

Serveur Web embarqué mbed



Objectifs : être capable de mettre en œuvre et d'effectuer des modifications logicielles sur un serveur web embarqué utilisant un module mbed .

Introduction

Dans notre monde hyper-connecté, les objets techniques deviennent tous communicants. Nous sommes dans l'ère de l'internet des objets (IoT : Internet of Things). De nombreux objets techniques disposent d'un serveur web embarqué : un dispositif qui permet de raccorder l'objet technique à un réseau ethernet ou à Internet et de dialoguer avec cet objet technique par l'intermédiaire de pages webs, accessibles à partir d'un navigateur.

L'objectif de ce TP est vous montrer comment implémenter et mettre un œuvre un serveur web embarqué basique sur un module mbed.

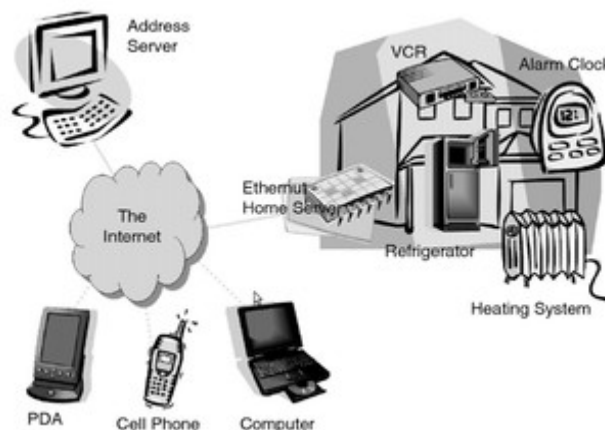
Définition et problématique

Voici ce que l'on peut trouver sur la page wikipedia « Serveur web embarqué » :

« Un serveur web embarqué est un serveur web destiné à être mis en fonction sur un système embarqué et permettant de consulter et d'agir sur ce dernier comme sur n'importe quel élément du web.

De tels serveurs peuvent ainsi prendre place dans des équipements de domotique tels que des matériels électro-ménagers, ou Hi-Fi par exemple, afin d'en permettre le pilotage depuis un poste informatique. Plus généralement, l'implantation d'un serveur web dans un système embarqué permet un accès simplifié à ce dernier à travers une interface Web.

La réalisation de tels serveurs est entravée par de nombreux verrous technologiques. Les systèmes embarqués ont, en général, des capacités limitées afin d'en réduire les coûts de production ou d'en augmenter l'autonomie énergétique. Or les technologies du web, initialement destinées à être opérées sur des serveurs en Rack aux capacités bien plus importantes, n'ont pas été définies pour être utilisées sur ces matériels limités. De là découlent les principales spécificités des serveurs web embarqués. »



Pour en savoir plus, lien :

https://fr.wikipedia.org/wiki/Serveur_web_embarqu%C3%A9

Quelques notions de HTML

1. Késaco HTML ?

Extrait de la page wikipédia HTML : https://fr.wikipedia.org/wiki/Hypertext_Markup_Language

« L'HyperText Markup Language, généralement abrégé HTML, est le format de données conçu pour représenter les pages web. C'est un langage de balisage permettant d'écrire de l'hypertexte, d'où son nom. HTML permet également de structurer sémantiquement et logiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédias dont des images, des formulaires de saisie, et des programmes informatiques. Il permet de créer des documents interopérables avec des équipements très variés de manière conforme aux exigences de l'accessibilité du web. Il est souvent utilisé conjointement avec le langage de programmation JavaScript et des feuilles de style en cascade (CSS). »

2. Analyse de quelques pages HTML

Les pages web suivantes, ont été écrites directement en html. Il existe des logiciels qui permettent d'obtenir du code html à partir d'une interface graphique comme le logiciel freeware « Kompozer » ou même LibreOffice ou Word mais les vrais « geeks » écrivent directement le code html avec notepad++ !



Manipulation n°1

Q.1) Aller chercher sur le serveur le répertoire appelé Pages_simples

Q.2) Ouvrir le fichier index.htm avec un navigateur et suivez tous les liens proposés

Q.3) Compléter le diagramme suivant en y faisant figurer les liens entre les pages :



Q.4) Sur le navigateur Google Chrome ou Firefox, effectuez la commande CTRL-u. Vous pouvez maintenant visualiser le code html correspondant aux pages html visionnées.

Une balise html se présente généralement comme suit
<balise></balise>. Exemple <title>index.htm</title>

Q.5) Analysez le code des pages visionnées. Sur les pages visionnées, quelle balises correspondent à :

- un nouveau paragraphe
- un titre de texte de 1^{er} niveau
- un lien
- une image
- du texte gras
- du texte italique
- du texte souligné

Ouvrir maintenant les pages html avec Notepad++, vous accédez ainsi au code html correspondant aux pages, comme avec ctrl-u sur votre navigateur mais vous pouvez par contre maintenant modifier les pages HTML.

Q.6) Modifier les pages 1 et 2 pour y mettre un lien vers Google. Vérifier le fonctionnement de votre modification.

Q.7) Créer une page appelée page3.htm (avec la méthode de votre choix) contenant le texte de votre choix (en respectant la politesse et la bien-séance) et les liens permettant de revenir au menu et aux autres pages. Modifier les autres pages pour avoir les liens vers cette nouvelle page.

Bonus : si vous êtes en avance, insérez sur cette page 3 une nouvelle image qui contient un lien accessible en cliquant sur l'image ! Faites vous aider par Google !

Avertissement 1 : le HTML a beaucoup de possibilités, le but de ce TP n'est pas de faire de vous des développeurs web mais juste de vous présenter quelques notions de HTML. Les fichiers d'exemples HTML présentés ne sont pas forcément optimisés.

Avertissement 2 : Pour les manipulations à venir, si vous devez modifier des pages html hébergées sur mbed, modifiez les pages sur PC, puis transférez-les sur le mbed. Ne pas éditer les pages directement sur mbed, cela peut endommager le fichier édité !

Mettre en œuvre un premier serveur web mbed

Pour effectuer les manipulations qui suivent, vous aurez besoin, en plus d'un PC, d'une carte d'évaluation mbed équipée d'un mbed, d'un câble mini-USB, d'un câble ethernet.



Manipulation n°2

Q.1) Connecter le mbed par USB au PC et placez-y le fichier webserver_LPC1768.bin (fourni) ainsi que les fichiers html et l'image associée.

Q.2) Relier le port ethernet du PC au port ethernet de la carte d'évaluation mbed.

Q.3) Modifier la configuration du PC (pensez à noter la configuration initiale avant modification) sachant que les paramètres réseau du serveur web embarqué sont les suivants :

(192.168.1.102), // IP	(255.255.255.0), // Subnet mask
(192.168.1.1), // Gateway	(192.168.1.1); // DNS

Q.4) Vérifier par ipconfig/all et un « ping » que le dialogue se fait bien entre le PC et le serveur web embarqué.

Q.5) Vérifier que l'on accède bien au serveur web lorsque l'on entre l'adresse IP du serveur web dans la barre d'adresse du navigateur. Vérifier l'accès aux autres pages.

Q.6) Le lien vers google.fr est-il accessible ? Pourquoi ?

Q.7) Renommer le nom de la page index.htm en accueil.htm. Faire un reset puis refaire les manipulations précédentes. Que se passe-t-il ? Existe-t-il un moyen de visualiser la page d'accueil ? Conclure.

Modifier le serveur web mbed

Le code source du fichier que vous avez implémenté précédemment est disponible à l'adresse suivante :

<https://os.mbed.com/users/LouisReynier/code/webserver/>

Pour la suite du TP, vous aurez besoin d'un accès à Internet : reconfigurer le PC pour avoir l'accès à Internet ou mieux, allez sur un autre poste (si il y en a un de libre). Vous vous connecterez à votre compte mbed.



Manipulation n°3

Q.1) Importer le code source du fichier dans votre espace. Compiler-le, remplacer le fichier binaire et vérifier le bon fonctionnement.

Q.2) Modifier l'adresse IP du serveur web en utilisant par exemple l'adresse 192.168.1.103, recompiler et vérifier le bon fonctionnement.

Q.3) Modifier l'adresse IP du serveur web en utilisant les paramètres IP du PC qui était connecté au réseau précédemment. Re-compiler, et vérifier que vous pouvez accéder au serveur à partir d'un autre poste de la salle connecté au réseau. Vérifier que le lien vers google fonctionne, maintenant que le poste sur lequel vous travaillez est relié à internet.

Q.4) Expliquer quelles sont les actions effectuées par le mbed lorsque le programme fonctionne en « régime permanent ».

Q.5) Déterminer la durée d'allumage/extinction de la led. Quel est l'intérêt de la technique utilisée par rapport à une temporisation « classique » (wait(x) ou delay(y)).

Mettre en œuvre un deuxième serveur Mbed

Le code source de ce serveur est disponible à l'adresse suivante :

https://os.mbed.com/users/LouisReynier/code/HTTPServerExample_LR/

Vous aurez aussi besoin du fichier index.htm (disponible lui aussi à l'adresse donnée, lisez bien la page) et de l'image serveur.jpg (fournie).

La page doit avoir l'allure suivante :

192.168.1.102

Rechercher

Les plus visités Débuter avec Firefox

Serveur MBED



Designed by LR d'apres Lotfi - v0.2

LED2: On

la valeur du potentiometre est : 0.13357754051685333
la valeur du compteur count est : 1491

[Lien vers les infos du jour](#)



Manipulation n°4

Q.1) Importer le code source du fichier dans votre espace de travail. Compiler-le, implanter le fichier binaire dans le mbed. Copier aussi dans le mbed le fichier index.htm et l'image server.jpg.

Q.2) Vérifier que la page affichée est conforme à l'image.

Q.3) Observer votre module mbed et noter ce qui se passe quand on clique sur le bouton LED2 : ON.

Q.4) Effectuer plusieurs rafraîchissement de la page. Que ce passe-t-il sur la variable count ?

Q.5) Cliquer le bouton « Reset variable Count » et observer ce qui se passe. Donner le rôle de cette variable et de ce bouton.

Q.6) Faire varier le potentiomètre Pot1 du kit. Rafraîchir la page et observer. Qu'affiche-t-on lorsque le potentiomètre est en butée gauche et en butée droite ? Conclure.

Contrairement au premier serveur web étudié, ce serveur web permet d'interagir avec mbed et donc avec l'extérieur !

Ce serveur utilise JavaScript associé au html ainsi que les commandes mbed RPC : remote procedure call.

Nous allons maintenant voir comment tout cela fonctionne !

1. Définition et mise en œuvre sur mbed

RPC est l'acronyme de Remote Procedure Call. En informatique et en télécommunication, RPC (remote procedure call) est un protocole réseau permettant de faire des appels de procédures sur un ordinateur distant à l'aide d'un serveur d'applications.

Sur mbed, il est possible d'utiliser des commandes RPC via la liaison série mais nous n'utiliserons dans ce TP que des commandes RPC via http : RPC Over http.

Les textes anglais présentés ci après sont extraits de la documentation présente sur le site mbed, accessible en suivant ce lien :

<https://os.mbed.com/cookbook/Interfacing-Using-RPC>



Travaux dirigés 1

Q.1) Traduire le paragraphe suivant, extrait de la documentation présente sur le site mbed.

The mbed is capable of receiving and interpreting RPC commands and this can be used to greatly simplify creating an interface. The RPC commands are in a predefined format and can be sent over any transport mechanism that can send a stream of text. They allow you to directly interact with objects on mbed.

Q.2) Traduire le texte suivant, extrait de la documentation présente sur le site mbed.

Remarque : handler pourrait être traduit en français par « gestionnaire d'événements »

RPC Over HTTP

The HTTP Server has an rpc handler, so by using the HTTP server example program below and download it on to your mbed you will be able to use RPC over HTTP. The rpc commands are sent by adding them to the URL of the mbed in a browser.

RPC Commands over HTTP are in the format: http://<url of mbed>/rpc/<Object name>/<Method name> <Arguments separated by spaces>

The important line of code for RPC is to add the RPC handler.

```
svr.addHandler<HTTPRpcRequestHandler>("/rpc");
```

It is also important to add the RPC classes you are using. For instance, the following line of code is important if you intend to use RPC on a DigitalOut object.

```
RPC::add_rpc_class<RpcDigitalOut>();
```

2. Mise en œuvre de RPC

Vous allez analyser le code du serveur web précédent.

Manipulation n°5

Q.1) Le handler RPC est-il bien présent ? A quelle ligne est il ajouté ?

Q.2) Entrez dans la barre d'adresse les commandes

<http://192.168.1.102/rpc/led2/write+1>

puis

<http://192.168.1.102/rpc/led2/write+0>

Que font ces deux commandes ?

Q.3) Allumer et éteindre par RPC la led 4. Quelles commandes avez-vous effectué ?

Q.4) Entrez dans la barre d'adresse les commandes suivantes et indiquer leurs actions ou fonctions :

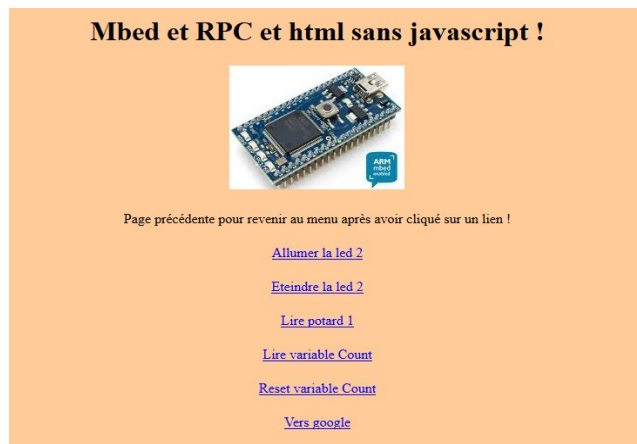
<http://192.168.1.102/rpc/pot1/read>

<http://192.168.1.102/rpc/count/read>

<http://192.168.1.102/rpc/count/write+0>

Q.5) Question bonus : écrire une page web html (à implémenter sur votre mbed) et présentant des liens cliquables permettant d'allumer et d'éteindre une led, de lire l'entrée reliée au potentiomètre ou de lire ou remettre à 0 la variable count.

Exemple d'implémentation :



Quelques notions de JavaScript

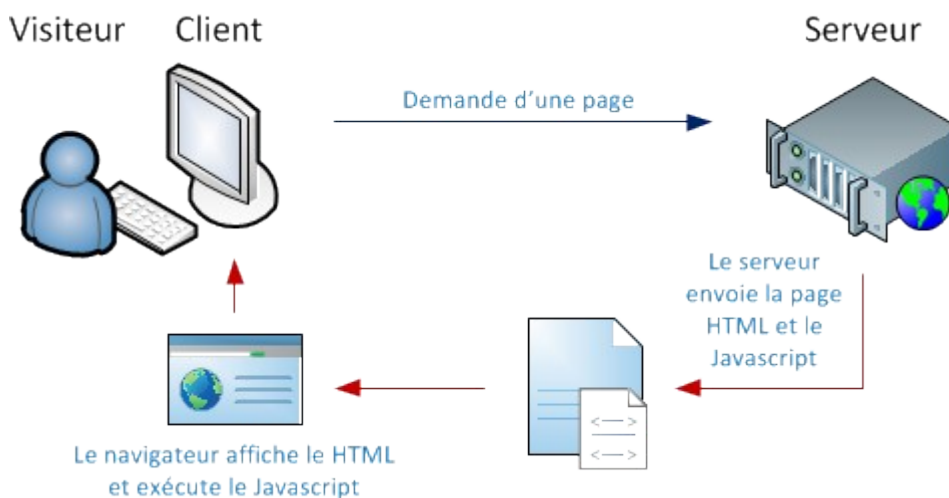
1. Késaco JavaScript ?

Le langage HTML permet de faire des mises en forme de texte mais ne permet pas vraiment d'interactivité, une page HTML reste figée.

JavaScript a été inventé en 1995 par Brendan Eich, qui travaillait à l'époque pour la société Netscape, créatrice du premier navigateur Web populaire (l'ancêtre de Firefox). L'idée de départ était de créer un langage simple pour rendre dynamiques et interactives les pages Web, qui étaient très simplistes à l'époque.

Le JavaScript est majoritairement utilisé sur Internet, conjointement avec les pages Web HTML. Le JavaScript s'inclut directement dans la page Web (ou dans un fichier externe) et permet de *dynamiser* une page HTML, en ajoutant des interactions avec l'utilisateur, des animations, de l'aide à la navigation, comme par exemple :

- Afficher/masquer du texte ;
- Faire défiler des images ;
- Créer un diaporama avec un aperçu « en grand » des images ;
- Créer des infobulles.



Pour plus d'informations :

<https://openclassrooms.com/courses/dynamisez-vos-sites-web-avec-javascript>

Pour un programmeur qui maîtrise le C/C++, le langage JavaScript ne devrait pas poser de problèmes. La syntaxe est assez ressemblante !

2. Analyse d'une page JavaScript simple

A partir du fichier donné : inter_led.htm



Manipulation n°6

- Q.1)** Ouvrir la page et analyser le fonctionnement de la page. On observera en particulier les différentes couleurs de la led.
- Q.2)** Ouvrir le code source de cette page. Quelles lignes sont du HTML, quelles lignes sont du JavaScript ?
- Q.3)** Quelle est le nom de la fonction JavaScript ?
- Q.4)** Quelles sont les variables utilisées par le script ? A quoi correspondent-elles ?
- Q.5)** Informatiquement, que se passe-t-il quand on clique sur le bouton Allumer/Eteindre?
- Q.6)** Comment le JavaScript agit sur le HTML ?
- Q.7)** Modifier le programme pour faire varier, en plus de la couleur, la taille de la led. Par exemple Led éteinte : largeur 50 pixel , led allumée largeur 200 px, état initial 150 px.
- Q.8)** Rajouter une seconde « led » de même dimension mais de couleur bleue lorsque elle est allumée qui s'allume lorsque la led rouge est éteinte.
- Q.9)** Comment sont indiqués les commentaires en HTML ? En JavaScript ?

3. Analyse de la page du serveur web mbed

Vous allez maintenant analyser la page index.htm du serveur web mbed.



Manipulation n°7

- Q.1)** Ouvrir le code source de cette page. Quelles lignes sont du HTML, quelles lignes sont du JavaScript ?
- Q.2)** Faites l'inventaire des différentes fonctions JavaScript codées dans le fichier.
- Q.3)** Lister les différentes variables utilisées par ces fonctions. Indiquez leur rôle et leur signification.
- Q.4)** Expliquer comment sont envoyées les commandes RPC. Repérez les lignes de code correspondantes.
- Q.5)** Expliquer comment le script JavaScript écrit du texte sur la page HTML. Repérez les lignes de code correspondantes.

Les bonus !

Si vous êtes en avance, vous pouvez travailler sur les problèmes suivants.



Bonus

Bonus 1 : Modifier la page web du serveur pour commander les 3 leds du Mbed.

Bonus 2 : Modifier le programme du Mbed et la page web du serveur pour afficher Bonjour sur l'afficheur LCD à partir d'un bouton de la page web.

Bonus 3 : Écrire un page HTML et JavaScript (sur PC) qui demande 1 chiffre puis affiche sa table de multiplication.

Bonus 4 : Si vous avez fait tous les bonus précédents, faites un serveur web mbed qui impressionne vos camarades et étonne vos professeurs !

Retrouvez d'autres cours et documents sur :

<http://www.louisreynier.com>