des données, mais seulement de leur destinataire. Dans notre analogie avec la poste, cette couche correspond au centre de tri primaire et à l'adresse figurant sur la lettre.

Radio

La couche radio est la couche physique, le niveau le plus bas de Bluetooth. À ce niveau, la seule préoccupation réelle concerne le signal radio proprement dit. Une connexion radio est établie et les données sont envoyées. Les paquets de données assemblés par la bande de base sont envoyés et les paquets de données entrants sont reçus et transmis à la couche bande de base.

Détails du matériel

Le matériel Bluetooth utilise un certain nombre de caractéristiques matérielles et logicielles communes dans ses communications, qu'il est utile de connaître.

Piconets

Un appareil Bluetooth peut faire partie d'un réseau comprenant jusqu'à 8 appareils Bluetooth. Le réseau est constitué d'un appareil appelé appareil maître et d'un maximum de sept autres appareils esclaves. Ce mini-réseau est appelé Piconet. Les esclaves ne peuvent envoyer des messages qu'à l'appareil maître et non aux autres esclaves. Le maître communique avec les esclaves dans un système "Round robin" en s'adressant à chaque appareil esclave à tour de rôle. Les signaux peuvent donc être transmis à un autre appareil esclave par l'intermédiaire de l'appareil maître, mais pas directement.

Les prochains appareils Bluetooth auront la possibilité d'étendre ce système en communiquant entre les Piconets. Un appareil sera le maître d'un anneau et l'esclave d'un autre, ce qui permettra la communication entre les deux anneaux.

Radiomessagerie et interrogation

Tous les appareils Bluetooth ont la possibilité de devenir des appareils maîtres ou esclaves. L'appareil maître d'un Piconet est celui qui a initié le contact, les esclaves ceux qui ont répondu. Le contact est

Les appareils sont initiés par l'envoi de signaux d'interrogation afin de déterminer quels sont les appareils à proximité. En même temps, les appareils sont à l'écoute du contact d'un autre appareil. Si un appareil est trouvé, il peut être appelé pour établir le contact et un Piconet peut être établi ou rejoint s'il en existe déjà un. Si un maître existe, comme c'est le cas lorsqu'on rejoint un Piconet, le nouvel appareil sera un esclave. Si aucun Piconet n'existe déjà, l'appareil qui a initié le contact devient l'appareil maître. Lequel dépend de celui qui a reçu le signal d'interrogation. Comme les appareils peuvent rejoindre ou quitter le réseau à volonté, il existe des mécanismes pour remplacer un appareil maître qui quitte le Piconet.

Spécifications radio

Bluetooth fonctionne dans la bande radio ISM de 2,45 GHz. Cette bande est divisée en 79 canaux différents entre lesquels Bluetooth passe 1600 fois par seconde (spécifications 1.1 et 1.2). Les dispositifs Bluetooth plus récents de la spécification 2 peuvent être jusqu'à trois fois plus rapides. Bluetooth saute des fréquences pour éviter les interférences avec d'autres systèmes sur la même fréquence (par exemple, un moniteur pour bébé) et pour empêcher les appareils Bluetooth d'accaparer un canal particulier et de le bloquer pour d'autres appareils. Il est évident que le système doit savoir sur quel canal il doit passer. Cela se fait au niveau radio, l'émetteur et le récepteur suivant automatiquement les sauts de fréquence des autres appareils du réseau. Les signaux envoyés au niveau radio contiennent des données intégrées sur le modèle de saut de fréquence utilisé par l'appareil, que les autres appareils peuvent utiliser pour déterminer le prochain canal. Heureusement, tout cela se fait automatiquement, loin dans les couches, et il n'est donc pas nécessaire de configurer ou de programmer la carte Bluetooth E-blocks2.

En raison de la faible puissance du signal radio (environ 1 milliwatt), la portée est limitée à environ 30 pieds, mais c'est plus que suffisant pour la plupart des appareils Bluetooth.

Transfert audio

En raison de ses origines dans le domaine des télécommunications, les signaux audio constituent un élément majeur de nombreux dispositifs Bluetooth.

Afin de simplifier le transfert audio, Bluetooth peut accélérer le passage des signaux audio numérisés dans le système jusqu'à la sortie audio de l'appareil récepteur. Les données audio peuvent être envoyées par les méthodes normales, et les applications telles qu'un lecteur MP3 Bluetooth utiliseront généralement la voie normale d'application à application, car elle offre une meilleure intégrité des données, et la mise en mémoire tampon des données à l'extrémité réceptrice résoudra la plupart des problèmes de latence. Pour les communications directes telles que le téléphone, la voie directe est préférable car la qualité des données - c'est-à-dire le bruit - est moins problématique que la latence - c'est-à-dire l'attente pour entendre ce qui a été dit, car les données mettent du temps à passer dans le système. Les signaux sont transmis directement aux couches inférieures, traités et compressés au niveau de la bande de base, puis envoyés au récepteur qui transmet les données directement de son niveau de bande de base à l'appareil de sortie audio.

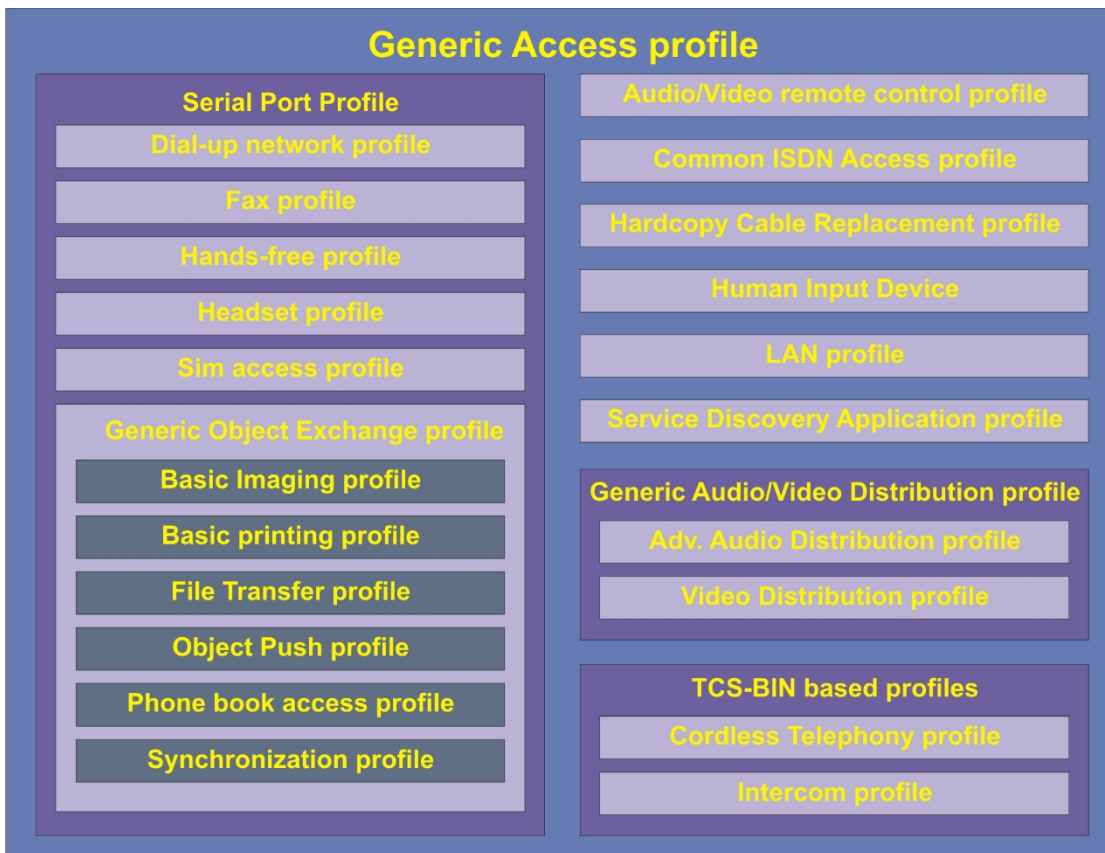
Authentification et cryptage

Bluetooth est un dispositif radio. Cela le rend intrinsèquement peu sûr, car tout appareil à portée de transmission peut écouter le signal. C'est pourquoi la sécurité est une préoccupation majeure pour Bluetooth.

Le premier niveau de sécurité est le couplage. Les appareils doivent être appariés, ce qui nécessite l'adresse et la clé d'accès de l'appareil avec lequel ils sont appariés. L'accès à l'adresse peut être bloqué par les appareils, ce qui rend l'appairage impossible, et la clé d'accès n'est pas accessible par les communications et doit donc être connue pour que l'appareil puisse être apparié. Dans la pratique, il s'agit d'un code PIN fourni avec le produit. Ainsi, pour coupler un téléphone et une oreillette, vous devez saisir le code PIN de l'oreillette dans le téléphone lorsque vous y êtes invité. Pas de code PIN - pas de communication.

Une fois l'appairage effectué, les communications peuvent commencer. Les appareils Bluetooth peuvent être configurés pour crypter et authentifier les communications, ce qui permet de s'assurer que les données proviennent d'un appareil approuvé et qu'elles sont codées, ce qui rend l'écoute infructueuse. N'importe quel code peut être craqué, mais le cryptage Bluetooth et la faible distance nécessaire à l'écoute en font un système très sûr.

Profils



Les profils sont des implémentations basées sur des codes à utiliser avec des structures matérielles standard telles que des casques ou des téléphones mobiles. La conformité à un profil permet au matériel d'utiliser les fonctions et les comportements appropriés à ce type d'appareil. Par exemple, l'utilisation du profil Headset permet à d'autres unités de communiquer avec le dispositif Bluetooth en utilisant des appels standard qui font partie de l'implémentation Headset. Le profil casque contiendra un certain nombre de fonctions qui se trouvent également dans l'implémentation d'un téléphone, mais de nombreuses fonctions téléphoniques ne se trouveront pas dans le profil casque. En effet, le profil casque est un sous-ensemble du profil téléphone.

La superposition des profils permet aux profils de niveau supérieur d'utiliser les mêmes fonctions que les profils de niveau inférieur. Le profil Fax, par exemple, peut utiliser les fonctions de transfert de fichiers qui font partie du profil Transfert de fichiers. Il peut également utiliser les caractéristiques du profil Synchronisation, qui est également un sous-ensemble du profil Échange d'objets génériques, qui est un sous-ensemble du profil Port série dans lequel réside le profil FAX.

Partie 2 : Exercices pédagogiques

Découverte

Théorie : Recherche d'autres appareils Bluetooth

Pour qu'un appareil Bluetooth puisse communiquer, il faut qu'il y ait un autre appareil avec lequel communiquer. Avec Bluetooth, il peut s'agir de n'importe quel appareil à portée de communication. La première étape de l'établissement de la communication consiste à trouver cet autre appareil. Le module RN4678 dispose de commandes d'interrogation, la commande "F,<numéro>", qui peut être utilisée pour s'informer sur les autres dispositifs à portée. Les commandes "F" envoient un signal qui est capté par tout appareil Bluetooth à portée. Ces appareils Bluetooth peuvent alors choisir de répondre à cette demande pour faire savoir à l'appareil demandeur qu'ils sont là. Les appareils répondent à une demande en envoyant l'adresse de leur appareil, qui peut ensuite être utilisée pour contacter le système Bluetooth récepteur.

Les commandes d'enquête

Tous les détails des commandes d'interrogation se trouvent dans le "RN4678 Bluetooth® 4.0 Dual Mode Module User's Guide" (Guide de l'utilisateur du module Bluetooth® 4.0 à double mode). Il est inclus dans les fichiers de ressources. Chaque fois qu'une nouvelle commande est introduite, reportez-vous au document sur le jeu de commandes pour comprendre la nouvelle commande. L'acquisition de connaissances sur le jeu de commandes du RN4678 constitue un ensemble de connaissances important. Les commandes "F,0" ou "F,4" amènent le module Bluetooth à transmettre un signal d'interrogation auquel d'autres dispositifs Bluetooth peuvent répondre. La commande "F,4" renvoie des informations sur le dispositif sans son nom. Nous utiliserons cette commande pour réduire la quantité d'informations à traiter et à afficher. Le paramètre facultatif <devclass> est utilisé pour filtrer les appareils à rechercher et sera traité ultérieurement. Pour l'instant, la commande de base "F" peut être utilisée.

En invoquant la commande "F,4", nous recevons les réponses suivantes :

- 123456789012 - L'adresse hexadécimale à 12 chiffres de l'appareil Bluetooth qui répond à la demande. Notez que tous les appareils à portée peuvent ne pas répondre. Les appareils Bluetooth peuvent être configurés pour ignorer la commande d'interrogation. Ce point sera abordé dans le prochain chapitre.
- Enquête terminée - Reçu lorsque l'enquête est terminée (par défaut 10 secondes) Err - Une erreur s'est produite

Un certain nombre de réponses peuvent être reçues : une réponse d'adresse de chaque dispositif répondant à la demande, et un message de confirmation une fois que toutes les réponses ont été reçues. Une vérification peut être effectuée et, s'il ne s'agit pas d'une réponse d'erreur, l'adresse peut être affichée.

Paramètres d'enquête supplémentaires

Les commandes d'interrogation peuvent prendre des paramètres supplémentaires facultatifs qui peuvent être utilisés pour filtrer les réponses.

F,<0-5>,<valeur 2>

Où <0 - 5> est l'indice du mode d'interrogation

et <valeur 2> dépend de la valeur du mode d'interrogation

Exemple : **F,2,001F00** (attend 3 octets), scanne l'heure par défaut (définie par la commande **SL**) à l'aide du COD 0x1F00.

Notez que les index 0 à 4 du mode d'interrogation ne s'appliquent qu'à Bluetooth Classic et que seul l'index 5 du mode d'interrogation permet un balayage BLE.

Exemple de découverte

Il y a 4 appareils Bluetooth à portée, nommés Blue1-Blue4. Les appareils sont configurés comme suit :

Dispositif	Adresse	Configuration
Bleu1	00809872F3D4	Accepte les commandes d'enquête
Bleu2	00809864DD44	Accepte les commandes d'enquête
Bleu3	00809894E620	N'accepte pas les ordres d'interrogation
Bleu4	00809894E5D5	Accepte les commandes d'enquête

Le dispositif Blue1 transmet la commande

d'interrogation. Les réponses suivantes sont

reçues :

- 00809864DD44
- 00809894E5D5

Exercice 1 : découverte des périphériques Bluetooth

Introduction

Pour pouvoir communiquer avec un dispositif Bluetooth, il est nécessaire de savoir qu'il existe. La première étape des communications Bluetooth consiste donc à s'enquérir des dispositifs présents et à pouvoir les identifier afin d'établir des communications avec eux.

Objectifs

- Développez un programme qui effectue une enquête sur les périphériques Bluetooth et affiche les adresses de tous les périphériques trouvés sur l'écran LCD.
- Afficher un message d'achèvement une fois que toutes les réponses ont été reçues.

Pré-requis

- Une compréhension des composants et icônes Flowcode standard, tels que les icônes de décision, le composant LCD et les macros.
- Une compréhension de l'envoi de commandes Bluetooth (voir le fichier d'aide du composant Bluetooth dans Flowcode pour plus de détails sur les macros impliquées).
- Une compréhension de la réception et de l'extraction des données des messages (voir le fichier d'aide du composant Bluetooth pour plus de détails sur les macros concernées et un aperçu des stratégies pertinentes).

Exigences en matière de matériel et de logiciel

Le matériel suivant est nécessaire :

- Solution Bluetooth pour le programme d'exercices.
- 1 ou plusieurs périphériques Bluetooth supplémentaires pour la démonstration du programme. Note : La deuxième carte Bluetooth de la solution Bluetooth peut être utilisée pour cet exercice. Le programme BT_EX1_NODE_A.FCFX peut être utilisé pour configurer la carte.
- Installez Flowcode et les composants matériels comme indiqué dans la section Démarrage.

Informations sur l'exercice

La commande d'interrogation est "F,4", voir le jeu de commandes dans le document RN4678 User Guide pour plus de détails sur les commandes du RN4678.

Les réponses attendues sont les suivantes :

- Une adresse hexadécimale à 12 chiffres pour les appareils qui répondent.
- Un message d'erreur.

"Trouvé x" une fois l'enquête terminée.

Les objectifs peuvent être décomposés comme suit :

Tâches

- Envoyer la commande d'interrogation.
- Vérifier les réponses.
- Afficher l'adresse de tout appareil qui répond.
- Afficher le message d'achèvement une fois l'enquête terminée.

Résultats de l'apprentissage

Les principaux résultats d'apprentissage de cet exercice sont les suivants

- Comprendre les commandes de base des appareils Bluetooth
- Comprendre le processus de découverte
- Créer et envoyer des commandes
- Vérification et récupération des réponses

Vérification des réponses spécifiques

Tâches supplémentaires

Les tâches suivantes sont des tâches supplémentaires qui peuvent être mises en œuvre pour ajouter des caractéristiques supplémentaires ou améliorer la fonctionnalité du programme de base :

- Mettez en œuvre votre propre comptage du nombre d'adresses trouvées.
- Effectuer un contrôle d'erreur de base des valeurs de retour du composant Flowcode
- Vérifier s'il y a une réponse d'erreur.
- Récupérer et afficher le(s) nom(s) de l'appareil en utilisant la commande "F,0" au lieu de "F,4"

Mise en œuvre pratique : Découverte des appareils Bluetooth

À découvrir et découvrable

Notez que les programmes créés pour l'exercice 1 doivent être utilisés en conjonction avec le programme créé dans l'exercice 2 pour former une paire de planches découvrables et découvrautes. L'utilisation des deux exercices en tant que paire liée permet aux élèves d'explorer la découverte et la découvrabilité en utilisant le même programme de l'exercice 1 dans les deux exercices, ce qui leur donne une continuité et la possibilité de voir l'exercice précédent en action.

Les deux programmes de l'exercice 1 et de l'exercice 2 constitueront la partie centrale de la plupart des exercices suivants, des commandes supplémentaires étant ajoutées au fur et à mesure de la progression du cours. Il est donc possible d'adopter une approche de continuité dans laquelle les programmes finaux des exercices précédents peuvent souvent être utilisés comme point de départ pour le suivant. Cette approche permet aux étudiants de voir comment le processus s'écoule et se développe au fur et à mesure qu'ils progressent dans les exercices.

Le programme que les élèves doivent écrire dans l'exercice 1 consiste à découvrir tous les autres périphériques Bluetooth à portée. Pour cela, il faut qu'un programme soit actif dans le périphérique Bluetooth correspondant et que ce dernier puisse être découvert.

Le composant Flowcode BL0170 RN4678 dispose d'une fonction Initialise qui configure le périphérique Bluetooth en fonction des paramètres des propriétés du composant. L'exemple d'exercice 1, nœud A, définit l'appareil comme découvrable sans qu'il soit nécessaire d'ajouter du code dans le programme.

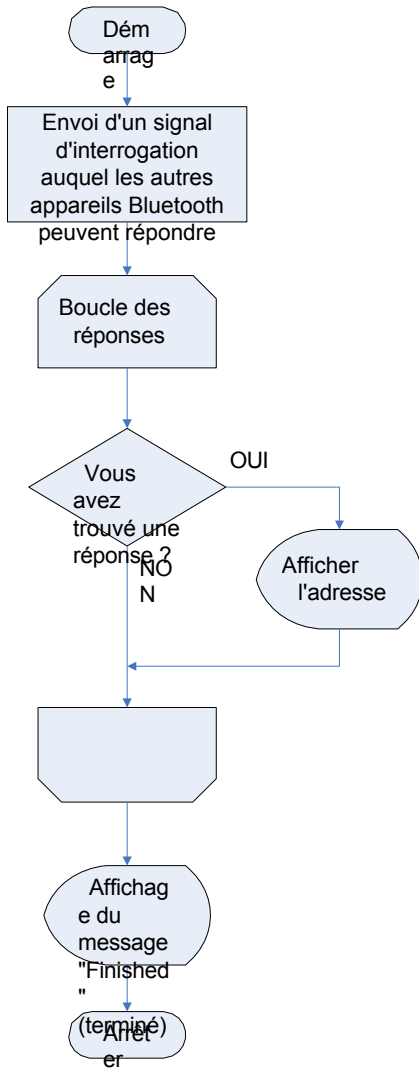
Cela permet à l'étudiant de commencer avec un appareil détectable avant de passer à l'apprentissage des commandes de l'appareil.

Planification du programme

Avant de rédiger un programme, il faut avoir une idée de ses objectifs et un aperçu de la manière dont il sera mis en œuvre. Vous trouverez ci-dessous un plan rudimentaire pour un programme d'enquête.

Le plan consiste à envoyer le signal d'interrogation, puis à vérifier en boucle s'il y a des réponses, à afficher le message publicitaire et à le transmettre à l'autorité compétente.

de tout appareil qui répond, et le message d'achèvement lorsque la demande est terminée.



Le matériel nécessaire à la réalisation du plan comprend un module Bluetooth et un dispositif d'affichage capable d'afficher les 12 chiffres de l'adresse complète des appareils qui répondent. Nous utiliserons ici l'écran LCD BL0169.

Les exigences en matière de logiciel sont les suivantes

- Envoi d'un signal d'interrogation
- Vérification des réponses
- Obtention des données de réponse Affichage des données.

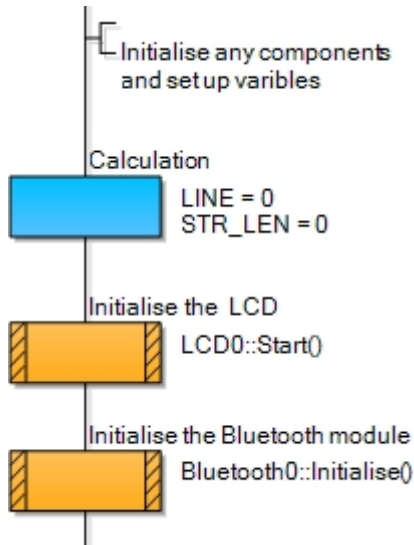
Macros requises

En supposant que l'étudiant ait une connaissance de base de Flowcode et des composants de base tels que l'écran LCD, seules les macros spécifiques au Bluetooth doivent être abordées.

Les macros suivantes du composant Bluetooth seront nécessaires pour le programme :

- Initialiser
- EnterCommandMode
- Envoyer la commande
- ReceiveByte

Initialisation



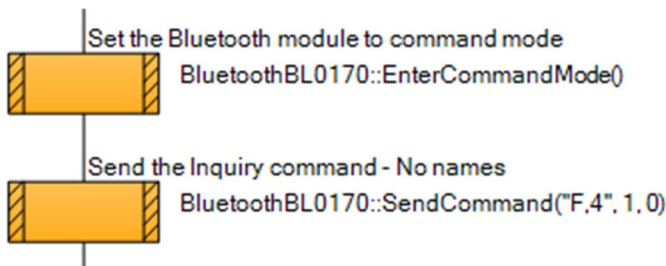
Le déroulement du programme de base consiste à commencer par initialiser tous les composants et à définir toutes les valeurs variables par défaut. Ajoutez une macro de démarrage LCD et une macro d'initialisation Bluetooth.

La macro Bluetooth Initialise est nécessaire pour tous les programmes qui utilisent le composant Bluetooth. Le meilleur endroit pour ajouter des macros d'initialisation est au début du programme.

Ajoutez une icône de calcul et saisissez les valeurs par défaut des variables. Ces valeurs peuvent être mises à jour lors de la construction du programme afin d'ajouter les nouvelles variables nécessaires. La définition de valeurs par défaut est une bonne pratique pour éviter les erreurs dues à des valeurs erronées.

Envoi d'une commande

Une fois que tous les composants et toutes les variables ont été configurés, la première tâche peut être envisagée, à savoir l'envoi de la commande d'interrogation.

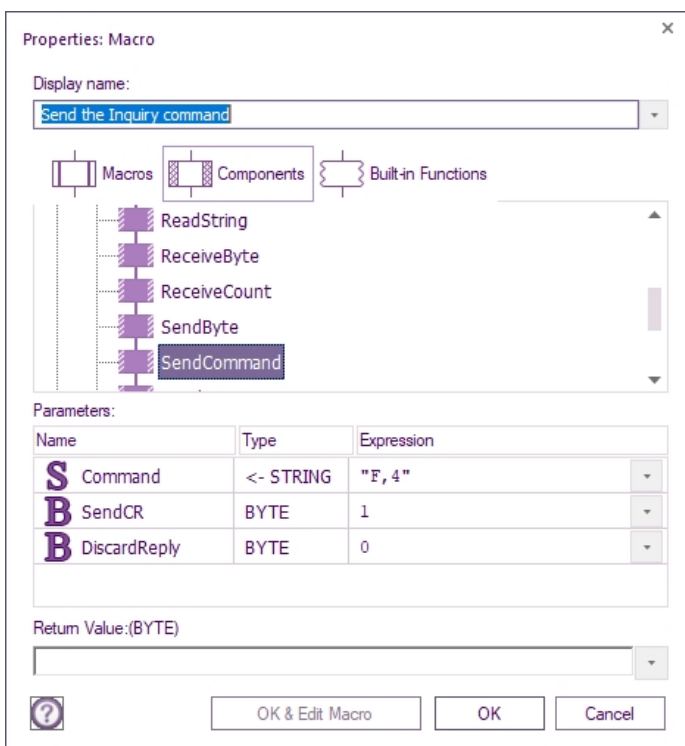


Avant qu'une commande puisse être envoyée au module Bluetooth, celui-ci doit être mis en mode commande à l'aide de la fonction EnterCommandMode du composant.

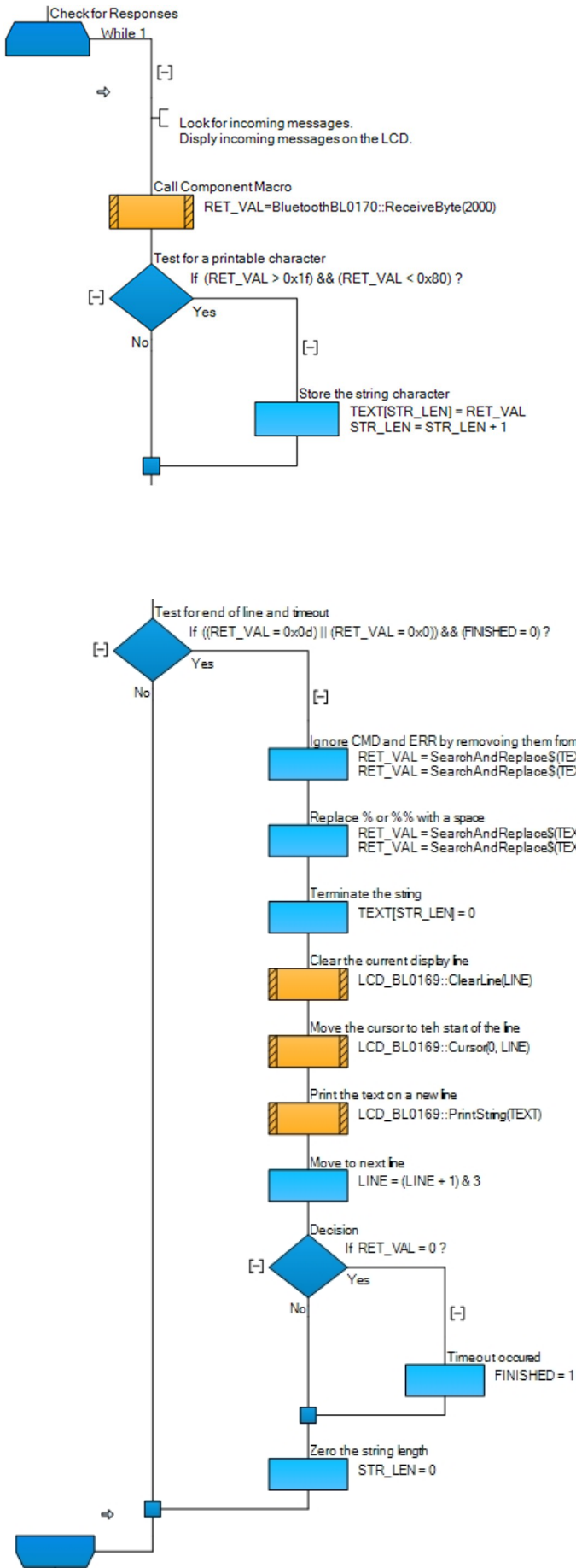
La commande d'interrogation peut maintenant être envoyée à l'aide de la fonction SendCommand. Entrer "F,4" comme commande et régler SendCR sur la valeur 1 et DiscardReply sur 0.

Cette fonction renvoie la valeur 1 si la commande est envoyée avec succès. Mais la valeur de retour ne sera pas vérifiée dans cet exemple pour éviter de rendre ce premier programme trop complexe. Cependant, en général, il est bon d'utiliser la valeur de retour pour un contrôle d'erreur de base afin de s'assurer que les commandes ont été envoyées.

L'écran LCD PrintString peut être utilisé pour afficher des messages sur l'état d'avancement des travaux, tels que "Démarrage de l'enquête".



Vérification des réponses



Une fois la demande envoyée, l'étape suivante consiste à écouter les réponses.

Il peut y avoir un certain nombre de réponses, de sorte qu'une structure en boucle est nécessaire pour les passer toutes en revue, ce qui est fait en mettant en place une boucle continue "while 1".

Un appel à ReceiveByte renvoie les caractères au fur et à mesure qu'ils sont reçus du module Bluetooth. Nous plaçons ce caractère dans la variable RET_VAL.

Nous testons ensuite cette variable pour vérifier qu'il s'agit d'un caractère imprimable (voir une table ASCII pour plus de détails). Si c'est le cas, nous la stockons dans un tampon temporaire nommé TEXT.

Les chaînes de texte reçues du module Bluetooth se terminent généralement par un retour chariot, représenté par le nombre 0x0d, exprimé en notation hexadécimale.

Lorsque nous recevons ce caractère, nous affichons la chaîne reçue qui est contenue dans notre tampon TEXT.

La fonction PrintString s'attend à ce que la chaîne de texte se termine par un caractère de valeur NULL ou 0. Nous commençons donc par insérer un à la fin de notre tampon TEXT.

Si un NULL est renvoyé, la fonction d'impression ne sera pas accessible jusqu'à ce qu'un nouveau caractère imprimable soit détecté.

Vous remarquerez également dans cet exemple d'organigramme que nous comparons d'abord la mémoire tampon TEXTE avec les réponses attendues que nous ne voulons pas afficher, auquel cas nous supprimons les réponses non souhaitées de la mémoire tampon TEXTE.

La variable STR_LEN garde la trace de la longueur de notre texte entrant, il faut donc réinitialiser la valeur à 0 avant de retourner dans la fonction

Exécution du programme complet TERMINÉ

Le programme complet se trouve dans l'exemple BT_EX1_NODE_B.FCFX.

Ce programme est téléchargé sur la carte processeur cible de l'exercice avec une carte Bluetooth BL0170 connectée au port B dans le cas du processeur PIC BL0011.

En outre, au moins un autre dispositif Bluetooth doit être actif et détectable. Il peut s'agir de l'autre carte de la solution Bluetooth ou d'un périphérique Bluetooth tiers.

Il faut cependant savoir que tous les appareils Bluetooth ne répondront pas. Lors de la démonstration du programme, il peut être prudent de tester les appareils Bluetooth avant de les utiliser pour la démonstration afin de vérifier s'ils peuvent être découverts ou non.

Par exemple, si vous avez un téléphone portable Bluetooth, vous devrez activer le Bluetooth et régler la visibilité du téléphone pour qu'il ne soit pas caché.

Compilez et téléchargez le programme sur le microcontrôleur de l'une des cartes Bluetooth. Débranchez brièvement l'alimentation, rebranchez-la et appuyez sur l'interrupteur de réinitialisation de la carte du programmeur. Cela permet de s'assurer que le module Bluetooth se réinitialise et redémarre correctement.

L'écran LCD affiche des messages de progression et l'adresse à 12 chiffres de tous les dispositifs trouvés, et se termine par un message indiquant le nombre de dispositifs trouvés.

Possibilité de découverte

Théorie : Possibilité de découverte

La commande d'interrogation décrite dans le chapitre précédent est envoyée à tous les appareils Bluetooth. Cependant, tous les appareils ne répondent pas au signal d'interrogation. En effet, les appareils Bluetooth peuvent être configurés pour se cacher des autres appareils.

Les appareils peuvent également être configurés pour être découverts, auquel cas ils répondront à la commande d'interrogation.

Configuration pour le démarrage

Les appareils Bluetooth disposent d'une série d'options de configuration qui vont des clés de passage et du cryptage à la vitesse de transmission et au nombre de sonneries avant de répondre. Les paramètres par défaut sont généralement utilisés pour que les appareils fonctionnent normalement "prêts à l'emploi". Cependant, il est bon de configurer les appareils avec les paramètres corrects au démarrage pour s'assurer que l'appareil fonctionne comme souhaité.

Dans le module RN4678, ces options peuvent être configurées à l'aide de diverses commandes, détaillées dans le Guide de l'utilisateur du module bi-mode RN4678 Bluetooth® 4.0. Ces commandes peuvent être regroupées en Set Commands (commandes de configuration), Get Commands (commandes d'obtention) et Action Commands (commandes d'action).

Utilisation des commandes Set

Les commandes de paramétrage modifient les configurations du RN4678 et prennent effet après un redémarrage via la commande R,1, une réinitialisation matérielle ou un cycle d'alimentation. Certaines commandes de réglage ne s'appliquent qu'au fonctionnement Bluetooth Classic ou BLE. Les commandes de paramétrage ont des paramètres et comprennent des éléments tels que :

- Mode de fonctionnement Bluetooth
- Méthode d'authentification
- Réglage par défaut
- Classe et ID de l'appareil
- Fenêtre d'interrogation
- Nom de l'appareil
- Code de l'appareil
- PIN de sécurité
- Fiabilité du flux de données
- Adresses distantes
- Nom du service
- Vitesse de transmission UART
- Activer ou désactiver l'invite de commande

Toutes les modifications de configuration apportées par les commandes Set restent dans la mémoire non volatile (NVM) et survivent au cycle d'alimentation. Toute modification de la configuration doit prendre effet après un redémarrage.

Utilisation des commandes Get

Les commandes Get permettent de récupérer et d'afficher les informations stockées dans l'appareil. La plupart de ces commandes n'ont pas de paramètres supplémentaires. Ces commandes renvoient des valeurs de réglage et des informations sur l'état de l'appareil.

Utilisation des commandes d'action

Les commandes d'action permettent d'effectuer des actions telles que des balayages d'enquête, la connexion et l'entrée ou la sortie du mode de commande. Certaines commandes d'action ne s'appliquent qu'à Bluetooth Classic ou BLE.

Toutes les commandes Action prennent effet immédiatement mais n'ont pas d'effet après un cycle d'alimentation.

Rendre un appareil détectable

Pour cet exercice, nous souhaitons que l'appareil réponde automatiquement, qu'il puisse être découvert et connecté, et que nous puissions lui envoyer des données.

Pour ce faire, la séquence d'événements et les paramètres de commande sont les suivants :

Entrer dans le mode commande

Le composant Flowcode RN4678 dispose d'une fonction qui met cela en œuvre, nommée "EnterCommandMode". Cette fonction envoie la séquence requise de caractères \$\$\$ à l'appareil.

Attendre l'exécution de la commande

Après chaque demande de commande, nous devons attendre que l'appareil confirme qu'il est prêt à accepter une nouvelle commande. L'appareil demande une commande avec la chaîne de texte CMD> . Nous utilisons donc la fonction Flowcode RN4678 WaitForStringValue("CMD> ") pour attendre que l'appareil soit prêt à recevoir une nouvelle commande.

Vitesse de transmission et réglages

Il existe une commande, SU,<Baud Rate Index>, qui définit le débit en bauds de la commande série. Toutefois, cette valeur est réglée en usine sur 115 200 et doit donc être laissée à cette valeur par défaut pour les exercices de ce cours.

Mode d'authentification

Pour les exercices initiaux, nous avons réglé le mode d'authentification sur le mode "Just Works". Ce mode fonctionne sans qu'il soit nécessaire d'afficher ou de saisir un code de sécurité. La commande pour cela est **SA,2**

Mode esclave

Lorsque certaines commandes du mode salve sont exécutées, le Bluetooth active automatiquement le mode "salve".

Activation de la configuration mise à jour

Maintenant que nous avons utilisé les commandes Set pour modifier les paramètres, nous devons redémarrer l'appareil. Toute modification apportée à la configuration de l'appareil à l'aide des commandes Set ne prend effet que lorsque l'appareil est redémarré. C'est ce que nous faisons avec la commande **R,1** Action.

Attendre le redémarrage et quitter le mode commande

Après le redémarrage, nous quittons le mode commande en utilisant la fonction Flowcode RN4678 LeaveCommandMode.

Résumé des commandes de configuration de base

Voici un ensemble de commandes simples permettant de découvrir l'appareil :

Commandement	Description
SA,2	Mode Secure Simple Pairing (SSP) "Just Works". Ce mode fonctionne sans qu'il soit nécessaire d'afficher ou d'entrer un code de sécurité.
R,1	Redémarrez l'appareil pour appliquer la nouvelle configuration.

Introduction

- Pour qu'un dispositif soit trouvé, il doit être détectable. Un appareil Bluetooth qui est détectable est capable de répondre à une commande d'interrogation d'un autre appareil Bluetooth.

Objectifs

- Développez un programme qui configure le périphérique Bluetooth pour qu'il puisse être découvert. Il s'agit de l'exercice complémentaire à l'exercice 1.

Pré-requis

- Une compréhension du processus de découverte tel qu'il est détaillé dans l'exercice 1. En particulier, la capacité à reconnaître une commande d'enquête.
- Compréhension de la réception et de l'extraction des données des messages (voir le fichier d'aide du composant Bluetooth pour plus de détails sur les macros concernées et un aperçu des stratégies pertinentes).
- Exigences en matière de matériel et de logiciel
- Le matériel suivant est nécessaire :
- Solution Bluetooth pour le programme d'exercices.
- 1 ou plusieurs périphériques Bluetooth supplémentaires pour la démonstration du programme. Note : La deuxième carte Bluetooth de la solution Bluetooth peut être utilisée pour cet exercice. Le programme de l'exercice 1 peut être utilisé pour envoyer la commande d'interrogation, Information de l'exercice

Les objectifs peuvent être décomposés comme suit :

- Tâches
- Envoyez les commandes de configuration.
- Vérifier les messages.
- Afficher les messages éventuels.
- Structures de contrôle :
- Vérification et affichage des réponses.
- Résultats de l'apprentissage
- Les principaux résultats d'apprentissage de cet exercice sont les suivants
- Comprendre le processus de découverte.
- Comprendre comment configurer l'appareil RN4678.
- Compréhension des commandes Set et Action.
- Vérification et récupération des réponses.
- Tâches supplémentaires
- Se référer aux paramètres de commande du manuel du module Bluetooth et faire en sorte que le module Bluetooth ne soit pas détectable.
- Quelles sont les implications pour la personne qui utilisera l'équipement après vous ?
- En utilisant les programmes des exercices 1 et 2, notez les adresses Bluetooth de chacun des modules Bluetooth - vous en aurez besoin dans l'exercice suivant.

Mise en œuvre pratique : Découvrabilité

L'exercice 2 accompagne l'exercice 1 dans la mesure où, après avoir montré le fonctionnement de la découverte des appareils dans l'exercice 1, l'étape suivante consiste à montrer comment rendre les appareils détectables. En outre, les méthodes de configuration et l'utilisation des commandes Set et Action sont abordées.

Il est important de souligner que les commandes utilisées ici sont propres au module spécifique utilisé (le dispositif Microchip RN4678) et sont donc des caractéristiques de la puce RN4678 et non une partie inhérente de Bluetooth. Des puces différentes peuvent avoir des structures de commande et des registres différents, voire n'en utiliser aucun. Toutefois, les principes généraux sont les mêmes.

Avant de commencer à connecter le câble USB au deuxième système Bluetooth, vous pouvez laisser le programme de l'exercice 1 dans le premier système Bluetooth et nous l'utiliserons pour tester si le programme créé ici fonctionne.

Développement continu

Les programmes Étudiants créés pour l'exercice 1 peuvent être utilisés en conjonction avec le programme créé dans cet exercice pour former une paire de tableaux découvrables et découvrants.

Les deux programmes de l'exercice 1 et de l'exercice 2 formeront la partie centrale de la plupart des exercices suivants, des commandes supplémentaires étant ajoutées au fur et à mesure de la progression du cours. Il est donc possible d'adopter une approche de continuité dans laquelle le programme final des exercices précédents peut souvent être utilisé comme point de départ pour le suivant.

Configuration de l'appareil Bluetooth

Pour cet exercice, le périphérique Bluetooth doit être configuré de manière à pouvoir être découvert et connecté. Nous ne nous y connecterons pas dans le cadre de cet exercice, mais nous pouvons tout de même mettre en place cette configuration de base dès maintenant.

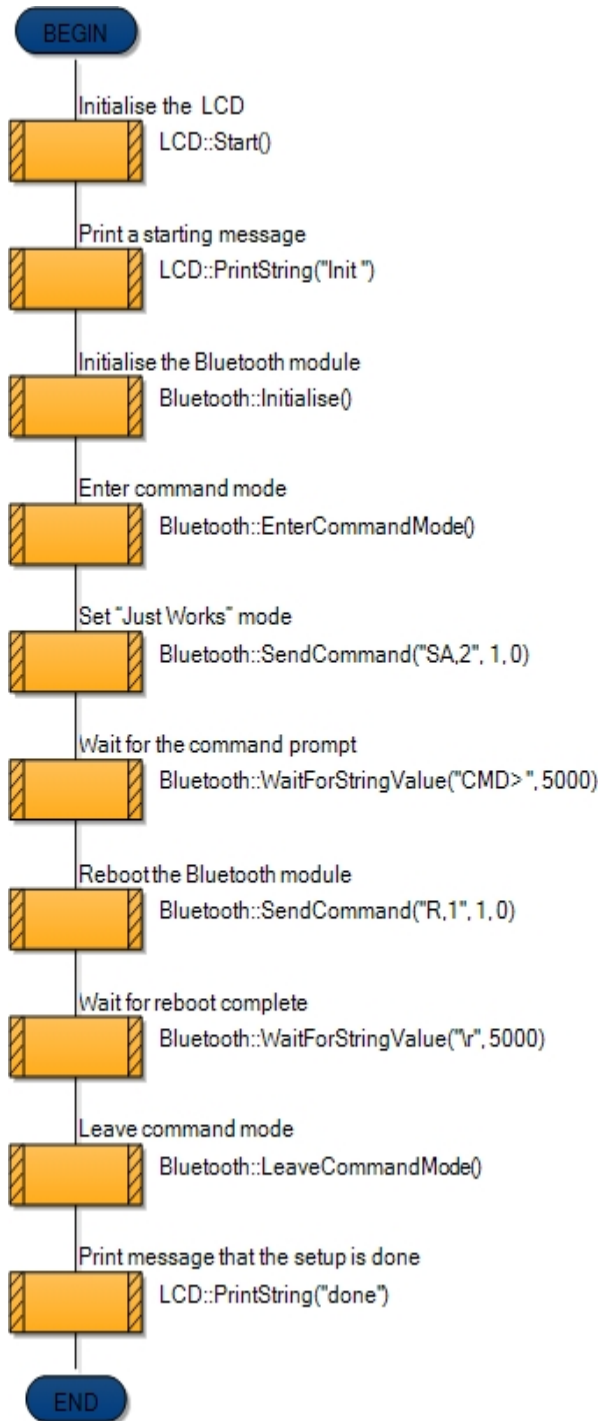
Pour cet exercice, utilisez les commandes de configuration suivantes :

SA,2

R,1

Ces commandes permettent à l'appareil de répondre immédiatement et d'être à la fois détectable et connectable.

Création du programme



Les macros *SendCommand* et *WaitFor-StringValue*, décrites précédemment, sont utilisées pour configurer le module Bluetooth, comme dans l'exemple illustré ici.

Exécution et démonstration du programme

Le programme construit ici peut être utilisé avec le programme d'enquête construit dans l'exercice 1 nœud B.

Le programme du nœud B de l'exercice 1 peut être utilisé dans une carte Bluetooth, et ce programme de l'exercice 2 peut être utilisé avec le nœud A dans une deuxième carte Bluetooth.

- Créez le programme en notant l'adresse de 12 chiffres de la carte Bluetooth qui sera utilisée avec le programme de l'exercice 2.
- Téléchargez et exécutez l'exercice 2 sur la carte Bluetooth dont vous avez noté l'adresse.
- Téléchargez et exécutez le programme d'enquête créé dans l'exercice 1 sur l'autre carte.
- Surveillez l'écran LCD de la carte à découvrir. Lorsque la commande d'interrogation est envoyée, elle s'affiche sur l'écran LCD.

Surveillez l'écran LCD de la carte du programme d'interrogation. Lorsque la carte détectable reçoit la commande d'interrogation, elle répond en renvoyant son adresse de 12 bits (le nombre que vous avez noté plus tôt). Vérifiez sur l'écran LCD que l'adresse est affichée.

Connexion des appareils Bluetooth

Théorie : Connexion - Adresses

Comme il n'y a pas de connexion physique entre les deux appareils, toutes les communications peuvent être captées par n'importe quel autre appareil Bluetooth. Un problème immédiat est de savoir comment communiquer avec un appareil particulier sans communiquer par inadvertance avec d'autres appareils.

Tous les appareils Bluetooth Classic ont une adresse (adresse MAC), un numéro hexadécimal unique à 12 chiffres. Cette adresse est le même numéro que celui qui est renvoyé en réponse à une commande d'interrogation. Cette adresse peut être soit récupérée via une commande d'interrogation, soit stockée dans le programme si une adresse de dispositif spécifique doit être utilisée et est connue à l'avance.

Si un dispositif Bluetooth est connectable et que l'adresse est connue, d'autres dispositifs Bluetooth peuvent établir une connexion avec ce dispositif. En général, les appareils sont détectables et connectables, de sorte que l'adresse peut être déterminée à partir d'une commande d'interrogation. Toutefois, il est possible que des dispositifs soient connectables, mais non détectables. Par exemple, un ensemble de téléphones conçus pour communiquer uniquement entre eux peut être configuré pour être connectable mais non détectable. La paire de téléphones a l'adresse de chaque téléphone en mémoire, de sorte qu'une commande de découverte n'est pas nécessaire, et le téléphone est prêt à commencer à communiquer avec son partenaire.

La commande de connexion de base est la suivante :

C,<bt_addr>

Où <bt_addr> est l'adresse hexadécimale à 12 chiffres de l'appareil avec lequel la connexion doit être établie.

L'adresse MAC Bluetooth Classic des modules RN4678 est définie en usine et ne peut être modifiée.

Introduction

Si l'adresse d'un appareil Bluetooth est connue et que l'appareil a été configuré pour être connecté, il est possible de se connecter à cet appareil.

Objectifs

- Développez un programme dans un système Bluetooth (nœud B) qui se connecte à un autre système Bluetooth (nœud A) et affiche la réponse pour montrer que la connexion a réussi.

Développez un programme dans un système Bluetooth (nœud A) qui permet à un autre système Bluetooth (nœud B) de s'y connecter. Ce programme est fourni dans le fichier BT_EX3_Node_A.fcfx.

Pré-requis

Une compréhension du processus de découverte tel qu'il est détaillé dans l'exercice 1 et l'exercice 2.

Exigences en matière de matériel et de logiciel

Le matériel suivant est nécessaire :

- Solution Bluetooth pour le programme d'exercices.

1 ou plusieurs périphériques Bluetooth supplémentaires pour la démonstration du programme. Note : La deuxième carte Bluetooth de la solution Bluetooth peut être utilisée pour cet exercice. Téléchargez le programme BT_EX3_Node_A.fcfx dans ce deuxième système.

Informations sur l'exercice

La commande Initier la connexion est :

Commandement	Description
C,<bt_addr>	Où <bt_addr> est l'adresse de la deuxième carte Bluetooth, le nœud A.

Mettez en place deux programmes : l'un dans le nœud B pour se connecter à un second dispositif Bluetooth dans le nœud A. Lorsque vous émettez la commande de connexion, le nœud récepteur envoie l'accusé de réception. Affichez tous les messages sur l'écran LCD des nœuds.

Résultats de l'apprentissage

Les principaux résultats d'apprentissage de cet exercice sont les suivants

Connexion à un autre appareil Bluetooth.

Autres travaux

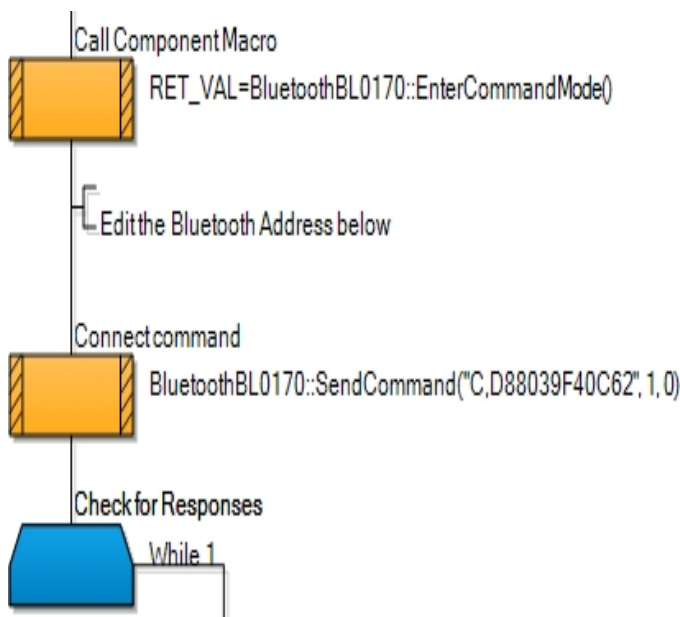
L'écran LCD ne dispose que de 20 caractères par ligne. Concevez une extension du code dans BT_EX3_Node_A.FCFX pour que tout dépassement de texte soit affiché sur la ligne suivante de l'écran.

Comme pour les exercices précédents, les étudiants doivent en fait développer deux programmes - un dans le dispositif Bluetooth initialisant la communication, et le second dans le dispositif recevant la communication initiale.

Cet exercice est rapide et simple du point de vue de la charge de travail, mais il comporte un certain nombre de pièges techniques dont il faut être conscient.

1) Adresse de l'appareil. Le programme d'exemple BT_EX3_Node_B.FCFX utilise l'adresse d'un périphérique Bluetooth utilisé pour les tests chez Matrix. Cette adresse devra être modifiée pour correspondre à l'adresse de la carte qui sera utilisée avec le programme d'exemple. Pour cela, vous devez utiliser l'adresse que vous avez trouvée dans l'exercice 2 et l'entrer en tant que paramètre de la commande

SendCommand comme ci-dessous.



Indiquez clairement quel est le nœud d'envoi et quel est le nœud de réception, ainsi que leurs adresses respectives.

2) L'appareil Bluetooth auquel il faut se connecter doit avoir été configuré pour être connectable. Une fois encore, le programme BT_EX3_Node_A.FCFX a été conçu dans cette optique.

3) Une fois la connexion établie, il faut pouvoir vérifier que la connexion a bien été établie. Le programme BT_EX3_Node_A.FCFX est configuré pour afficher tous les messages envoyés, de sorte que nous pouvons simplement envoyer un message et vérifier qu'il est reçu par l'autre appareil. Le programme que vous écrivez doit afficher le message de retour du nœud A.

Aucun contrôle d'erreur n'est effectué pour l'instant. Le programme suppose que la connexion aboutira, ce qui n'est pas toujours le cas. Une meilleure méthode consisterait à utiliser les réponses pour vérifier les erreurs du processus. Les réponses et le contrôle des erreurs seront traités dans un chapitre ultérieur.

Réinitialisation des systèmes

Le module Bluetooth n'est pas équipé d'un bouton de réinitialisation. Le seul mécanisme de réinitialisation consiste à couper l'alimentation. Si vous avez programmé le module Bluetooth pour qu'il effectue une activité, le fait d'appuyer sur le bouton de réinitialisation de la carte du processeur ne réinitialisera pas nécessairement le module Bluetooth.

C'est pourquoi, lors du développement de paires de programmes, il **peut** être nécessaire de couper l'alimentation du système et de redémarrer les modules Bluetooth.

Théorie : Passkeys et Connecting

Bluetooth communique par radio, un moyen de transmission intrinsèquement non sécurisé. Tout appareil à portée de transmission peut recevoir les signaux envoyés à n'importe quel autre appareil. L'utilisation d'une adresse unique résout le problème de la spécification de l'appareil avec lequel vous avez l'intention de communiquer, mais d'autres méthodes sont nécessaires pour empêcher l'accès non autorisé à un appareil.

Le système de sécurité de base utilisé avec Bluetooth s'appelle l'appairage. L'appairage consiste à connecter deux appareils l'un à l'autre en utilisant à la fois l'adresse de l'appareil et la "clé de liaison" secrète de l'appareil, connue sous le nom de "Passkey". La clé de liaison ne peut pas être récupérée d'un autre appareil de la même manière que l'adresse. Pour pouvoir vous connecter à un appareil, vous devez connaître le numéro de la clé de passage de cet appareil.

Envoi de la commande Passkey

Pour configurer un appareil afin qu'il puisse être apparié avec un Passkey, il faut configurer l'appareil en question. La commande de configuration d'un Passkey est la suivante :

SP,<clés de passe>

Où <passkey> est le code PIN de la clé. Le Passkey est un numéro à 4 chiffres (pour l'appairage par code pin, un code pin à 6 chiffres est utilisé pour l'authentification SSP dans BLE). En règle générale, les numéros PIN de la clé de sécurité sont fournis dans la documentation qui accompagne un dispositif particulier. Les codes PIN des appareils doivent être conservés en lieu sûr, tout comme les codes PIN des cartes bancaires.

Initiation de l'appariement et de la connexion

Une fois la clé de sécurité définie, la connexion de données peut être établie en lançant la commande de connexion comme suit :

C,<bt_addr>

Où <bt_addr> est l'adresse de l'appareil Bluetooth Classic auquel il faut se connecter.

Dans de nombreux cas, par exemple lorsqu'il s'agit de coupler un téléphone portable à un casque, la clé d'accès figure dans la documentation du produit et le programme vous invite à saisir la clé d'accès.

Exercice 4 : clés d'accès et connexion

Introduction

Les appareils peuvent être appariés les uns aux autres pour communiquer. L'appairage nécessite l'adresse de l'appareil avec lequel il doit être apparié, ainsi qu'une valeur de clé d'accès pour établir le contact. Dans ce cas, vous devrez développer deux programmes - l'un qui détermine la clé de sécurité d'un système, et l'autre qui se connecte à ce système et lui envoie des données.

Objectifs

- Développer des programmes pour deux systèmes Bluetooth qui permettent l'appariement. Développez un programme pour le nœud A qui attribue un mot de passe de "1234", et rendez-le détectable. Développez des routines qui affichent sur l'écran LCD toutes les données envoyées au nœud. Développez un programme pour le nœud B qui s'apparie avec le nœud A en utilisant l'adresse et la clé de passage du nœud A.
- Une fois la connexion établie, une simple communication entre les systèmes montre qu'une communication a lieu, c'est-à-dire qu'un compteur sur le nœud d'envoi compte les données sous la forme d'un octet unique qui est également affiché sur le nœud de réception.

Pré-requis

- Une compréhension du processus de connexion tel qu'il est détaillé dans l'exercice 3.

Compréhension de la création, de l'envoi et de la réception de commandes, comme indiqué dans les exercices 1 et 2.

Exigences en matière de matériel et de logiciel

Le matériel suivant est nécessaire :

- Les deux solutions Bluetooth sont nécessaires pour cet exercice.
- La solution 1, nœud A, sera connectée.

La solution 2, le nœud B, initiera la connexion.

Informations sur l'exercice

Commandement	Description
SP,<clés de passe>	Où <passkey> est la valeur du passkey de l'appareil à appairer.
C,<bt_addr>	Où <bt_addr> est l'adresse de la deuxième carte Bluetooth. Si la commande de connexion réussit, une réponse CONNECT <bd_addr> est renvoyée.

Une fois que la connexion a été établie, il faut passer en boucle et envoyer les chiffres de 0 à 9 pour qu'ils soient affichés sur l'autre appareil.

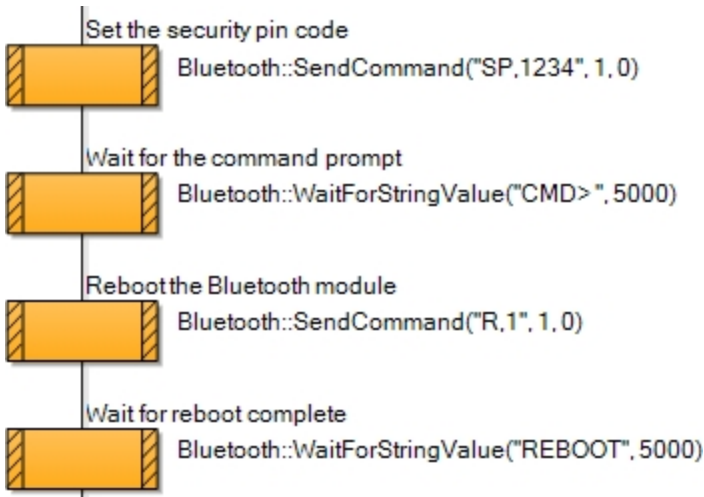
Résultats de l'apprentissage

- Compréhension du processus de connexion.
- Compréhension du rôle et des implications de la clé de sécurité.

Autres travaux

- A l'aide d'un clavier et d'un tableau de variables, développez deux programmes qui permettent de définir le mot de passe du système de réception et le mot de passe utilisé par le système de transmission.

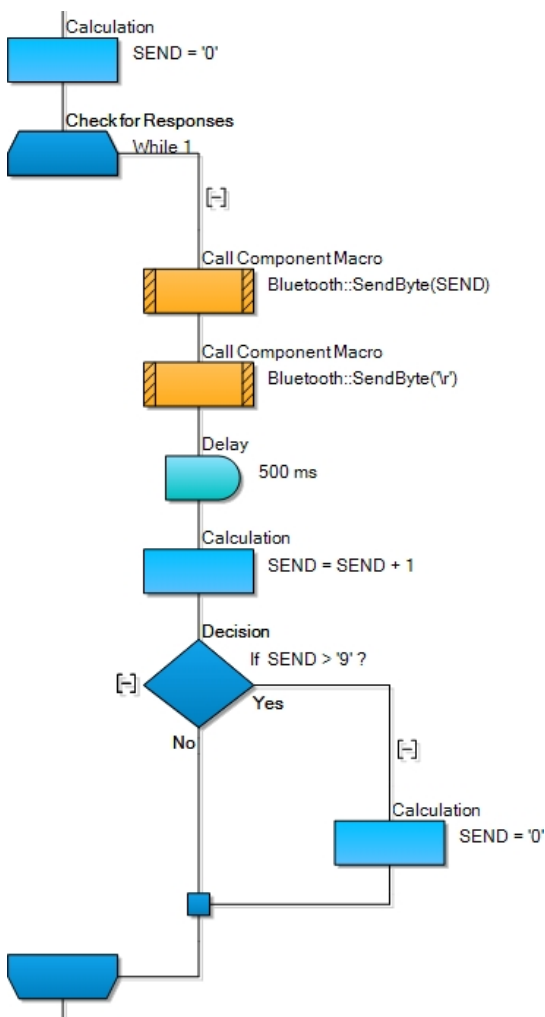
Le programme de base



L'appairage est un processus clé dans les opérations Bluetooth. La clé doit être configurée et, comme pour toutes les commandes "set", elle est activée après un redémarrage de l'appareil, ce qui peut être fait avec la commande "R,1".

Si les détails sont connus à l'avance, ils peuvent être simplement entrés dans le programme et les commandes construites et envoyées. Si le mot de passe et l'adresse ne sont pas connus à l'avance, ils doivent être établis.

L'adresse du dispositif Bluetooth peut être obtenue en téléchargeant le programme Flowcode BT_TEST. Vous pouvez noter les adresses des différentes cartes.



Pour démontrer que la connexion fonctionne, un simple signal répété de 0 à 9 peut être envoyé pour être affiché sur l'appareil récepteur.

Les adresses des appareils peuvent être récupérées à l'aide de la commande Inquiry. Cependant, savoir si nous devons connecter le dispositif 007284972989 ou 00234869030 est une autre affaire. La commande Inquiry peut être étendue pour récupérer des informations supplémentaires au-delà de l'adresse à 12 chiffres. La commande "F,0" permet de récupérer non seulement l'adresse du périphérique, mais aussi son nom amical. Le nom amical est une chaîne de description de l'appareil qui est plus facile à comprendre pour les utilisateurs que l'adresse de l'appareil.

Par exemple :

012345678912, "NodeA"

012345678913, "NodeB"

L'affichage de l'adresse amicale peut aider les utilisateurs à sélectionner le bon appareil dans une liste. En pratique, l'affichage du nom convivial est une simple modification du code de réponse d'affichage de l'exemple 1. En ajoutant l'espace de commande, les 14 premiers caractères peuvent être affichés sur la ligne supérieure de l'écran LCD et le reste sur la ligne inférieure en déplaçant le curseur lorsque les 14 premiers caractères ont été affichés.

En outre, nous pouvons ajouter les 12 caractères d'adresse dans un tableau, en stockant l'adresse actuelle pour une utilisation ultérieure. Cette opération peut être effectuée en même temps que l'impression des caractères. Ajoutez les 12 premiers caractères à l'adresse actuelle.

L'obtention du mot de passe est plus difficile car il ne peut pas être récupéré par une commande. Pour les exemples utilisés dans ce cours, la clé "1234" est utilisée tout au long du cours. Dans la pratique, il peut être nécessaire d'obtenir un code d'accès à partir de la documentation de l'appareil et de le saisir dans le code du programme, ou de le saisir manuellement dans le programme.

Réinitialisation des systèmes

La carte Bluetooth BL0170 n'est pas équipée d'un bouton de réinitialisation. Le seul mécanisme de réinitialisation consiste à remettre l'appareil sous tension ou à lancer la commande de redémarrage (R,1). Si vous avez programmé le module Bluetooth pour qu'il effectue une activité, le fait d'appuyer sur le bouton de réinitialisation de la carte du programmeur ne réinitialisera pas nécessairement le module Bluetooth.

C'est pourquoi, lors du développement de paires de programmes, il peut être nécessaire de couper l'alimentation du système et de redémarrer les modules Bluetooth.

Pour cet exercice, nous vous recommandons de couper l'alimentation des deux systèmes chaque fois que vous téléchargez un programme. Mettez ensuite le système de réception sous tension, appuyez sur reset et laissez-lui quelques secondes pour s'installer. Mettez ensuite le système d'émission sous tension, appuyez sur reset et laissez-lui quelques secondes pour s'installer.

Vérification des réponses

Théorie : Vérification des réponses

Dans certains des exercices précédents, nous envoyons des paramètres ou des commandes et supposons qu'ils seront exécutés. Une méthode plus utile consisterait à surveiller le résultat de la macro du composant Flowcode et à vérifier en outre la réponse du processus.

La plupart des macros du composant Bluetooth ont une valeur de retour qui indique si la demande a abouti ou non. Cette valeur de retour peut être testée et utilisée pour déterminer ce qui doit être fait ensuite, soit continuer avec l'étape suivante du programme, soit sauter à la fin et abandonner.

Lorsqu'une commande est envoyée au dispositif Bluetooth, une réponse est généralement envoyée en guise de réponse à cette commande. Pour de nombreuses commandes et occasions, la réponse sera un message "OK". Cependant, de nombreuses commandes ont des réponses spécifiques. La commande Discover, par exemple, envoie des adresses comme réponses, ainsi qu'un message d'achèvement du processus.

Réponses sollicitées et non sollicitées

Lorsqu'une réponse est le résultat direct de l'envoi d'une commande, il s'agit d'une réponse sollicitée, c'est-à-dire d'une réponse attendue. Par exemple, l'envoi d'une commande SP,1234 entraînera une réponse sollicitée "OK".

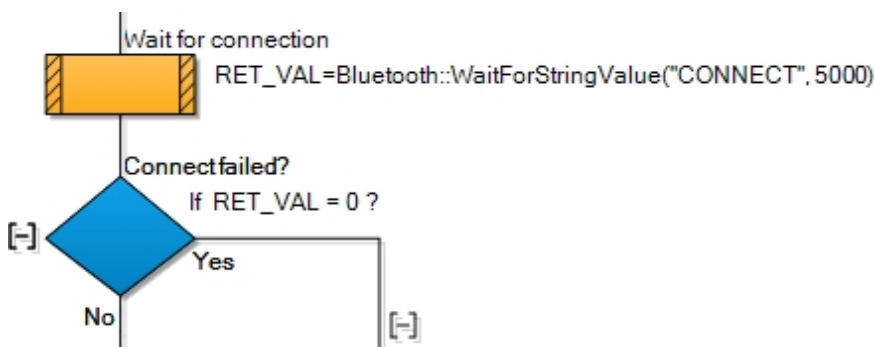
Il arrive qu'une réponse soit envoyée sans qu'il s'agisse d'une commande spécifique. Il s'agit d'une réponse "non sollicitée". Les réponses non sollicitées peuvent être envoyées à différents moments pour indiquer un besoin d'interaction supplémentaire ou pour fournir un retour d'information supplémentaire. Par exemple, les réponses CONNECT envoyées pendant l'établissement des communications pour informer l'autre appareil de l'état actuel de l'opération.

Macros de traitement des réponses

Certaines macros de composants Bluetooth sont disponibles pour faciliter la gestion des réponses :

WaitForStringValue

Cela permet d'interrompre le déroulement du programme jusqu'à la réception d'un message de réponse spécifique ou jusqu'à l'expiration de la période de temporisation. Par exemple, nous attendons ici le message "CONNECT" pendant un maximum de 5 secondes (5 000 mS).



Ces macros nous permettent de lire et donc de tester ou d'afficher tout texte reçu du module Blue tooth.

Par exemple, nous pouvons créer un tampon de texte global MESSAGE et utiliser la macro *ReadString* pour lire la réponse dans ce tampon. *ReadString* a pour paramètre la taille de la mémoire tampon (32 octets dans cet exemple).

The screenshot shows a development environment with two main panels. On the left is the 'Project Explorer' showing a table of global variables. On the right is a flowchart titled 'Display Responses'.

Expression	Value
Globals	
Constants	
Variables	
B CHAR	n/a
B CNT	n/a
B ERROR_VAL	n/a
B FINISHED	n/a
B INDEX	n/a
S MESSAGE[32]	n/a
B RET_VAL	n/a
B STR_LEN	n/a
Locals [DISP_RESP]	
Parameters	

```

graph TD
    BEGIN([BEGIN]) --> Look[Look for incoming messages.  
Disply incoming messages on the LCD.]
    Look --> Go[Go to next line  
LCD_BL0169::Cursor(0, 1)]
    Go --> Read[Call Component Macro  
MESSAGE=BluetoothBL0170::ReadString(32)]
    Read --> Print[Call Component Macro  
LCD_BL0169::PrintString(MESSAGE)]
    Print --> END([END])
    
```

Exercice 5 : vérification des réponses

Introduction

Lorsque le dispositif Bluetooth reçoit une commande, il renvoie une réponse. En vérifiant ces réponses, il est possible de contrôler les erreurs éventuelles et d'attendre les réponses attendues avant de poursuivre une séquence de commandes.

Objectifs

- Développer des programmes pour deux systèmes Bluetooth qui permettent d'établir une connexion. Développez un programme pour le nœud A qui attribue un mot de passe de "1234", et rendez-le détectable. Développez des routines qui affichent toutes les données envoyées au nœud sur l'écran LCD. Vous pouvez utiliser le programme de l'exercice 4 pour cela.
- Développez un programme pour le nœud B qui se connecte au nœud A en utilisant l'adresse du nœud A.

Signaler toute erreur survenant sur le

nœud B Pré-requis

- Une compréhension du processus de connexion tel qu'il est détaillé dans l'exercice 4.

Une compréhension de la macro WaitForStringValue telle qu'elle est décrite à la page précédente et dans les infobulles d'aide du composant Bluetooth.

Exigences en matière de matériel et de logiciel

Le matériel suivant est nécessaire :

- Les deux kits Bluetooth sont nécessaires pour cet exercice.
- L'ensemble 1, nœud A, attendra la connexion. L'ensemble 2, nœud B, initiera la connexion.

Informations sur l'exercice

Commandement	Description
SP,<clés de passe>	Où <passkey> est la valeur du passkey de l'appareil à appairer.
C,<bt_addr>	Où <bt_addr> est l'adresse de la deuxième carte Bluetooth. Si la commande de connexion réussit, une réponse CONNECT <bd_addr> est renvoyée.

Résultats de l'apprentissage

Les principaux résultats d'apprentissage de cet exercice sont les suivants

- Séquences de commandes et de réponses.
- Types de réponses.
- Vérification des erreurs et contrôle des séquences.
- Réponses non sollicitées.

Mise en œuvre pratique : Vérification des réponses

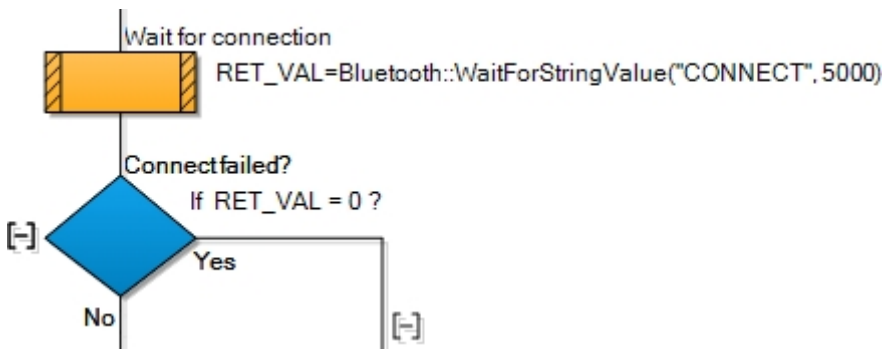
Les deux informations importantes à noter sont les adresses des périphériques Bluetooth et les codes d'accès. Pour cet exercice, on suppose que la clé générale "1234" sera utilisée. Toutefois, les périphériques Bluetooth devront être modifiés en conséquence.

Cette tâche nécessite deux programmes, l'un pour initier la connexion et l'autre pour afficher les données envoyées à l'appareil couplé. Les programmes développés dans l'exercice 4 peuvent être utilisés comme base de l'exercice 5, ce qui permet à l'étudiant de voir comment son code progresse.

Le programme d'affichage ne doit afficher que les données envoyées, de sorte que le programme d'affichage de l'exercice 4 peut être utilisé tel quel. L'objectif principal sera d'étendre le programme afin d'inclure la vérification des réponses.

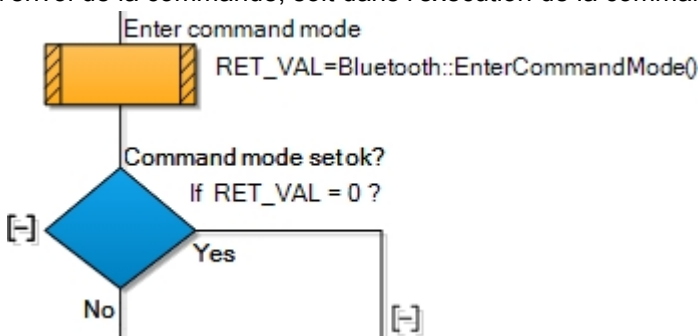
Utilisation de la macro WaitForStringValue

Cela permet d'interrompre le déroulement du programme jusqu'à la réception d'un message de réponse spécifique ou jusqu'à l'expiration de la période de temporisation. Par exemple, nous attendons ici le message "CONNECT" pendant un maximum de 5 secondes (5 000 mS).



Méthodologie de vérification des erreurs

Il convient d'adopter une méthodologie de base de vérification des erreurs, dans le cadre de laquelle les commandes sont vérifiées pour détecter les erreurs. Une erreur indique qu'il y a un problème soit dans la tentative d'envoi de la commande, soit dans l'exécution de la commande.



Par exemple, nous testons ici la valeur de retour pour vérifier que le mode de commande a été saisi avec succès.

Une fois qu'une commande a été envoyée, la fonction *WaitForStringValue* peut être utilisée pour vérifier que la réponse attendue a été reçue.

Introduction à Bluetooth Low Energy (BLE)

10.1 Modes de fonctionnement de Bluetooth Low Energy (BLE)

Le module Microchip RN4678 Bluetooth Dual Mode utilisé dans le cadre de cette formation prend en charge les communications Bluetooth Classic et Bluetooth Low Energy (BLE).

Dans le cadre de Bluetooth Classic, le RN4678 met en œuvre le profil de port série (SPP) standard qui prend en charge le transfert de données série entre deux appareils Bluetooth Classic.

Outre le SPP pour la connectivité Bluetooth Classic, le RN4678 dispose d'un service privé Generic Attribute Profile (GATT) pour le transfert de données en série entre deux dispositifs BLE.

Ce service de flux de données BLE fourni par le RN4678 est appelé "Transparent UART". Par conséquent, le RN4678 est un module Bluetooth bimode, qui prend en charge à la fois la connectivité Bluetooth Classic et la connectivité de données série BLE.

Le module fonctionne selon deux modes : le mode Données (par défaut) et le mode Commande.

Lorsque le module est "connecté" à un autre appareil et qu'il est en mode "données", il agit comme un tuyau de données.

En d'autres termes, tout ce qui est reçu de l'UART est transmis au dispositif homologué connecté via SPP s'il est connecté à un dispositif Bluetooth Classic, ou via un service GATT privé s'il est connecté à un dispositif BLE. Lorsque des données sont reçues du dispositif homologué par SPP pour Bluetooth Classic ou par UART Transparent pour BLE, ces données sont transmises directement à UART.

Le module Bluetooth est configuré et contrôlé en le mettant en mode commande et en exécutant des commandes de texte ASCII sur l'UART.

Il dispose de plusieurs modes de fonctionnement que l'utilisateur peut définir à l'aide de la commande SM :

[Mode par défaut \(SM,0\)](#) - dans lequel d'autres appareils Bluetooth peuvent découvrir le module et s'y connecter. Les connexions sortantes peuvent être initiées dans ce mode.

Mode de reconnexion automatique (SM,6) - Dans ce mode, le module tente de se connecter à l'appareil distant qui correspond à l'adresse distante enregistrée. Pour définir l'adresse distante, utilisez la commande SR.

Toutes les modifications de configuration apportées par les commandes Set restent dans la mémoire non volatile (NVM) et survivent au cycle d'alimentation. Les changements de configuration ne prennent effet qu'après un redémarrage.

Toutes les commandes Action prennent effet immédiatement mais n'ont pas d'effet après un cycle d'alimentation.

Exercice 6 : Service de données série UART transparent BLE GAP

Le RN4678 peut initier une connexion BLE en mode Generic Access Profile (GAP) Central avec un autre dispositif BLE prenant en charge le service "Transparent UART".

Objectifs

- Développez un programme qui se connecte à un deuxième appareil Bluetooth.
- Envoi de données sérielles au deuxième appareil
- Imprimer les messages et les données reçus sur l'écran LCD du nœud

Pré-requis

- Vous devez utiliser les programmes que vous avez développés précédemment comme point de départ.
- Une compréhension du processus d'enquête et de découverte
- Connaître l'adresse de l'appareil du module à connecter
- Pour les appareils qui sont uniquement BLE, utilisez la commande "F,5" pour connaître leur adresse.

Exigences en matière de matériel et de logiciel

Le matériel suivant est nécessaire :

- Les deux solutions Bluetooth sont nécessaires pour cet exercice.
- Solution 1, le nœud A sera connecté à

Solution 2, le nœud B connaîtra l'adresse du nœud A et établira une connexion avec lui.

Commandement	Description
SG,1	Régler le module en mode Bluetooth Low Energy 4.0
SA,2	Mode Secure Simple Pairing (SSP) "Just Works". Ce mode fonctionne sans qu'il soit nécessaire d'afficher ou de saisir des données sécurisées.
R,1	Redémarrez l'appareil pour appliquer la nouvelle configuration.
F,5	Interroger les appareils BLE dans le voisinage
C,<0,1>,<adresse MAC>	Tentative de connexion avec l'appareil distant, où le premier paramètre indique le type d'adresse qui peut être trouvé dans le résultat de l'enquête (0 pour une adresse publique et 1 pour une adresse privée). adresse)

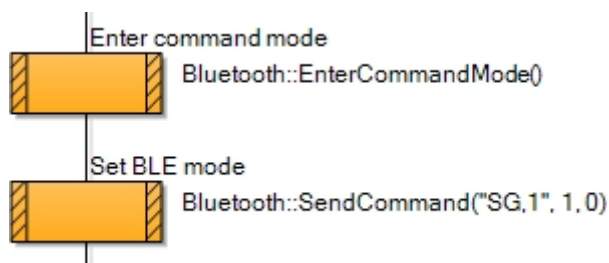
Résultats de l'apprentissage

Les principaux résultats d'apprentissage de cet exercice sont les suivants

- Configuration du module RN4678 pour qu'il fonctionne en mode BLE
- Configurer le module RN4678 pour qu'il fonctionne en mode "Just works" du SSP
- Connexion à un deuxième module Bluetooth BLE

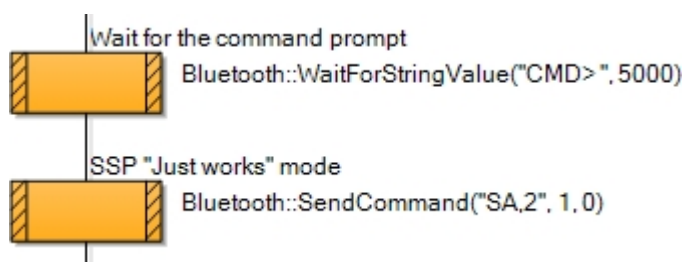
Mode BLE

Mettre le module en mode BLE avec la commande "SG,1".



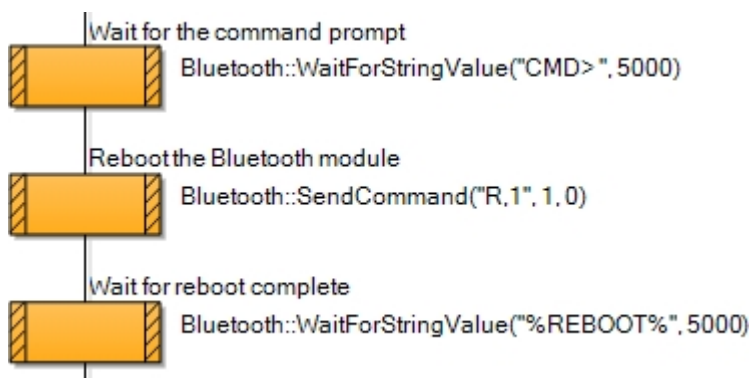
Appairage simple et sécurisé

Mettre le module en mode SSP "Just Works" avec la commande "SA,2".



Activation des paramètres

Redémarrer le module pour mettre en œuvre les paramètres avec la commande "R,1".



Pour le nœud A, affichage des réponses sur l'écran LCD comme pour les exercices précédents.

Pour le nœud B, envoyez la commande de connexion BLE et l'envoi de données série comme dans les exercices précédents.

Théorie : Confiance et sécurité

Les communications radio sont intrinsèquement vulnérables. Tout appareil capable de recevoir un signal radio dans les mêmes bandes de fréquences peut écouter le signal. Au fil des ans, les entreprises de radiocommunication ont déployé des efforts considérables pour sécuriser l'envoi des signaux. Et tout autant d'efforts ont parfois été consacrés à briser cette sécurité. Enigma et Bletchley Park sont peut-être les deux noms les plus célèbres de l'ère de la guerre des communications, mais leurs descendants continuent à vivre. Malheureusement, la menace la plus importante pour la sécurité ne provient pas des services de renseignement alliés, mais des pirates informatiques et des auteurs de virus, plus prosaïques mais insidieux, qui sévissent sur l'internet. Par ailleurs, si la résistance française ne surveille pas vos signaux, peut-être que vos rivaux commerciaux aimeraient le faire ? Les détails des contacts, des accords, des offres et des prix sont des munitions inestimables pour les membres du conseil d'administration, et l'humble téléphone, avec sa panoplie d'accessoires Bluetooth, est en première ligne.

Sécurité en général

Alors, comment Bluetooth s'y prend-il pour assurer votre sécurité ?

Trois systèmes complémentaires sont en place.

- Matériel
- Numéros d'identification personnelle (PIN)

Authentification, cryptage et confiance

Au niveau matériel, Bluetooth utilise des techniques de saut de fréquence qui passent d'une fréquence à l'autre. L'écoute d'une fréquence ne permet d'obtenir que des parties infimes du message. Pour écouter une conversation complète, il faut connaître les fréquences sur lesquelles sauter, une information qui n'est pas partagée par les appareils qui ne participent pas à la communication.

Les codes PIN sont souvent utilisés par des appareils tels que les téléphones mobiles, les écouteurs, etc. Le code PIN est fourni avec l'appareil et doit être saisi pour être vérifié par le programme.

Remarque : il ne s'agit pas de la même clé que celle qui est transmise par programme, mais d'un code PIN saisi par l'utilisateur. Les numéros PIN sont généralement utilisés pour informer l'appareil que vous êtes le propriétaire légitime afin d'empêcher les voleurs d'utiliser l'appareil. Pour des raisons de sécurité, les numéros PIN ne doivent pas être stockés avec l'appareil, tout comme vous ne laisseriez pas votre numéro PIN avec vos cartes bancaires.

Au niveau du programme et de l'appareil, les communications peuvent être authentifiées et l'appareil peut être enregistré comme appareil de confiance.

Modes de sécurité

Le module RN4678 prend en charge le cryptage et l'authentification avec des modes de sécurité de 1 à 4. La définition du mode de sécurité est la suivante :

- Mode de sécurité 1 : Confirmation de la clé de sécurité
- Mode de sécurité 2 : fonctionne simplement
- Mode de sécurité 3 : Saisie de la clé
- Mode de sécurité 4 : Code PIN hérité

Pour Bluetooth Classic, tous les modes de sécurité sont pris en charge. Pour BLE, seuls les modes de sécurité 1 à 3 sont pris en charge.

Pour les modes de sécurité 1 et 3, par défaut, un code de sécurité aléatoire à 6 chiffres est généré et affiché à une extrémité de la connexion et l'autre extrémité doit saisir le code de sécurité. En option, si les deux extrémités de la connexion sont RN4678, il est possible de fixer le code de sécurité à 6 chiffres pour BLE en fournissant un code à 6 chiffres à la commande SP au lieu d'un code à 4 chiffres pour l'ancien mode Code Pin.

Authentification

La commande Set Authentication du module RN4678 définit la méthode d'authentification lorsqu'un dispositif distant tente de se connecter. Une fois qu'un dispositif distant a échangé des codes pin avec le module RN4678, une clé de liaison est stockée en vue d'une utilisation ultérieure. L'appareil stocke automatiquement et en permanence jusqu'à quatre appareils homologues dans la mémoire flash à l'aide de la méthode FIFO (First-In, First- Out).

Un code pin de sécurité à 4 chiffres est utilisé pour l'appairage par code pin traditionnel, tandis que le code pin à 6 chiffres est utilisé pour l'authentification SSP dans BLE si un code pin fixe est souhaitable. Le code pin fixe à 6 chiffres dans BLE n'est pas pris en charge par de nombreuses implémentations Bluetooth Low En-ergy.

Confiance

Lorsque deux appareils Bluetooth ont initié le couplage et établi des communications entre eux, on dit qu'ils forment une paire de confiance. Une fois qu'une paire de confiance a été établie, ils peuvent communiquer entre eux sans avoir besoin de découverte ou d'authentification. Comme cela accélère manifestement les processus de connexion et de communication, il existe un certain nombre de commandes et d'options permettant de contrôler l'enregistrement des appareils auxquels vous faites confiance.

Les dispositifs de confiance sont mis en cache et stockés dans une liste. La liste des dispositifs de confiance est stockée dans une mémoire non volatile, ce qui permet de la conserver lors de la mise hors tension. Le cache, quant à lui, est temporaire et ne répertorie que le dernier dispositif à avoir été apparié et à avoir obtenu la confiance.

Exercice 7 : confiance et sécurité

Introduction

Une bonne communication est à la fois sûre et efficace. La sécurité est nécessaire pour bloquer les communications non autorisées. Toutefois, la sécurité ne doit pas être trop intrusive, et les dispositifs dont on sait qu'ils sont dignes de confiance doivent pouvoir communiquer facilement.

Objectifs

Créer un programme qui établit une connexion authentifiée avec un autre appareil

Bluetooth. Donnez à l'appareil trois options contrôlées par des commutateurs sur le port

A :

- Liste des dispositifs figurant actuellement dans la liste des dispositifs de confiance
- Ajouter l'appareil actuel à la liste des appareils de confiance
- Retirer le dispositif actuel de la liste des dispositifs de confiance.
- Utiliser le programme pour démontrer l'utilisation de la liste des dispositifs de confiance (Trusted Devices List).

Pré-requis

- Une compréhension du processus de connexion par paires tel qu'il est détaillé dans l'exercice 4.
- Une compréhension des réponses telles qu'elles sont détaillées dans l'exercice 5.

Exigences en matière de matériel et de logiciel

Le matériel suivant est nécessaire :

- Les deux solutions Bluetooth sont nécessaires pour cet exercice.
- Le nœud A sera utilisé comme dispositif à ajouter ou à supprimer de la liste des dispositifs de confiance.
- Le nœud B sera utilisé pour mettre en œuvre les objectifs.

Informations sur l'exercice

Commandement	Description
SP,<code à 6 chiffres>	Définir le code pin de sécurité
SA,1	Mode Secure Simple Pairing (SSP)
Y	Afficher les adresses des appareils dans la liste des appareils appariés avec l'index de 1 à 8
C,<1-8>,<MAC adresse>	Tentative de connexion avec l'appareil distant
K,1	Déconnexion
U,< 1-8>	Supprimer un appareil de la liste des appareils appariés
U,Z	Supprimer tous les appareils de la liste des appareils appariés

Pour plus de détails sur les commandes, veuillez consulter la référence des commandes dans le guide de l'utilisateur du module.

Résultats de l'apprentissage

Les principaux résultats d'apprentissage de cet exercice sont les suivants

Principes de sécurité, y compris :

- Authentification
- Dispositifs de confiance

Utilisation de la liste des dispositifs de confiance, y compris :

- Ajout de dispositifs
- Retrait des dispositifs
- Dispositifs d'inscription

Mise en œuvre pratique : Confiance et sécurité

Objectifs généraux

Le programme du nœud A servira de dispositif passif à distance à appairer et à connecter. Par conséquent, ce nœud est configuré en mode esclave BLE avec les commandes "SG,1" et "SM,0" et le code pin de sécurité est défini avec la commande "SP,123456". L'appareil doit ensuite être redémarré à l'aide de la commande "R,1" pour configurer ces paramètres.

Pour identifier ce nœud Bluetooth, nous utilisons la commande "GB" pour obtenir et afficher l'adresse du dispositif.

Le nœud B est le programme actif et est mis en mode maître BLE avec les commandes "SG,1" et "SM,1". L'appairage simple et sécurisé (SSP) est activé avec la commande "SA,1" et le code pin est défini avec la commande "SP,123456".

Le programme principal doit alors entrer dans une boucle continue qui saisit l'état des interrupteurs et répond à toute entrée, en envoyant la commande appropriée si nécessaire.

Pour cet exercice, utilisez le commutateur 0 comme "Liste des dispositifs de confiance", le commutateur 1 comme "Ajouter un dispositif et se connecter", le commutateur 2 comme "Déconnecter" et le commutateur 3 comme "Supprimer un dispositif".

Les commandes à envoyer sont les suivantes :

Interrupteur	Commandement	Description
Interrupteur 0	Y	Obtenir et afficher la liste des appareils de confiance
Interrupteur 1	C,<1-8>,<adresse MAC>	Appairer et se connecter avec un appareil, ajouter à la liste
Interrupteur 2	K,1	Déconnexion de l'appareil distant
Interrupteur 3	N,<1-8>	Supprimer l'adresse de l'appareil distant de la liste

Un moyen simple de déterminer si la fonction Ajouter et supprimer un appareil de la liste des appareils de confiance a fonctionné consiste à dresser la liste des appareils et à rechercher l'adresse de l'appareil.

L'écran LCD peut être utilisé pour afficher non seulement la liste des dispositifs de confiance, mais aussi les messages de réponse pour les autres options.

Caractéristiques supplémentaires

En option, le programme de base peut être étendu pour ajouter des dispositifs supplémentaires et pour effacer toute la liste des dispositifs de confiance. Les détails de ces commandes peuvent être trouvés dans la section "RN4678 Bluetooth® 4.0 Dual Mode Module User's Guide".

Principes de conception des projets

Une fois que les étudiants se sont familiarisés avec Bluetooth et ont réalisé les exercices de ce cours, l'étape suivante consiste à utiliser ces connaissances pour concevoir et mettre en œuvre un projet.

Les considérations relatives à la conception sont abordées ci-dessous, mais aucun code ou instruction n'est donné. Elles peuvent servir de point de départ à l'élaboration des objectifs et des spécifications du projet.

Projet - Télécommande simple

Une utilisation simple du transfert de données pourrait consister à contrôler un appareil sans fil. Cela peut être démontré par l'utilisation d'une carte de commutateurs attachée à un processeur et d'un module Bluetooth configuré comme maître pour transmettre les données de commutation à un second processeur avec un module Bluetooth configuré comme esclave qui reçoit les données et les répète sur une carte LED attachée.

Extension du projet

Réfléchissez à la manière dont le nombre de transferts de données pourrait être réduit au minimum.

Projet - Enregistreur de données médicales

Une tâche utile pour Bluetooth serait l'enregistrement de données médicales. Les patients ont besoin d'être surveillés, mais les fils et les prises attachent le patient à son lit et créent un surcroît de travail pour le personnel. Si un patient doit passer des examens, ou même aller aux toilettes, l'équipement doit être débranché et déplacé, ou bien il doit être mobile. Et qu'y a-t-il de plus mobile que le Bluetooth ? Pour les patients, la possibilité de se déplacer leur donne plus de liberté et d'indépendance. Pour le personnel, le système permet d'enregistrer des données 24 heures sur 24 et de faciliter les déplacements des patients. Il est également possible d'utiliser l'alerte sur les données, ce qui permet au personnel d'être alerté des problèmes dès qu'ils surviennent.

Ces enregistreurs de données pourraient également être utilisés dans le cadre de tests médicaux, par exemple dans le domaine du sport, où les capteurs câblés traditionnels ne feraient qu'entraver le bon déroulement des opérations.

La première chose à considérer pour l'enregistreur de données à distance est de savoir quelles données vous souhaitez enregistrer.

Un simple enregistreur de données permet de surveiller la température. Il est également possible d'ajouter d'autres capteurs, tels que des moniteurs de fréquence cardiaque et de respiration.

Vous devez ensuite déterminer la fréquence à laquelle les données doivent être mises à jour. Pour une donnée telle que la température, il peut s'agir d'un relevé toutes les 5 ou 10 minutes. Pour des données plus critiques, telles que la surveillance de la fréquence cardiaque, vous pouvez avoir besoin de relevés toutes les quelques secondes. Après avoir déterminé la fréquence d'envoi des données, vous devez décider d'une stratégie pour l'envoi des données.

Maintenir un contact permanent ? Contacter et envoyer les données ? Collecter des données pendant un certain temps, puis les envoyer par lots ?

Ensuite, y a-t-il des cas particuliers à prendre en compte ? Une augmentation soudaine du rythme cardiaque est-elle importante ? Qu'en est-il de la température ? Faut-il nous avertir par des feux clignotants et des sonneries d'alarme ? Faut-il déclencher nos bips pour que nous sachions qu'il y a un problème ? Ou bien une simple alerte LED et un message sur un écran LCD nous suffisent-ils ? Avons-nous besoin de savoir tout de suite ou dans 5 minutes, lors de la prochaine diffusion de données ?

Extension du projet

Les moniteurs médicaux posent deux problèmes évidents par rapport aux moniteurs fixes qui limitent les mouvements.

- Que se passe-t-il lorsque le patient sort de la portée du réseau ? Où se trouve le patient lorsque l'alarme se déclenche ?

Comment ces problèmes peuvent-ils être résolus ?

Certaines de ces questions sont trop importantes pour être abordées dans le cadre d'un projet, mais l'étudiant pourrait montrer comment il les a prises en compte s'il les rédigeait dans une section "Caractéristiques étendues/complémentaires" de son rapport de projet.

