



Systèmes Embarqués 1 & 2
Objectifs du travail écrit n° 1

Les étudiant-e-s devront être capable :

Programmation en C

- de décrire la structure et l'organisation des fichiers en C
- de décrire sommairement le processus de développement (édition, compilation, édition de liens) propre à C
- de manipuler correctement les types de base de C
- de coder un algorithme de traitement de données en C
- de concevoir une interface d'une bibliothèque C dans les règles de l'art (header-file)
- de coder dans les règles de l'art des appels de fonctions en C
- de décrire le passage d'arguments par valeur et par référence lors d'appel de fonctions en C
- de différencier les fonctions globales des fonctions locales dans une application codée en C
- de différencier les variables globales des variables locales et des variables rémanentes dans une application codée en C
- de manipuler correctement les types complexes (énumérations, tableaux, structures, unions, ...) de C
- de concevoir une interface C permettant d'accéder aux registres d'un périphérique
- de manipuler correctement les pointeurs en C
- d'allouer et restituer correctement des objets dynamiques en C
- de décrire et utiliser les conversions des types en C
- de manipuler correctement les pointeurs de fonction en C
- d'expliquer les conditions pour qu'une fonction C soit réentrante



Systèmes Embarqués 1 & 2
Objectifs du travail écrit n° 1

Chaîne d'outils

- de décrire l'organisation des fichiers sources
- de décrire la gestion de versions et révisions ainsi que des outils utilisés
- de décrire les méthodes de documentation d'applications en C et des outils utilisés
- de décrire les méthodes de vérification et de tests en C
- de concevoir un programme de tests unitaires en C
- de décrire sommairement la fonction des différents utilitaires utilisés pour le développement de logiciel en C
- de concevoir un Makefile pour la génération d'applications codées en langage C (plusieurs cibles, règles suffixes, dépendances, conditions, sous-make,...)

Travail pratique TP.01

- de décrire les différentes étapes et opérations pour mettre en place un dépôt Git
- de décrire les opérations pour ajouter un nouveau fichier dans un dépôt Git
- de décrire les opérations pour synchroniser les dépôts locaux avec le dépôt centralisé
- d'expliquer comment traiter les conflits de la synchronisation des dépôts Git
- d'expliquer comment trouver la taille et l'emplacement en mémoire d'une variable
- d'expliquer comment trouver la taille d'une application stockée dans la mémoire du µP
- d'expliquer le fonctionnement de la macro « ARRAY_SIZE(x) »

Travail pratique TP.02

- de citer les composants mis en œuvre sur le µP AM3358 de TI pour accéder aux portes d'entrée/sortie
- de décrire le rôle du multiplexeur PAD-Mux
- de décrire le principe de fonctionnement du contrôleur d'entrée/sortie GPIO
- de décrire et mettre en œuvre les services de la bibliothèque « am335x_gpio.h »
- de décrire le principe de fonctionnement de l'afficheur 7-segments
- de décrire le principe de fonctionnement d'un encodeur rotatif
- de décrire et concevoir la machine d'états logicielle de l'encodeur rotatif