# Liste des codes

Mon idée est de réaliser dans un premier temps des codes très simples (qq lignes) pour tester facilement le fonctionnement de chaque composant.

Ensuite dans un deuxième temps je voudrais réaliser des codes plus complexes qui reflètent ce qui sera abordé dans le cadre du cours de systèmes embarqués, ainsi ces codes pourront servir de support dans la présentation du cours. Ils permettront d’illustrer les objectifs de chaque tp.

1. Codes simples
   * Bouton tactile

Utilisation de la fonction touchRead(pin) qui renvoie une valeur en fonction de si on touche ou pas.

A vérifier la valeur qui est renvoyée quand on touche pour calibrer le bouton tactile.

à Comme toutes les entrées de l’esp peuvent fonctionner comme interruptions on peut utiliser le bp tactile comme interrupt externe. Exemple utilisation des interruptions.

* + Bouton simple

Le test du bouton passe par un simple digital read ?? ==> je vais l’intégrer dans le code avec le bp tactile ainsi on pourra tester tous les boutons directement.

* + Rgb + pot

Un petit code qui allume la led rgb et fait varier la couleur à partir de l’entrée analogique du potentiomètre qui détermine la vitesse de variation.

* + Dht + lcd oled + ~~thermocouple via max6675~~

On peut réaliser une lecture humidité et temp via dht et l’afficher sur le lcd // + utilisation du spi pour le thermocouple

Max6675 séparé

* + Can è juste exemple mais je dois épurer

Pour tester le CAN je pensais partir de ce qu’il me reste du projet intégré à mcp 2515 pour communiquer en CAN avec la carte è pour le test : l’envoi et la réception d’un compteur fera office de démo (mais pour un code plus construit j’ai quelques idées qui seront abordées plus loin)

* + Nrf è +/- fini simple test comme exemple rf24

Mais pt aller plus loin

* + I2c
    1. Lcd
    2. Mcp23017

Je pense réaliser un code de démo qui intègre tous les participants du bus i2c de manière assez simple

* On peut ainsi tester les composants cependant ces codes font office de diagnostic pour tester le fonctionnement des composants donc important de différencier les différents fonctionnements et les isoler
  + Je vais intégrer différents « print » pour pouvoir tracker les actions et l’exécution du code

Taper un chiffre sur le clavier et l’afficher sur le 7seg è et utiliser le lcd i2c pour afficher des infos ex : (si le nb est plus grand qu’un certain seuil affiche « seul atteint »)

* Le lcd OLED qui fonctionne aussi en i2c n’est pas exploité dans cet exemple pour éviter de le rendre inutilement lourd.

Pour détailler le fonctionnement --> je pense taper 4 chiffres sur le clavier et les remettre sur le 7 segments tout simplement. Une boucle qui attend 4 appuis puis affiche puis attend 4 appuis

* + Ldr

Faire la lecture analogique de la photo résistance pour tester son fonctionnement

Ajouter ça a code simple lecture analogique et ensuite traitement. Exemple allumer une led si lumi trop faible

Mosfet + boutons

Faire commuter le relais par un appui sur un BP sans mettre de charge juste pour entendre le bruit et s’assurer du fonctionnement. Pas de démo simple digitalwrite

On peut ajouter ce code à celui de la ldr on remplace le bouton par ldr

Simple code de démo

* + Encodeur rotatif

1. Codes avancés

Comme évoqué, dans cette section je vais produire des codes de démonstration directement en relation avec le cours de systèmes embarqués2. L’objectif étant qu’ils puissent servir d’exemple voir de support dans la présentation du cours.

* Idées :
  + Multi fichiers è je pense qu’il est plus qu’int de travailler avec différents fichier C intégrés dans un même code et ce pour deux raisons : la première étant l’aspect didactique puisque pour illustrer la méthodologie à adopter quand on travaille avec plusieurs fichiers. La deuxième, qui est selon moi la « vraie » raison, c’est que je vais être amené à utiliser bcp de composants et il est bcp plus facile d’éditer un code quand celui-ci ne fait que qq ligne avec des appels précis que sur un code de 1000 lignes.
  + Je pensais réaliser un code dans la même optique que celui réalisé pour mon examen : où j’intègre des fonctions d’IoT les unes avec les autres è il est selon moi plus visuel de comprendre le fonctionnement de nodered quand on analyse une application complète.
    - Créer une dashboard via n’importe quel outil è et lier cette dashboard à un broker configuré manuellement (déployé depuis un raspberry par exemple àsi j’ai letemps ou sinon depuis un ordi)
    - Ensuite explorer d’autre outils utilisés dans le monde de l’IoT et éventuellement les intégrer
  + Le cœur du code sera de contrôler toutes les fonctions de la carte
  + …

Idéalement, j’aimerais le faire en trouvant un scénario réaliste è je trouve qu’il est souvent plus stimulant d’apprendre quand on peut facilement voir l’utilité et l’utilisation directe de ce qu’on apprend.

En plus cela donne un aspect plus concret au travail, pour ne pas avoir cet aspect purement gadget.

L’aspect modulaire peut être ajouté au deuxième concept. En créant des fichier C annexes qui contiendront toutes les fonctions nécessaires.

Les fonctions imposées :

* Connexion à internet pour être commandé à distance WiFi
* Mise en réseau de plusieurs cartes via nrf
* Utilisation du bus I2c et SPI
* … la liste sera complètée détaillée avec d’autres idées.

Pour réaliser ça je pensais à une carte dédiée au déploiement d’un site qui conteindra toute la documentation pour comprendre l’installation. + des infos sur la procédure pour interfacer d’autres cartes ect.

A quoi on ajoute la communication à distance via mqtt + MAJ d’une base de données.

Cette carte fait office de master.

Une seconde (troisième ,…) communique via le module rf avec le master celle-ci, peut être imaginée comme une carte d’acquisition placée pour gérer et visualiser les données d’un porcess. On peut penser à la gestion d’une salle de réception de manière automatique (lumière, température, humidité ,ventilation ,…)

--> pour l’utilisation du clavier j’avais pensé à : au démarrage pour enclencher/éteindre le système demander un code

--> lcd i2c display données capteurs

--> on peut afficher l’heure sur le 7 segments

...