Haute école d'ingénierie et d'architecture Fribourg
Hochschule für Technik und Architektur Freiburg

Systèmes Embarqués 1 & 2

tp.10b - Mini-Projet 2 - OS

Classes T-2/I-2 // 2016-2017

Daniel Gachet | HEIA-FR/TIC

tp.10b | 23.04.2017

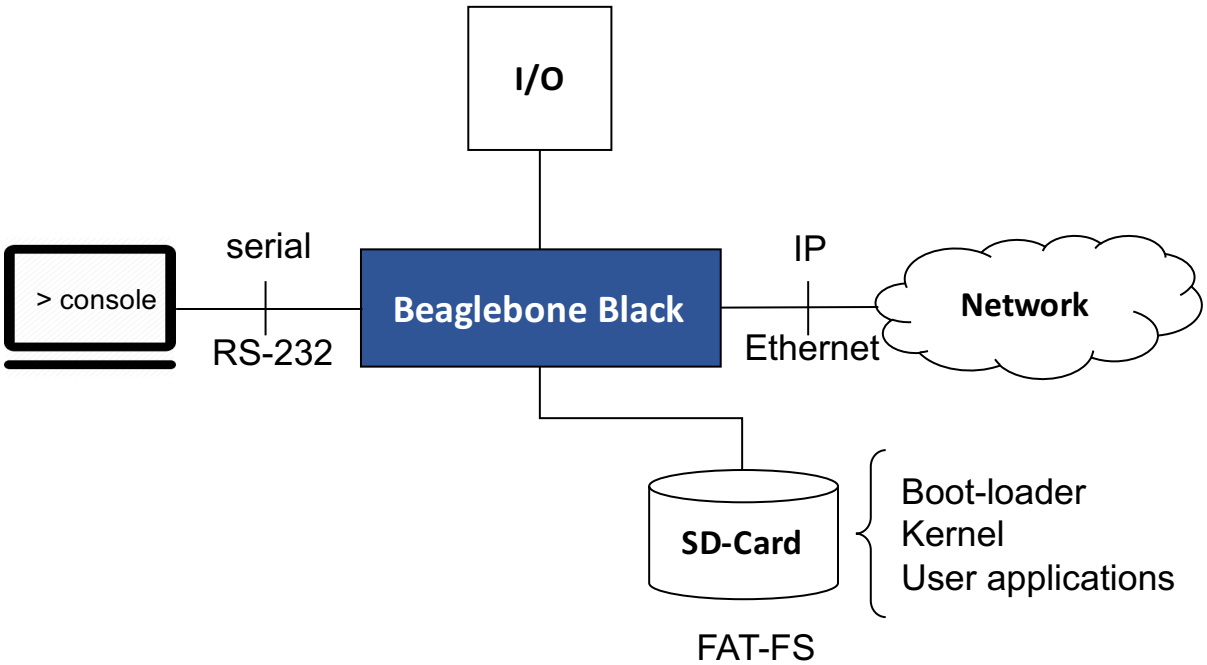
Hes·so
Haute Ecole Spécialisée
de Suisse occidentale
Fachhochschule Westschweiz

Objectifs

- A la fin du laboratoire, les étudiant-e-s seront capables de
  - Synthétiser les notions acquises durant les cours
  - Gérer de bout en bout un mini-projet informatique en groupe
  - Analyser et clarifier l'énoncé d'un projet
  - Elaborer et comparer diverses variantes de réalisation
  - Concevoir des pilotes de périphériques simples et étudier leur datasheet
  - Concevoir et développer une application modulaire en C sur un système embarqué mettant en oeuvre un système d'exploitation élémentaire
- Durée
  - Les 5 dernières séances de laboratoire
- Rapport
  - Le journal de laboratoire et le code source devront être rendus le vendredi de la semaine P15 à 23h59

Travail à réaliser - Infrastructure

- Le Beaglebone Black, avec sa carte SD, son interface réseau, sa ligne sérielle et ses divers périphérique, dispose d'une infrastructure suffisante pour la réalisation d'un système d'exploitation

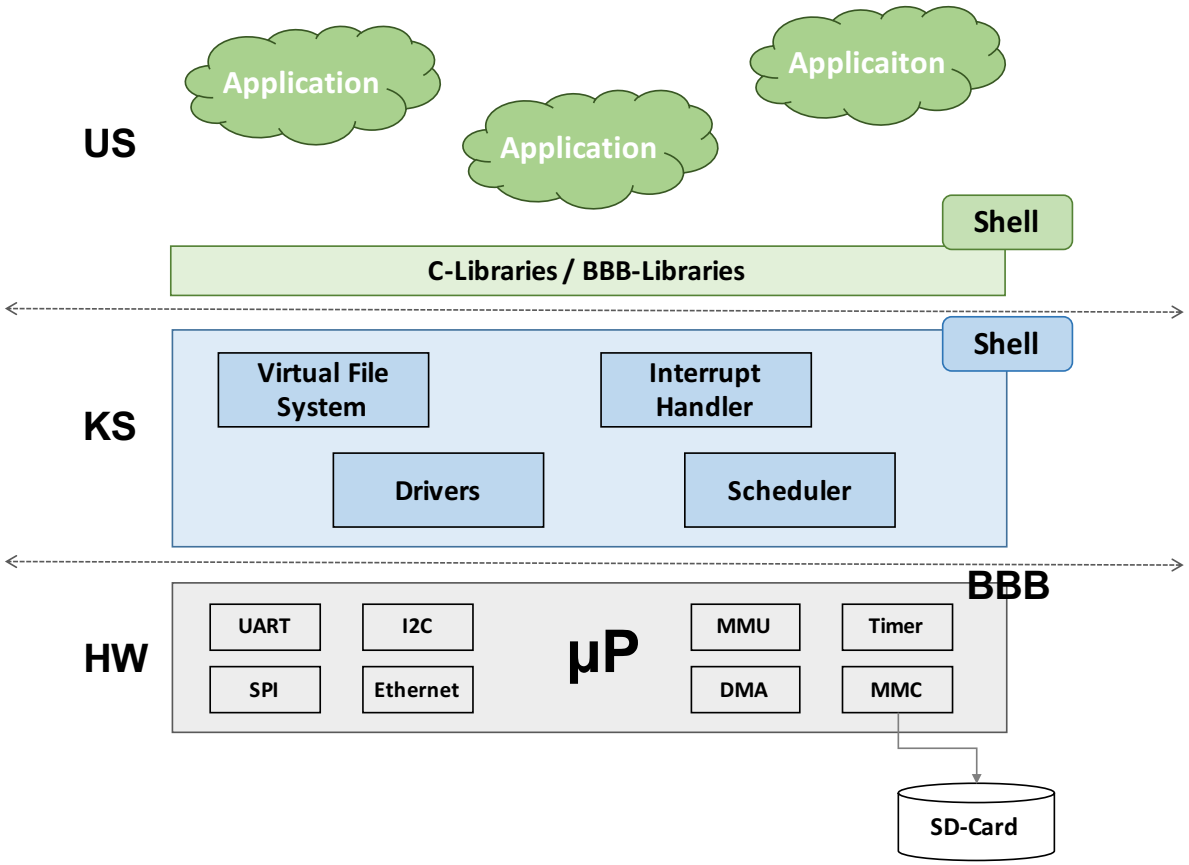


Travail à réaliser - Application

- Migration de l'application "thermo-buzzer" en un processus en espace utilisateur
- Pour son bon fonctionnement, ce processus utilisera les données fournies par des pilotes en espace noyau, soit
  - Pilote du contrôleur ePWM1
  - Pilote du contrôleur GPIO
  - Pilote du thermomètre I2C
  - Pilote du display 7-segment
  - Pilote de l'encodeur rotatif

Travail à réaliser - Architecture SW

- Le système d'exploitation sera réalisé en 2 couches bien distinctes, l'espace noyau (kernel space) et l'espace utilisateur (user space), lesquelles seront reliées par un système de fichiers virtuels



Travail à réaliser - Espace Noyau

- L'espace noyau offrira la fonctionnalité suivante
  - Démarrage du noyau depuis la SD-Card lors du boot de la carte BBB
  - Gestion via un système de fichiers virtuels
  - Gestion de la MMU / Cache
  - Gestion de thread dans l'espace noyau
  - Gestion de processus dans l'espace utilisateur
  - Lancement d'applications stockées sur la SD-Card
  - Accès aux pilotes au travers du système de fichiers virtuels
  -

Travail à réaliser - Système de fichiers virtuels

- Le système de fichiers virtuels proposera l'arborescence suivante
  - /sdcard : accès aux fichiers stockés sur la SD-Card
  - /dev : accès aux données des différents périphériques
  - /sys : accès aux données des pilotes de périphériques
  - /proc : accès aux données du noyau (option)
- Les services sur les fichiers offerts par la bibliothèque standard C aux informations aux applications en espace utilisateur et noyau d'accéder aux données du système de fichiers virtuels

Travail à réaliser - Shell

- Le système devra lancer une console (shell) offrant les services/commandes suivant(e)s
  - cat : visualisation du contenu d'un fichier
  - cd : changement de répertoire
  - echo : affichage d'un string avec redirection dans un fichier
  - fork : lancement d'une nouvelle application stockée sur la SD-Card
  - help : affichage d'un texte d'aide pour une commande
  - ls : liste des fichiers contenus dans un répertoire
  - mkdir : création d'un répertoire
  - ps : liste des tâches (threads et processus)
  - pwd : affichage du répertoire courant
  - rm : effacement d'un fichier et/ou d'un répertoire
- La console devra de préférence être réalisée en espace utilisateur
- Dans l'idéal, la console offrira un historique des commandes avec édition de la ligne de commande

Mise à jour et conditions

- Pour télécharger le squelette du projet du dépôt centralisé

```
$ cd ~/workspace/se12/tp
$ git pull upstream master
```

- Pour mettre à jour les paths des includes dans eclipse
  - ouvrir `Properties` pour votre projet
  - aller `C/C++ General` → `Paths and Symbols`
  - ouvrir `Includes` → `GNU C`
  - ajouter `~/workspace/se12/bbb/source`
- Le code et le rapport seront rendus au travers du dépôt Git centralisé
  - *sources* : *.../tp/tp.10b*
  - *rapport* : *.../tp/tp.10b/doc/report.pdf*