

## Construction de logiciel en environnement embarqué

**Groupe 10**

Mercredi, de 17h30 à 20h30 SH-3120 (cours)

Mercredi, de 13h30 à 16h30 PK-4765 (atelier)

---

### Responsable(s) du cours

---

**Nom du coordonnateur :** MIGLIORINI, Cédric**Nom de l'enseignant :** MIGLIORINI, Cédric**Local :** PK-4150**Téléphone :** (514) 987-3000 #3699**Courriel :** migliorini.cedric@uqam.ca

---

### Description du cours

---

Ce cours vise à permettre aux étudiants d'acquérir les connaissances et les compétences qui permettent de faire le développement et l'intégration d'applications dans des environnements embarqués.

Processeurs (entrée/sortie, mode superviseur et déroulement); plateformes de traitement embarquées (bus processeur, mémoire, périphériques d'entrée/sortie, interfaces); structures et services des systèmes d'exploitation; construction de programme ( patrons de conception, test); compilation; processus et multi-tâches, changement de contexte; politiques d'ordonnancement temps-réel; services d'un système d'exploitation temps-réel; mécanismes de communication entre processus; réseaux pour systèmes embarqués; études de cas dans un environnement de développement.

Préalables académiques :

Cours de 3 heures et un laboratoire de 3 heures/semaine.

---

### Objectifs du cours

---

Ce cours vise à familiariser les étudiants aux différents aspects de la programmation pour l'embarqué. Il vise notamment à comprendre le fort couplage existant entre le logiciel et la plateforme matérielle sur laquelle le logiciel s'exécute afin de produire un code efficace.

Spécifiquement, l'étudiant qui complète le cours avec succès sera capable :

- De comprendre pourquoi les systèmes embarqués sont si populaires;
- De connaître les compétences recherchées chez un développeur de logiciel embarqué;
- De coder efficacement pour mettre en oeuvre les périphériques d'une plateforme matérielle;
- De connaître les caractéristiques principales d'un système d'exploitation temps réel;
- De coder une application embarqué à l'aide d'un système d'exploitation temps réel;
- D'utiliser certains périphériques plus évolués pour augmenter la fonctionnalité d'un système embarqué;
- De connaître quelques techniques d'optimisation du code pour une plus grande performance.

---

### Contenu du cours

---

Voici un aperçu des thèmes que seront traités dans ce cours.

Chapitre 1 : Introduction

- Présentation du plan de cours
- Définition, enjeux, domaine d'application exemples des systèmes embarqués; compétences d'un développeur de logiciel embarqué
- Les langages dans l'embarqué (C/C++, assembleur, etc.)

Chapitre 2 : Architecture d'un système embarqué et environnement de développement

- Connaître sa plateforme matérielle
- Construire son programme (compiler, éditer les liens, localiser)
- Télécharger et déboguer son programme
- Interruptions
- Mémoires et périphériques

Chapitre 3 : Système d'exploitation temps réel

- Systèmes d'exploitation (tâches, ordonnanceur, synchronisation des tâches, gestion de interruptions, etc.)
- L'exemple de  $\mu$ C/OSIII
- L'exemple de FreeRTOS
- L'exemple de Linux embarqué

Chapitre 4 : Périphériques plus évolués

- Augmenter la fonctionnalité (contrôleurs USB, Ethernet et de mémoire externe)
- Protocole USB

Chapitre 5 : Optimisation d'un système embarqué

- Techniques d'optimisation pour un code plus rapide, réduire la consommation de puissance et/ou sauver de l'espace mémoire
- Options d'optimisation des compilateurs

## Modalités d'évaluation

---

Description	Date	Pondération
Présentation d'un RTOS	-	10 %
Travaux pratiques	Spécifiée dans les énoncés	45 %
Examen final	Fin du trimestre	45 %

### Présentation d'un RTOS

Chaque étudiant devra effectuer une étude détaillée d'un système d'exploitation temps réel (RTOS) utilisé dans l'embarqué. Le choix du RTOS est à la discrétion du professeur (suggestions possibles des étudiants). Cette étude conduira à une présentation en classe. La présentation sera notée en partie par les pairs.

## Travaux pratiques

La mise en pratique des concepts vus en classe se fera par la réalisation de travaux pratiques en laboratoire, faisant appel à différents environnements de développement et chaînes d'outils. Ces travaux seront réalisés par équipe de deux étudiants.

## Remise des rapports

Les rapports doivent être rendus électroniquement par l'intermédiaire du site Moodle du cours. Les travaux remis en retard seront pénalisés, à raison de 20 % de la note globale par jour (incluant samedi, dimanche et congés) de retard.

Exceptionnellement (par ex., panne de Moodle), une copie pourra être rendue par courriel régulier.

Chaque fichier doit être nommé de façon à ce qu'on puisse identifier les membres de l'équipe (par exemple, par l'utilisation d'initiales), de même que le titre de la manipulation / simulation. Si plusieurs versions d'un même rapport sont remises, un numéro de version significatif doit être inclus dans le nom. Attention : des erreurs de titres pourraient faire que des copies ne soient pas corrigées, ou que des résultats soient confondus par mégarde.

Le format de fichier pour les documents doit absolument être PDF (Portable Document Format), ce qui assure que ce qui est rendu est conforme à la version de l'étudiant et ne risque pas d'être modifié par la suite. Un rapport doit normalement être présenté en un seul fichier, avec annexes, le cas échéant, pour les codes sources, etc. Un guide détaillé disponible sur le site du cours, donne davantage d'informations sur la présentation des rapports de laboratoire.

Les règlements concernant le plagiat seront strictement appliqués. Pour plus de renseignements, veuillez consulter le site suivant :

<http://www.sciences.uqam.ca/etudiants/integriteacademique.html>

## Examen

Une moyenne d'au moins 50 % à l'examen est exigée pour réussir le cours. L'utilisation de documentation personnelle (notes de cours, manuels) à l'examen sera limitée à quelques pages de notes personnelles.

### Politique d'absence aux examens

**L'autorisation de reprendre un examen en cas d'absence est de caractère exceptionnel. Pour obtenir un tel privilège, l'étudiant-e doit avoir des motifs sérieux et bien justifiés.**

Il est de la responsabilité de l'étudiant-e de ne pas s'inscrire à des cours qui sont en conflit d'horaire, tant en ce qui concerne les séances de cours ou d'exercices que les examens. **De tels conflits d'horaire ne constituent pas un motif justifiant une demande d'examen de reprise.**

Dans le cas d'une absence pour raison médicale, l'étudiant-e doit joindre un certificat médical original et signé par le médecin décrivant la raison de l'absence à l'examen. Les dates d'invalidité doivent être clairement indiquées sur le certificat. Une vérification de la validité du certificat pourrait être faite. Dans le cas d'une absence pour une raison non médicale, l'étudiant-e doit fournir les documents originaux expliquant et justifiant l'absence à l'examen &ndash; par exemple, lettre de la Cour en cas de participation à un jury, copie du certificat de décès en cas de décès d'un proche, etc. Toute demande incomplète sera refusée. Si la direction du programme d'études de l'étudiant-e constate qu'un étudiant a un comportement récurrent d'absence aux examens, l'étudiant-e peut se voir refuser une reprise d'examen.

L'étudiant-e absent-e lors d'un examen doit, dans les cinq (5) jours ouvrables suivant la date de l'examen, présenter une demande de reprise en utilisant le formulaire prévu, disponible sur le site Web du département à l'adresse suivante : <http://info.uqam.ca/politiques/>

L'étudiant-e doit déposer le formulaire dûment complété au secrétariat de la direction de son programme d'études : PK-3150 pour les programmes de premier cycle, PK-4150 pour les programmes de cycles supérieurs. Pour plus de détails sur la politique d'absence aux examens du Département d'informatique, consultez le site web suivant : <http://info.uqam.ca/politiques>

## Intégrité académique

### PLAGIAT Règlement no 18 sur les infractions de nature académique. (extraits)

**Tout acte de plagiat, fraude, copiage, tricherie ou falsification de document commis par une étudiante, un étudiant, de même que toute participation à ces actes ou tentative de les commettre, à l'occasion d'un examen ou d'un travail faisant l'objet d'une évaluation ou dans toute autre circonstance, constituent une infraction au sens de ce règlement.**

La liste non limitative des infractions est définie comme suit :

- la substitution de personnes;
- l'utilisation totale ou partielle du texte d'autrui en la faisant passer pour sien ou sans indication de référence;
- la transmission d'un travail pour fins d'évaluation alors qu'il constitue essentiellement un travail qui a déjà été transmis pour fins d'évaluation académique à l'Université ou dans une autre institution d'enseignement, sauf avec l'accord préalable de l'enseignante, l'enseignant;
- l'obtention par vol, manoeuvre ou corruption de questions ou de réponses d'examen ou de tout autre document ou matériel non autorisés, ou encore d'une évaluation non méritée;
- la possession ou l'utilisation, avant ou pendant un examen, de tout document non autorisé;
- l'utilisation pendant un examen de la copie d'examen d'une autre personne;
- l'obtention de toute aide non autorisée, qu'elle soit collective ou individuelle;
- la falsification d'un document, notamment d'un document transmis par l'Université ou d'un document de l'Université transmis ou non à une tierce personne, quelles que soient les circonstances;
- la falsification de données de recherche dans un travail, notamment une thèse, un mémoire, un mémoire-crédation, un rapport de stage ou un rapport de recherche;
- Les sanctions reliées à ces infractions sont précisées à l'article 3 du Règlement no 18.

Les règlements concernant le plagiat seront strictement appliqués. Pour plus de renseignements, veuillez consulter les sites suivants : <http://www.sciences.uqam.ca/etudiants/integrite-academique.html> et <http://www.bibliotheques.uqam.ca/recherche/plagiat/index.html>

## Médiagraphie

?<sup>VR</sup> BARR, M. & MASSA, A. - Programming Embedded Systems with C and GNU Development Tools - 2nd edition. 2006.

<sup>VR</sup> WHITE, E. - Making Embedded Systems - 2011

<sup>VR</sup> LEWIS, D. W. - Fundamentals of Embedded Software with the ARM Cortex-M3 - 2013

<sup>VR</sup> LABROSSE, J. J. - µC/OS-III, the Real-Time Kernel - 2009

<sup>SR</sup> Compaq, HP, Intel, Lucent, Microsoft, NEC, Philips - Universal Serial Bus Specification, Revision 2.0 - 2000

<sup>VR</sup> LABROSSE, J. J. [ET AL.] - Embedded Software: Know It All - 2008

<sup>VR</sup> PECKOL, J.K. - Embedded Systems: A Contemporary Design Tool - 1st Edition. 2008

<sup>VC</sup> GANSSELE, J., BARR, M. - Embedded Systems Dictionary - 2003

<sup>VC</sup> LIPIANSKY, E. - Embedded Systems Hardware for Software Engineers - 2011

<sup>VC</sup> GANSSELE, J. - The Art of Designing Embedded Systems - 2008

<sup>VC</sup> SMITH, W.A. - C Programming for Embedded Microcontrollers - 2008

<sup>VC</sup> HAMACHER, C., VRANESIC, Z., ZAKY, S., MANJIKIAN, N. - Computer Organization and Embedded Systems - 2012

### Autres lectures

D'autres documents seront soumis pour lecture durant la session. La liste sera tenue à jour sur le site web du cours.

A : article - C : comptes rendus - L : logiciel  
S: Standard - U : uri - V : volume

C : complémentaire - O : Obligatoire - R : recommandé