



Introduction aux systèmes linux embarqués et à la programmation web en DUT GEII : applications en domotique et en robotique

Serge Bes, Patrick Gumuchian

► To cite this version:

Serge Bes, Patrick Gumuchian. Introduction aux systèmes linux embarqués et à la programmation web en DUT GEII : applications en domotique et en robotique. Colloque sur les Objets et systèmes Connectés, Ecole Supérieure de Technologie de Casablanca (Maroc), Institut Universitaire de Technologie d'Aix-Marseille (France), Jun 2019, CASABLANCA, Maroc. <hal-02298767>

HAL Id: hal-02298767

<https://hal.science/hal-02298767v1>

Submitted on 27 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Introduction aux systèmes linux embarqués et à la programmation web en DUT GEII : applications en domotique et en robotique

Serge BES (serge.bes@univ-amu.fr), Patrick GUMUCHIAN (patrick.gumuchian@univ-amu.fr)
IUT AMU - Département Génie Electrique et Informatique Industrielle de Marseille
142, Traverse Charles Susini – BP157 - 13388 Marseille cedex 13

RESUME : L'objectif est d'illustrer les réseaux informatiques, les systèmes embarqués et les bus qu'ils utilisent, tout en faisant découvrir la programmation web à des étudiants préparant le DUT GEII. Le module d'enseignement présenté conduit à la réalisation d'un système, basé sur un nano ordinateur linux (Raspberry Pi), qui héberge un serveur web Apache2 et pilote ses périphériques en exécutant des programmes multitâches écrits en langage C. Ce travail pédagogique est inscrit dans un parcours « Informatique pour système embarqué » au sein de la deuxième année du DUT GEII.

Mots clés : USB, SPI, I2C, Bluetooth, Internet des objets, programmation Web, CGI, langage C, Raspberry Pi, linux, analyseur de protocoles, http, css, php, javascript, système embarqué, analyseur de spectre, antenne doublet, bande ISF, modulation FSK, convertisseur BUCK, modulation PWM.

1 PRESENTATION DU CONTEXTE

1.1 Le DUT GEII

La formation conduisant au Diplôme Universitaire de Technologie (DUT) en Génie Électrique et Informatique Industrielle (GEII) correspond aux deux premières années d'enseignement supérieur. Elle est organisée sur 4 semestres et découpée en modules. Elle comporte un tronc commun et des modules complémentaires choisis par les étudiants, en fonction de leur projet professionnel. Ces modules complémentaires sont dispensés sur la fin de la deuxième année. Cet article concerne le travail réalisé auprès d'étudiants qui choisissent de s'orienter vers l'informatique industrielle et les réseaux, dans un parcours « informatique pour système embarqué ».

1.2 Les modules « d'informatique industrielle » et « réseaux » au département GEII de Marseille

Le langage C est étudié au cours des deux premiers semestres de la formation. Le troisième semestre comporte un module sur les réseaux informatiques. Nous avons décidé de le compléter par un module complémentaire obligatoire d'informatique, au cours duquel les étudiants apprennent à utiliser linux et approfondissent l'étude du langage C. Les travaux pratiques de réseaux sont réalisés sur des PC linux. Avant d'aborder le module présenté, les étudiants ont réalisé et programmé des cartes à microcontrôleurs, connaissent le système de fichiers de linux et ses droits d'accès, les protocoles du modèle Internet, les sockets TCP/IP et savent utiliser le logiciel d'analyse de protocoles Wireshark.

1.3 Les modules « d'Études et Réalisations » (ER) au département GEII de Marseille

Dans le cadre de ces modules, les étudiants réalisent des petits projets en une ou plusieurs séances de quatre heures en première année et de six heures en deuxième année.

Ils disposent d'un ordinateur double boot linux et Windows connecté au réseau informatique, de générateurs de fonctions basses et hautes fréquences, d'appareils de mesures, de composants électroniques, d'un fer à souder et de tous les outils nécessaires pour réaliser des cartes électroniques.

2 ORGANISATION DE L'ENSEIGNEMENT

Il est constitué de 12 heures de travaux dirigés (TD), de 28 heures de travaux pratiques (TP) et de 18 heures d'études et réalisations (ER). Nous avons fait le choix d'utiliser ce module pour permettre aux étudiants de découvrir la programmation web. C'est pourquoi les TP sont effectués sur un ordinateur (PC), sous linux Ubuntu et la réalisation est basée sur une carte Raspberry Pi, sous linux Raspbian.

Ces TP sont très guidés. Pour aider les étudiants à bien les comprendre et à les assimiler, le logiciel d'analyse de protocoles Wireshark est systématiquement utilisé. Leur évaluation repose sur des questions précises sur les commandes utilisées et sur l'interprétation de trames capturées avec Wireshark. L'étudiant ne peut avancer dans son TP qu'après avoir réfléchi et répondu à certaines questions.

2.1 Serveur web Apache2 sous linux

- Présentation du logiciel apache2, de ses fichiers de configuration et du protocole http.
- Installation et mise en service d'Apache2 sur un PC linux.
- Analyse de la connexion TCP et des demandes et réponses http avec le logiciel Wireshark.
- Pour permettre l'utilisation des PC sur plusieurs séances par plusieurs groupes, chaque étudiant crée ses propres fichiers de configuration d'Apache2 et le répertoire racine de son site web, dans son répertoire personnel. Seul le fichier de configuration `"/etc/apache2/apache2.conf"` doit être légèrement modifié au début de chaque séance de TP. Les étudiants doivent réfléchir à la

sécurité de leur serveur web et appliquer des droits d'accès adaptés aux répertoires et aux fichiers.

2.2 Les langages HTML et CSS

- Présentation rapide de ces langages.
- Création d'un site web simple en langage html.

2.3 Notions de langages PHP et de javascript

Ces langages sont simplement vus succinctement en TP.

PHP nécessite l'activation d'un module sur le serveur web. L'analyseur de protocoles Wireshark permet de faire comprendre aux étudiants que PHP est exécuté sur la machine serveur et javascript sur la machine cliente (le navigateur) : ils observent que les instructions en javascript sont envoyées directement sur le réseau alors que PHP génère dynamiquement du langage html.

2.4 Traitement de formulaires par un programme CGI en langage C

Un programme CGI, écrit en langage C, traite les données provenant d'un formulaire html rempli par l'utilisateur.

Il génère un flux html vers le navigateur pour y afficher une page web.

- Création d'un formulaire en langage html.
- Étude des liens entre les attributs des éléments html et les informations envoyées sur le réseau.
- Écriture en langage C d'un programme de traitement des données reçues.
- Activation du module CGI et mise en application par des fichiers exécutables en langage script shell, puis langage C.
- Le chemin d'accès au répertoire dans lequel se situent les programmes exécutables et son alias sont indiqués dans un fichier de configuration d'apache2. Nous insistons sur l'importance des droits d'accès pour la sécurité du serveur.
- Analyse des données transmises sur le réseau avec Wireshark.

2.5 Programmation multitâche en langage C sous linux

Étude des processus, des signaux, des threads et des tubes.

- Présentation succincte du fonctionnement d'un système d'exploitation multitâche et des processus.
- Description des processus, signaux, threads et tubes au travers de programmes en langage C utilisant des fonctions telles que : fork, kill, pipe, sigaction, pthread create, etc.

- Des notions de langage C qui n'ont pas été vues dans le tronc commun, comme les pointeurs de fonction sont abordées.
- Les TP s'appuient sur une maquette de mesure de la température ambiante qui communique avec le PC par une liaison USB en émulant un port série. Différentes fonctions sont disponibles comme le changement de la période d'émission et l'autorisation ou l'arrêt de l'envoi des mesures vers le PC. La commande est faite par l'émission d'un caractère ASCII. En retour la maquette envoie des chaînes de caractères ASCII vers le PC.
- La plupart des programmes, écrits en langage C, sont donnés aux étudiants. Ces derniers doivent commenter les résultats obtenus en utilisant, par exemple la commande shell "ps -ely" et en observant le répertoire "/proc" relatif aux processus.

2.6 Réseaux Bluetooth

- Présentation des réseaux Bluetooth et des protocoles associés.
- Écriture en langage C d'un programme client ou serveur utilisant le protocole RFCOMM.
- Configuration du contrôleur bluetooth par des commandes linux : activation / désactivation des drapeaux « Page scan » et « Inquiry scan », de l'appairage simple, de l'authentification, du cryptage des données, etc.
- Développement en langage C d'un programme client et d'un programme serveur utilisant le protocole L2CAP.
- Analyse par Wireshark des paquets échangés entre l'hôte et le contrôleur Bluetooth, fig 1.

No.	Time	Source	Destination	Protocol	Length	Info
48	3...	Controller	host	HCI_EVT	8	Rcvd Number of Completed Packets
49	3...	b8:27:eb:44:30:d...	localhost ()	RFCOMM	69	Rcvd UIH Channel-1
50	4...	b8:27:eb:44:30:d...	localhost ()	RFCOMM	29	Rcvd UIH Channel-1
51	5...	b8:27:eb:44:30:d...	localhost ()	RFCOMM	13	Rcvd DISC Channel-1
52	5...	localhost ()	b8:37:eb:aa:20:4b:f...	RFCOMM	12	Cont. UIH Channel-1
> Frame 50: 29 bytes on wire (232 bits), 29 bytes captured (232 bits) on Interface 0 - Bluetooth - Bluetooth HCI H4 - Bluetooth HCI ACL Packet - Bluetooth L2CAP Protocol - Bluetooth RFCOMM Protocol - Address: E/A flag: 1, C/R flag: 0, Direction: 0, Channel: 1 - Control: Frame type: Unnumbered Information with Header check (UIH) (0xaf), P/F flag: 0 - Payload length: 16 - Frame Check Sequence: 0xd0 > Data (16 bytes)						

```

0000  02 0c 20 18 00 10 00 40 15 00 ef 21 55 69 5e 26  ....0...Zin
0010  56 55 20 63 64 6e 65 70 69 6f 6f 6f 6f 6f 6f  65 70 69 6f 6f 6f 6f 6f 6f 6f 6f 6f 6f 6f 6f

```

fig 1. : Outil Wireshark pour analyse de paquets échangés.

2.7 Liaison SPI et bus I2C

Le cours permet d'aborder les protocoles de communication SPI et I2C, l'aspect électrique (les limitations) et leur utilisation sur le Raspberry Pi en langage C, avec la librairie "WiringPi" (GPIO, SPI et I2C). Leur mise en œuvre est faite dans le module « Étude et Réalisation » (ER), spécifique aux DUT, sur deux applications décrites ci-dessous : l'une en domotique et l'autre en robotique.

3 MISE EN ŒUVRE SUR RASPBERRY PI

3.1 Présentation du Raspberry Pi

C'est un nano-ordinateur à processeur ARM, fig. 2.



fig 2 : Carte Raspberry PI modèle 3B

Le système d'exploitation utilisé est Raspbian qui, comme Ubuntu, est basé sur la distribution linux Debian. Sa prise en main est donc rapide par les étudiants qui connaissent Ubuntu. Nous utilisons le modèle 3B qui est équipé d'un processeur 64 bits à 4 cœurs ARM (Broadcom BCM2837), d'un GPIO à 40 broches, de 4 ports USB, d'un port interface Fast-Ethernet, d'une puce Wifi 802.11n et Bluetooth 4.0. Le système d'exploitation Raspbian est installé sur une carte micro SD.

3.2 Travail demandé aux étudiants

Ils appliquent ce qu'ils ont appris en cours, TD et TP pour activer et configurer un serveur Apache2 sur ce système. Des programmes écrits en langage C permettent de piloter les périphériques (GPIO, Bluetooth, liaison SPI, etc.).

3.3 Application domotique

3.3.1 Présentation de l'application

L'objectif est de piloter l'arrosage et l'éclairage d'un jardin par une interface web, fig. 3.

- Contrôle de l'arrosage par électrovannes à partir d'un planning
- Contrôle des éclairages
- Mesure de la température et de l'humidité du sol (hygrométrie)

Le programmeur est formé de 2 systèmes intelligents qui communiquent par une liaison Bluetooth.

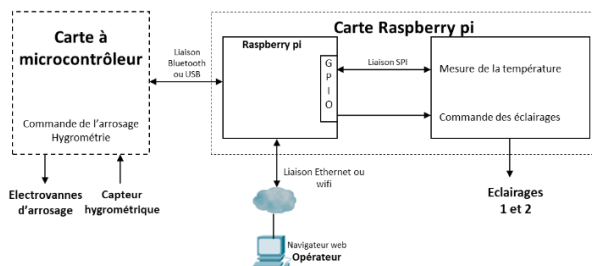


fig. 3 : Système d'arrosage et d'éclairage d'un jardin contrôlé à distance en utilisant une liaison Bluetooth.

Le premier système est constitué par une carte Raspberry pi et par une carte électronique fabriquée par les étudiants.

Il assure 3 fonctions principales :

- Contrôle et commande par le connecteur GPIO du Raspberry pi
 - Commande de 2 éclairages
 - Mesure de la température
- Communication par liaison bluetooth avec le deuxième système intelligent :
 - Émission du planning d'arrosage
 - Émission des commandes de réglage de la date et de l'heure
 - Réception des informations hygrométriques
- Exécution d'un serveur web Apache2 pour une gestion à distance du système par un opérateur à partir d'un navigateur web. Le site web permet de :
 - Contrôler la température
 - Contrôler et commander les éclairages
 - Contrôler l'humidité du sol
 - Créer ou modifier un planning d'arrosage

Le deuxième système est réalisé par une carte à microcontrôleur. Il assure les fonctions suivantes :

- Communication par liaison Bluetooth avec le premier système intelligent
 - Réception du planning d'arrosage
 - Réception des commandes de réglage de la date et de l'heure
 - Émission des informations hygrométriques
- Hygrométrie du sol
- Commande des électrovannes d'arrosage
- Réglage de l'heure et de la date
- Application du planning d'arrosage

3.3.2 Mise en œuvre du Raspberry Pi, de son serveur web embarqué, de ses programmes d'application et développement de la carte de commande des éclairages et de mesure de la température.

Il se déroule en « Etude et Réalisation » sur 3 séances de 6 heures chacune. Les étudiants travaillent en binôme. Ils commencent par graver la carte micro SD avec le logiciel « etcher », à partir d'une image préparée par les enseignants, sur laquelle tous les paquets nécessaires ont été préinstallés. Les étudiants configurent l'interface réseau du Raspberry pi, puis peuvent travailler par bureau à distance sur leur PC de développement en utilisant VNC Viewer.

La carte électronique est réalisée par les étudiants. La mesure de température est effectuée par un capteur LM35 et un convertisseur analogique numérique MCP3201 qui communique avec le Raspberry Pi par liaison SPI. Les étudiants écrivent en langage C leur programme de commande des éclairages et de commu-

nication par liaison SPI, le testent, puis l'interfacent avec le serveur web.

3.3.3 Sujet de projet tutoré : Réalisation de la carte à microcontrôleur

Les étudiants qui le souhaitent peuvent poursuivre la réalisation dans le cadre de leur projet tutoré, en mettant en œuvre les fonctions remplies par la carte à microcontrôleur : liaison Bluetooth sur le microcontrôleur et le Raspberry pi, commande des électrovannes d'arrosage, mesure de l'hygrométrie, etc.

3.4 Application « robot roulant »

3.4.1 Présentation de l'application

L'objectif est de piloter un robot roulant autonome par une liaison wifi, fig. 4.

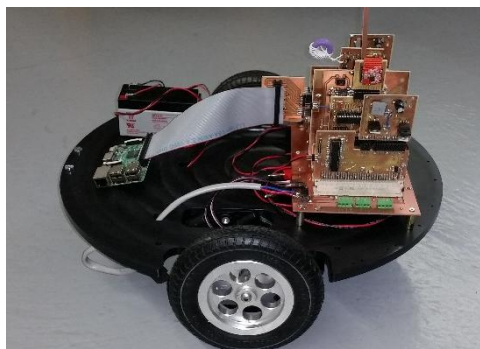


fig. 4 : Robot autonome avec Interface Homme-Machine réalisée par navigateur web utilisant une liaison wifi.

Ce robot est équipé de 2 moteurs à courant continu avec encodeur, d'une batterie 12 V et d'une carte fond de panier, développée par les enseignants et réalisée par les techniciens de l'IUT. Elle permet d'enficher les différentes cartes électroniques développées par les étudiants au cours des ER de 2^{ème} année du DUT.

Un Raspberry Pi permet de piloter le robot à distance par une liaison wifi.

3.4.2 Les cartes développées par les étudiants dans le module pédagogique « Étude et Réalisation »

- Carte convertisseur Buck

Elle illustre le cours d'énergie sur les alimentations à découpage. Elle permet d'adapter la tension de la batterie pour alimenter le Raspberry pi. Elle est réalisée autour du circuit LT1376 de Linear Technology.

Sa réalisation nécessite 3 séances de 6h.

- Commande du moteur

Elle est composée d'une carte mère (commande) et d'une carte fille (puissance).

La partie commande est réalisée autour d'un DSP de Microchip (DSPIC33EP512MC502). Elle communique avec le Raspberry pi par une liaison I2C et génère les PWM (Pulse Width Modulation) pour la carte fille. La programmation est faite en langage C, en utilisant des bibliothèques développées par les enseignants. La partie puissance est réalisée par un pont en H composé de 4

transistors MOS (canal N et P).

Sa réalisation nécessite 4 séances de 6h.

- Carte mesure RF dans la bande ISF (863 - 870 MHz)

Elle permet, entre autres, d'effectuer un relevé de température avec un capteur LM35. Elle est réalisée autour du composant MRF89XA de Microchip connecté sur un microcontrôleur DSPIC33EP512MC502.

Une bibliothèque développée par les enseignants permet de configurer le MRF89XA et de communiquer avec lui.

Une antenne de type doublet est réalisée par les étudiants avec un analyseur de réseau. Elle est dimensionnée par un logiciel « Mmana-Gal ». Elle permet l'observation des signaux RF sur un analyseur de spectre dans la bande ISF (en particulier la modulation FSK et son occupation spectrale).

Cette partie est réalisée sur 4 séances de 6h, à la fin du semestre 3.

3.4.3 Le serveur web sur le Raspberry Pi

Il permet de réaliser l'interface entre l'homme et la machine, par un navigateur web via une liaison wifi. Comme pour l'application domotique, les étudiants configurent l'interface wifi et le serveur web Apache2. Ensuite, ils développent, en langage C, le programme qui permet de communiquer avec les cartes de commande des moteurs via la liaison I2C.

Cette partie vient s'inscrire à la fin du semestre 4.

Elle est réalisée en 3 séances de 6h.

4 CONCLUSION

Au cours de la formation au DUT GEII, certains étudiants décident d'orienter leur parcours professionnel vers l'informatique et les réseaux.

Tout en illustrant une grande partie des thèmes enseignés pendant ces deux années, les applications et la méthodologie que nous avons présentées permettent aux étudiants qui le souhaitent, de découvrir la programmation web, les systèmes embarqués, les systèmes multitâches et la domotique. Ce module complémentaire leur donne les connaissances de base qui leur permettront d'acquérir rapidement d'autres compétences dans ces domaines.

Au vu des retours que nous avons eu, deux années après sa mise en place au département, les étudiants semblent être satisfaits de cet enseignement, ce qui nous encourage à continuer de proposer ce module complémentaire.

Bibliographie

<https://www.wireshark.org/>
<http://httpd.apache.org/docs/2.4/fr/>
<https://doc.ubuntu-fr.org/apache2>
<https://developer.mozilla.org/>
<http://php.net/manual/fr/index.php>
<https://www.bluetooth.com/specifications/>
<https://www.raspberrypi.org/>
<http://wiringpi.com/>
<https://www.balena.io/etcher/>