**Chapitre : Informatique industrielle**

**Être capable**

- d’écrire un algorithme (écrit du bac et projet) et de le traduire en un programme (projet) [Aspect logiciel]

- de choisir un microcontrôleur (limité à l’identification des E/S en projet) [Aspect matériel]

**Durée :** 4 semaines

**Position dans l’année :** Révisions -> Chaîne de mesure -> **Informatique Industrielle** -> Réseaux -> Motorisation -> Identification

**Matériel, Logiciel** : technologie .Net, Visual studio, C#

**Ressources**: [Wikispaces](https://csharpembarquenetduino.wikispaces.com/), NAS\_SIN

**B.O. Bulletin officiel spécial n°9 du 30 septembre 2010**

**A2. Analyser le système**

|  |  |  |  |
| --- | --- | --- | --- |
| **Connaissances** | **Capacités** | **1er** | **T** |
| Systèmes logiques évènementiels  Langage de description : graphe d’états, logigramme, algorigramme | Décrire et analyser le comportement d’un système. | C |  |
| Composant réalisant les fonctions de la chaîne d’information | Justifier la solution choisie |  | B |
| Système de numération codage | Analyser et interpréter une information numérique | C |  |

**B2. Proposer ou justifier un modèle**

|  |  |  |  |
| --- | --- | --- | --- |
| **Connaissances** | **Capacités** | **1er** | **T** |
| Chaîne d’information | Associer un modèle aux composants d’une chaîne d’information |  | C |
| Systèmes logiques à évènements discrets  Langage de description : **graphe d’états**, logigramme, **algorigramme** | Traduire le comportement d’un système |  | C |

**C2. Mettre en œuvre un protocole expérimental**

|  |  |  |  |
| --- | --- | --- | --- |
| **Connaissances** | **Capacités** | **1er** | **T** |
| Routine, procédure etc  Systèmes logiques à évènements discrets | Générer un programme et l’implanter dans un système cible |  | C |

**Planning : Cours / Travaux Dirigés/TP**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sem** | **Titre** | **Contenu / Objectifs** | **Ressources** | **Remarques** |
| **S1**  **Cours**  **(1h)** | Algorithmique  Affectation | Travail élève : Ecrire l’algorithme MesureAngle du *"TP0 Informatique industrielle Programmation Orientée Objet (Introduction)"*  **1 - Matériel : Généralités µC**  - Notion de port d’entrée, sortie (matériel)  **2 – Logiciel : Algorithmique**  - Notion de type des variables  - Rappel des règles d’écriture d’un algorithme en II  (Nom, Parties déclarative, exécutive dans une boucle infinie ou composée de routine d’interruption) | **Documents**  - Généralités sur les systèmes à microcontroleur.doc  - Cours\_Algorithmique\_TS .doc + Exercices du cours | A distribuer (commentaires)  A mettre à jour Netduino 3 |
| **S1**  **TP0/TD**  **(2h)** | **Introduction II**  **Prise en main**  **de Visual Studio** | Objectif : Adopter une démarche de conception lors de la réalisation d’un programme relevant de l’informatique industrielle.  Travail élève : Prendre en main Visual Studio.  Remarques   * Faire apparaître le fait que tout programme d’II comporte deux aspects (matériel et logiciel). * Commenter l’exemple « MesureAngle » et faire apparaître la démarche de conception illustrée dans l’annexe du document. **TS\_II\_TP0\_Intro\_CSharp.docx**. * Prise en main de Visual Studio d’après Ch4 wiki Netduino * Création projet et programmation « MesureAngle » * TD avec PC et Tome 1 Netduino C# (Exemples… | **Documents**  - TP0 Informatique industrielle Programmation Orientée Objet (Introduction)  **- NETMF - C# : Exemples pour les cartes Netduino - Tome 1 (Sur Wiki)**  - Résumé de langage C#  **Lien**  Wiki Netduino  **Matériel**  Netduino, Tinkerkit, LCD, pot | Le 6/1/2016  TD Configuration des ports d’entrées, sorties non fait en cours par manque de temps -> travail maison)  **PC en administrateur et ressources sur NAS\_SIN** |
| **S2**  **Cours**  **(1h)** | Algorithmique  Affectation  Alternatives | Correction "*Configuration des ports d’entrées, sorties*" dans TP0  Exercices sur les structures alternatives  - Affectation, si alors sinon | - Cours\_Algorithmique\_TS + Exercices du cours | **Maison** : Algorithme TP1a MiniSerre |
| **S2**  **TP1a**  **(2h)** | **[MiniSerre]**  **Contrôle de la température dans une enceinte** | Objectif : Contrôler la température dans une enceinte (régul. TOR)  Remarques   * Lien avec le cours « Capteurs et chaîne de mesure » * Algorithmique (affectation, alternatives)   - Version 1a : Mesure et affichage de la température  - Version 1b : Commande du chauffage  - Version 1c : Programmation d’une alarme | **Documents**  - Sujet + DR du TP1a  - Résumé de langage C#  **Lien**  Wiki Netduino  **Matériel**  Netduino, Tinkerkit, LCD, pot, Thermomètre GHI | **Synthèse non faite en 1/2016**  **Faire maquette enceinte pour 2017** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sem** | **Titre** | **Contenu / Objectifs** | **Ressources** | **Remarques** |
| **S3**  **Cours**  **(1h)** | Algorithmique  Alternatives | Objectif : Connaître les structures alternatives utilisées pour coder un graphe d’états.  Exercices sur les structures alternatives  - si imbriqué, selon…. | - Cours\_Algorithmique\_TS + Exercices du cours |  |
| **S3**  **Cours/**  **TD**  **(2h)** | **Graphe états**  **/**  **[MiniSerre]**  **Contrôle de la température dans une enceinte** | Objectif : Ecrire les algorithmes du TP1b Miniserre (Humidité + Diagnostique) [Préparation du TP1b]  Cours : Graphe état  Préparation du TP1b : Mesure humidité, algorithme Superviser, Algorithme diagnostique | **Documents**  - Cours\_Algorithmique\_TS + Exercices du cours  - Sujet + DR du TP1b  - Fiche graphe état (UML)  **Lien**  Wiki Netduino | Fiche graphe état (UML) A rédiger  Acheter AltovaUML |
| **S4**  **Cours**  **(1h)** | Algorithmique  Graphe d’état | Objectif : Dessiner et coder un graphe d’états simple  Exercices du cours (graphes d’états)  Exercices du cours (Structure alternatives) | - Cours\_Algorithmique\_TS + Exercices du cours | Voir site SI |
| **S4**  **TP1b**  **(2h)** | **[MiniSerre]**  **Contrôle de l’humidité dans une enceinte**  **Découpage du programme en modes de fonctionnement** | Objectif : Implémenter les algorithmes Humidité + Diagnostique | **Documents**  - Sujet + DR du TP1b  - Résumé de langage C#  **Lien**  Wiki Netduino  **Matériel**  Netduino, Tinkerkit, LCD, pot, Thermomètre GHI | **Synthèse non faite en 1/2016**  **Faire maquette enceinte pour 2017** |
| **S5** | DS |  |  |  |

Autres TP possibles

TP2 : Doseuse (Motorisation/Codeur)

TP3 : IHM Gadgeteer

A faire

QCM algorithmique