



Prise en main de l'environnement de programmation du microcontrôleur

Lyes Saad Saoud

► To cite this version:

Lyes Saad Saoud. Prise en main de l'environnement de programmation du microcontrôleur. Master. Algérie. 2017. hal-04390853

HAL Id: hal-04390853

<https://hal.science/hal-04390853>

Submitted on 17 Jan 2024

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.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

UNIVERSITE M'HAMED BOUGARA-BOUMERDES



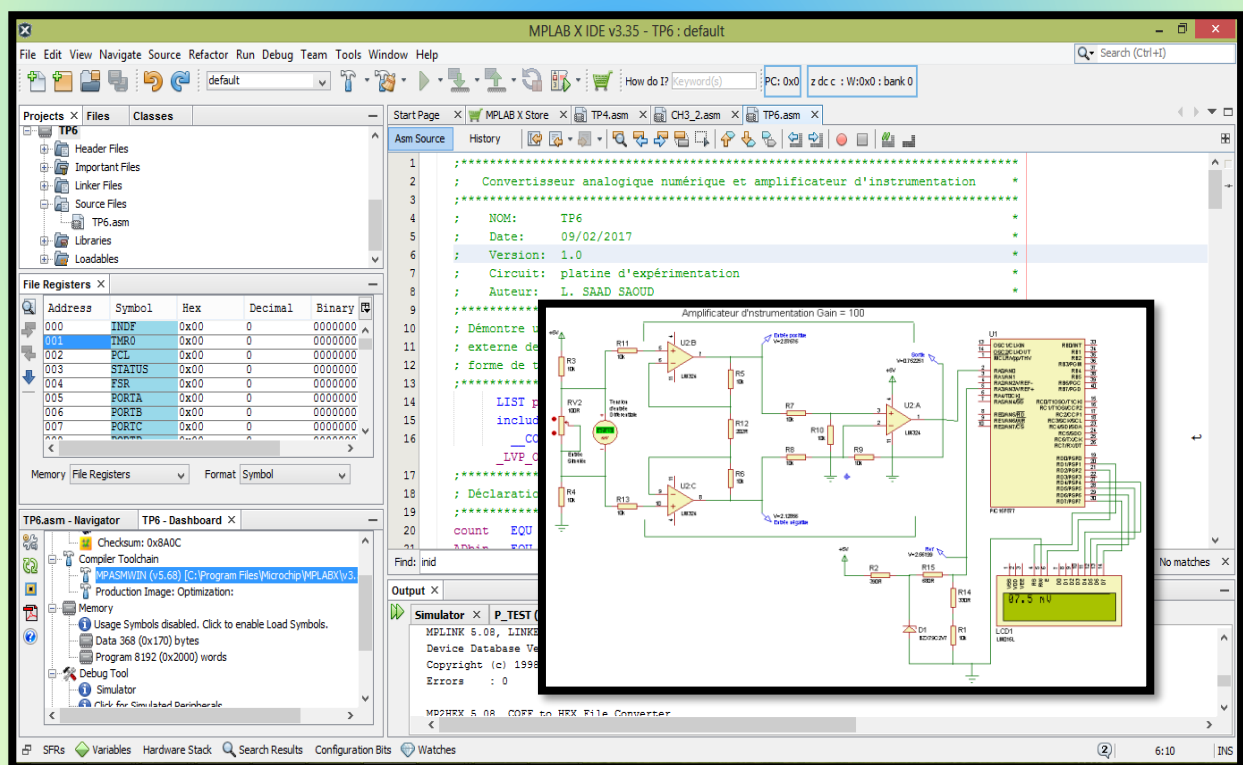
L. SAAD SAOUD

Systèmes à microcontrôleurs

Travaux pratiques en utilisant MPLAB X et Proteus

Master 1 : Instrumentation biomédicale

1ère Edition



Manuscrit préparé et élaboré selon le programme officiel du ministère de l'enseignement supérieure et de la recherche scientifique

Année Universitaire : 2016/2017

Avant-propos

Le présent manuscrit a été préparé et élaboré conformément au programme officiel du ministère de l'Enseignement Supérieur et de la Recherche Scientifique. L'objectif de ce support de cours, destiné au module "Systèmes à microcontrôleurs", est de regrouper et d'expliquer les bases de la conception des circuits électroniques utilisant des microcontrôleurs PIC de Microchip. Il a été rédigé spécialement pour les étudiants en première année du Master en Instrumentation Biomédicale, mais il peut également être utile pour d'autres spécialités nécessitant l'utilisation des circuits fondamentaux décrits dans ce support de cours.

Le chapitre 1 offre une explication détaillée des différences entre le microprocesseur et le microcontrôleur, des architectures Harvard et Von Neumann, ainsi que des architectures RISC et CISC.

Le chapitre 2 propose une étude approfondie de l'architecture interne et externe des microcontrôleurs PIC.

Le chapitre 3 se consacre à la programmation en langage assembleur des PICs.

Le chapitre 4 présente les systèmes de développement, y compris l'édition, la compilation et le débogage à l'aide des logiciels MPLAB X et PROTEUS.

Le chapitre 5 est dédié aux interruptions et à leur programmation.

Dans le chapitre 6, nous examinerons les interfaces du microcontrôleur, notamment la lecture-écriture dans la mémoire EEPROM et les liaisons série (USART, MSSP) avec les réseaux locaux de communication SPI, I2C, etc.

Enfin, le dernier chapitre contient des travaux pratiques couvrant l'essentiel des différents points décrits au cours de ce manuscrit.

Copyright

Copyright © 2017, Lyes SAAD SAOUD. Tous droits réservés.

Le droit de nom d'auteur doit être identifié comme l'auteur de ce travail a été affirmé conformément à la Loi de 1988 sur le droit d'auteur, les dessins et modèles et les brevets.

Aucune partie de cette publication ne peut être reproduite, stockée dans un système de recherche documentaire ou transmise sous quelque forme ou par quelque moyen que ce soit électronique, mécanique, photocopie, enregistrement ou autrement sans l'autorisation écrite préalable de l'auteur

Les autorisations peuvent être demandées directement à l'auteur en envoyant un email à l'adresse : saadsaoudl@univ-boumedres.dz.

1.1.TP1 : Prise en main de l'environnement de programmation du microcontrôleur

1.1.1. Objectif

L'exercice du TP1 montre comment créer, éditer et construire des projets avec MPLAB® X IDE. Ce laboratoire est une étape par étape à travers le développement du projet MPLAB X. Un projet MPLAB X est créé et un fichier source C existant est ensuite ajouté au projet. Le laboratoire continue avec l'édition du fichier source et la construction réussie d'un projet. Le laboratoire comprend des exercices pour démontrer certaines fonctionnalités utiles de manipulation de fichiers et de données de MPLAB X IDE

1.1.2. Création d'un projet MPLAB X

1.1. Installation du logiciel MPLAB X

- Télécharger MPLAB X depuis le site officiel :

<http://www.microchip.com/mplab/mplab-x-ide>

- Exécuter l'icône d'installation



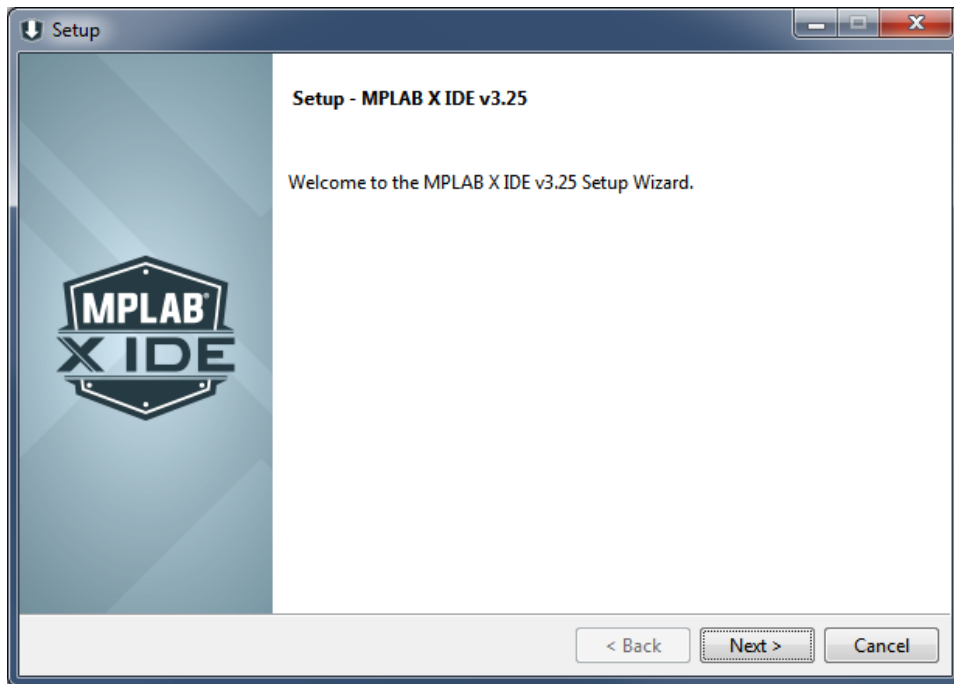
Allez à l'emplacement où vous avez téléchargé le programme d'installation. Décompressez le fichier téléchargé et exécutez le programme d'installation :

MPLABX-vX.XX-windows-installer.exe.

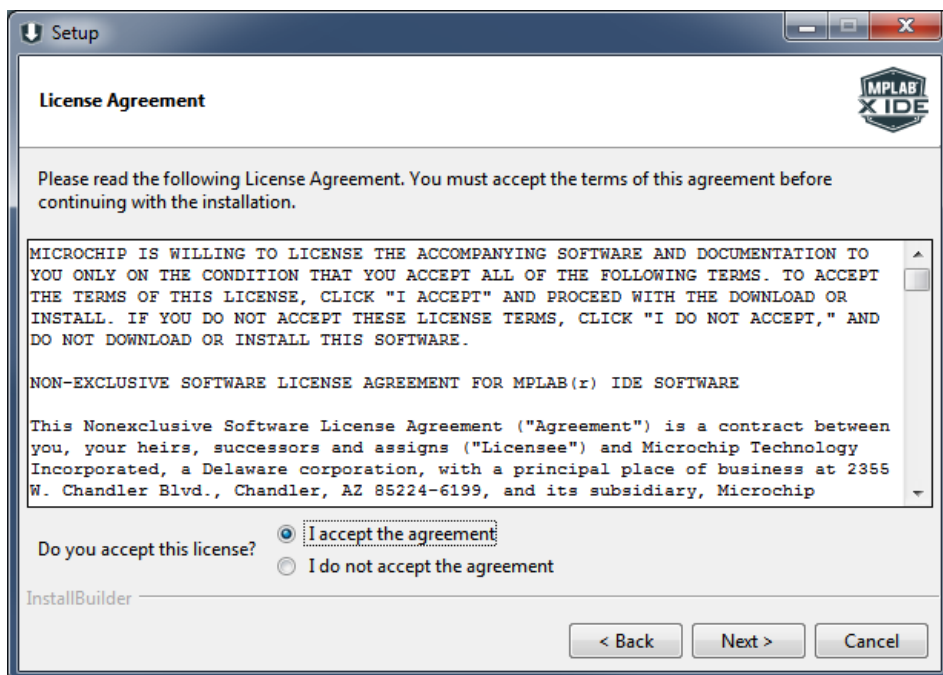
Selon vos paramètres de sécurité Windows, vous pouvez obtenir une fenêtre demandant si vous êtes sûr que vous voulez exécuter ce programme. Répondez «Oui».

- **Installer**

Cliquez sur Suivant>.



- **Accord de licence**



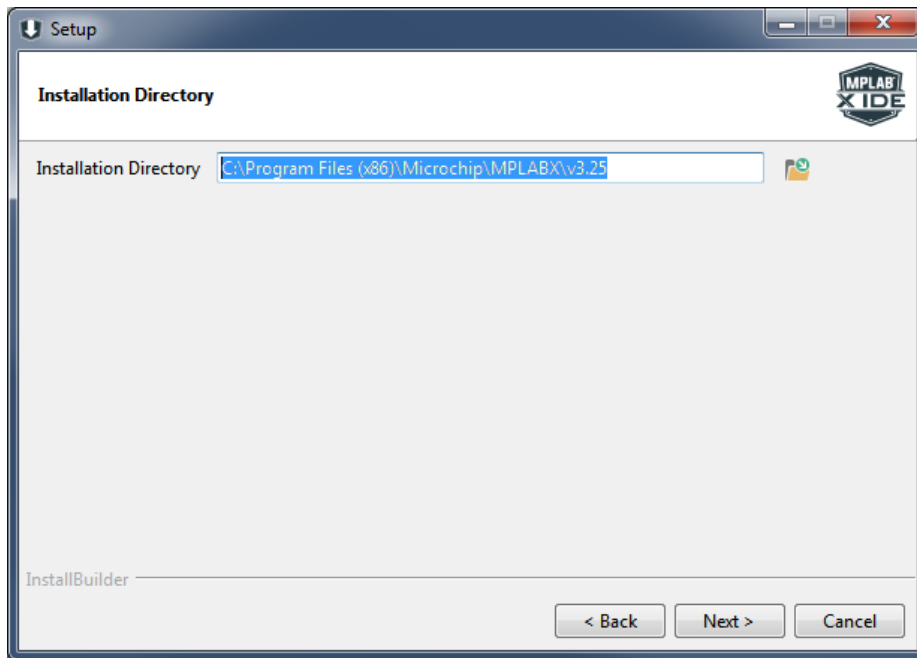
Cliquez sur le bouton radio en regard de "J'accepte l'accord".

Cliquez sur Suivant>.

- **Répertoire d'installation**

Par défaut, MPLAB X IDE sera installé à C:\ Program Files (x86) \ Microchip \ MPLABX. Si vous préférez un répertoire différent, cliquez sur l'icône de dossier à droite de la zone de texte et sélectionnez l'emplacement d'installation souhaité.

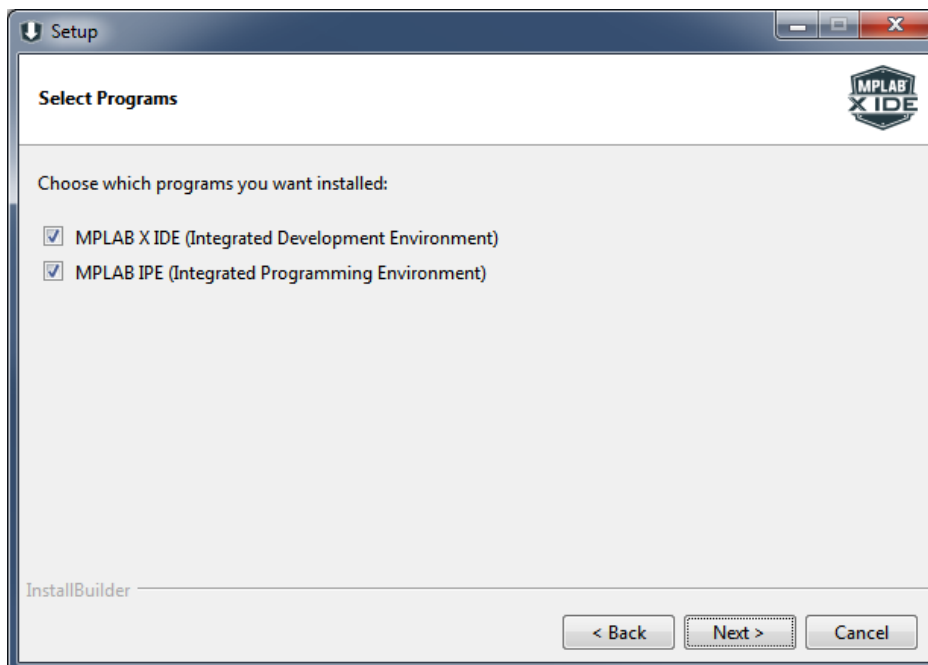
Cliquez sur Suivant>.



- **Sélectionner des programmes**

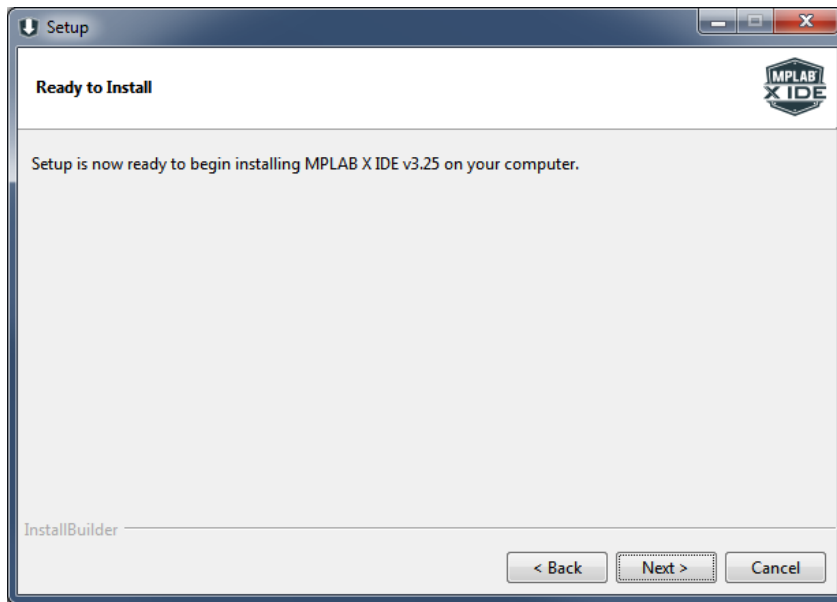
Si vous souhaitez installer MPLAB X IDE (environnement de développement intégré) ou MPLAB IPE (environnement de programmation intégré), cochez / décochez les cases appropriées. Généralement, vous devez installer les deux programmes.

Cliquez sur Suivant>.



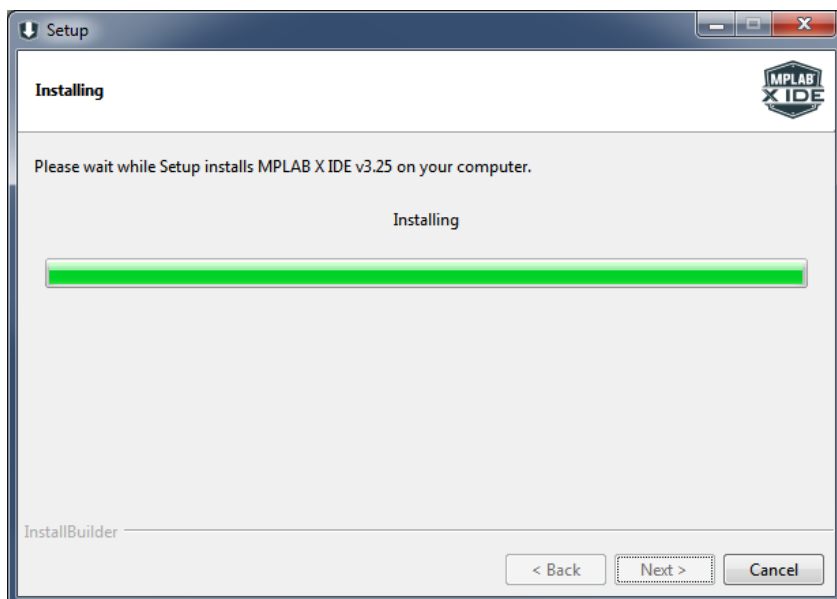
- **Prêt à installer**

Cliquez sur Suivant>.



- **Installation**

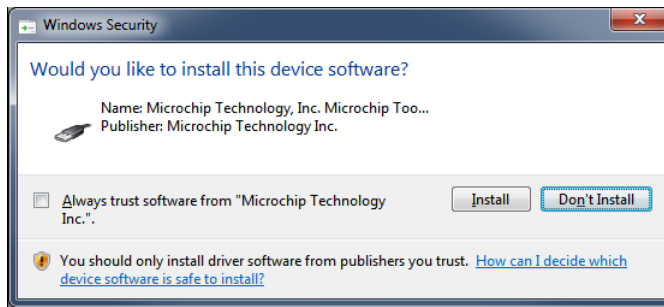
Attendez que le programme d'installation ait terminé l'installation de tous les composants de l'IDE.



- **Permettre l'installation d'un pilote de périphérique**

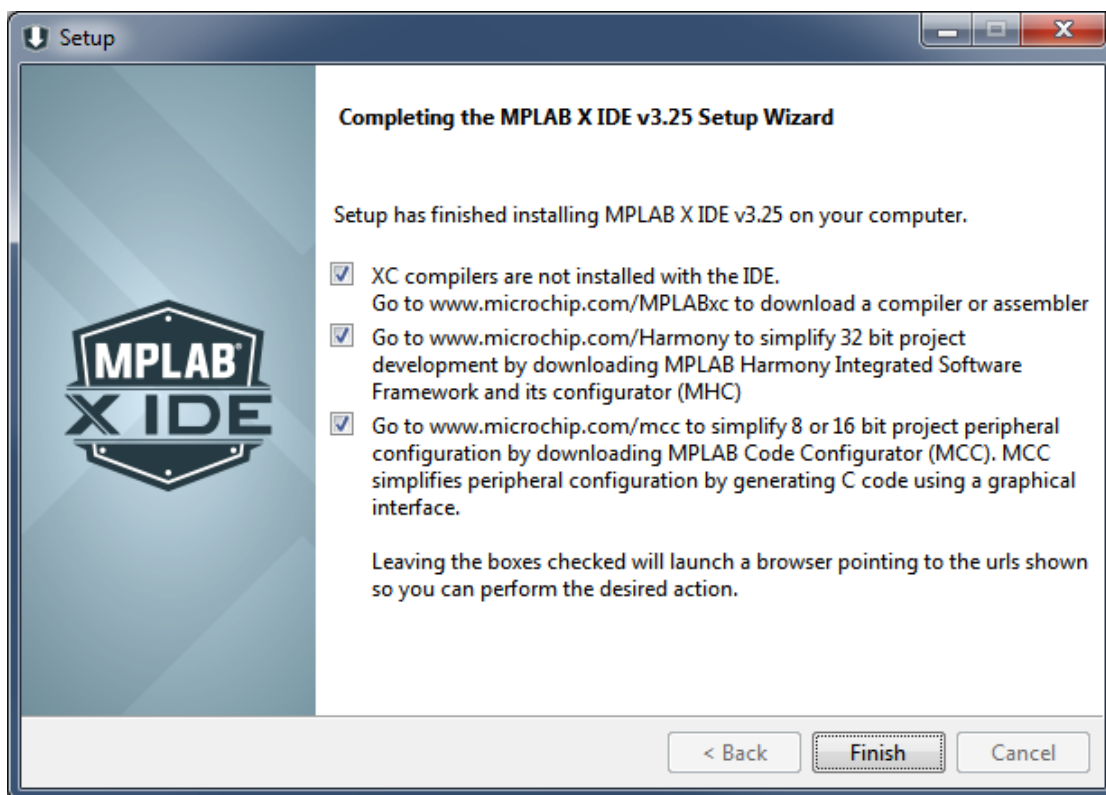
Les pilotes de périphériques USB doivent communiquer avec les outils de développement matériel de Microchip.

Cochez la case à côté de «Toujours faire confiance au logiciel de Microchip Technology» pour empêcher cette boîte de dialogue à l'avenir.



- **Achevée**

Laissez la case cochée si vous souhaitez que votre navigateur Web soit ouvert sur la page de téléchargement du compilateur MPLAB XC de Microchip pour télécharger un compilateur à utiliser avec MPLAB X IDE.



Décochez la case si vous avez déjà un compilateur ou si vous souhaitez en télécharger un plus tard.

Cliquez sur Terminer.

L'installation est terminée. Il devrait y avoir des icônes pour MPLAB X IDE, MPLAB IPE et le commutateur de pilote MPLAB sur le bureau. Il y aura également des lanceurs dans le menu Windows / Démarrer sous Tous les programmes ► Microchip ► MPLAB X IDE.

1.1.3. Notre premier projet

- **Création de projet**

Ouvrir MPLAB X IDE

Fermez tous les projets ouverts dans MPLAB X IDE en cliquant avec le bouton droit de la souris sur le nom du projet et en sélectionnant Fermer ou en allant dans Fichier et en sélectionnant Fermer tous les projets.

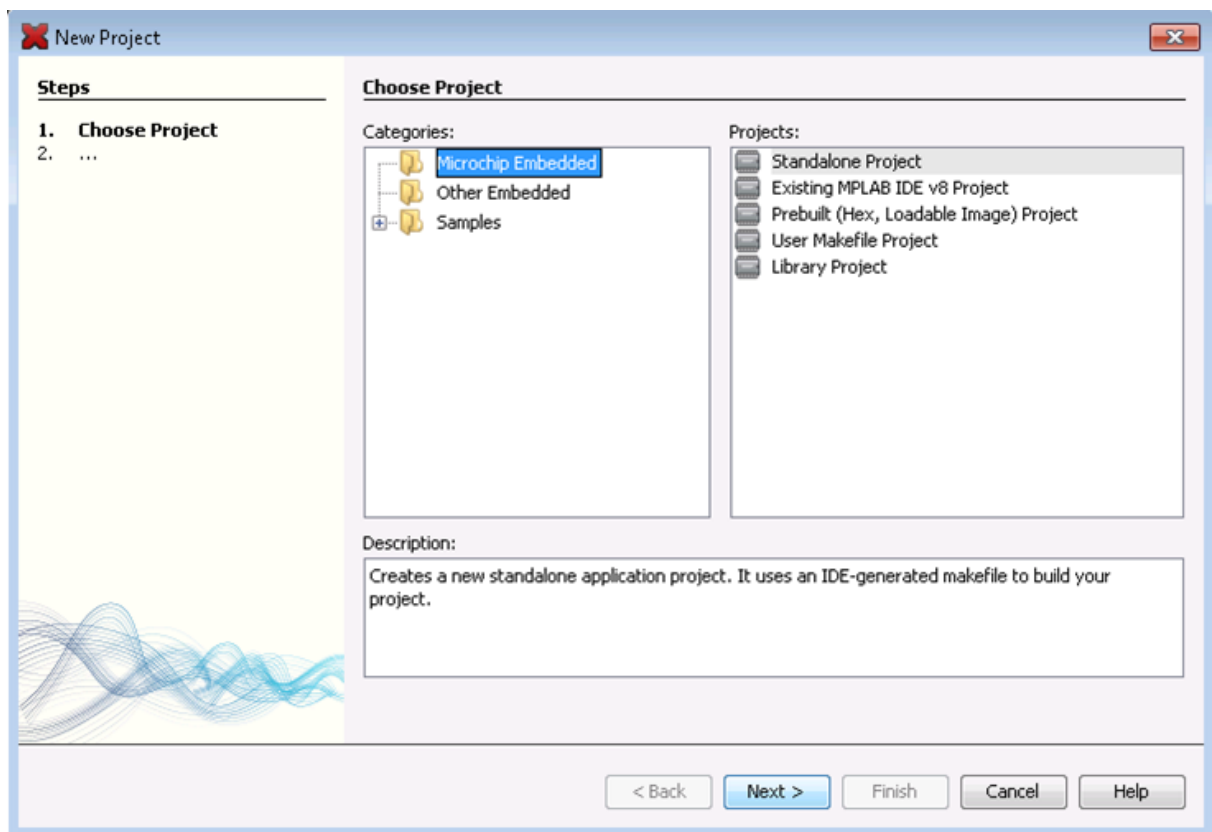
Lorsque MPLAB® X démarre, il ouvre le dernier projet travaillé. Pour éviter toute confusion, ce laboratoire vous demande de fermer tout projet ouvert

- **Début de la création du projet**

Cliquez sur l'icône Nouveau projet  pour lancer le processus de création du projet

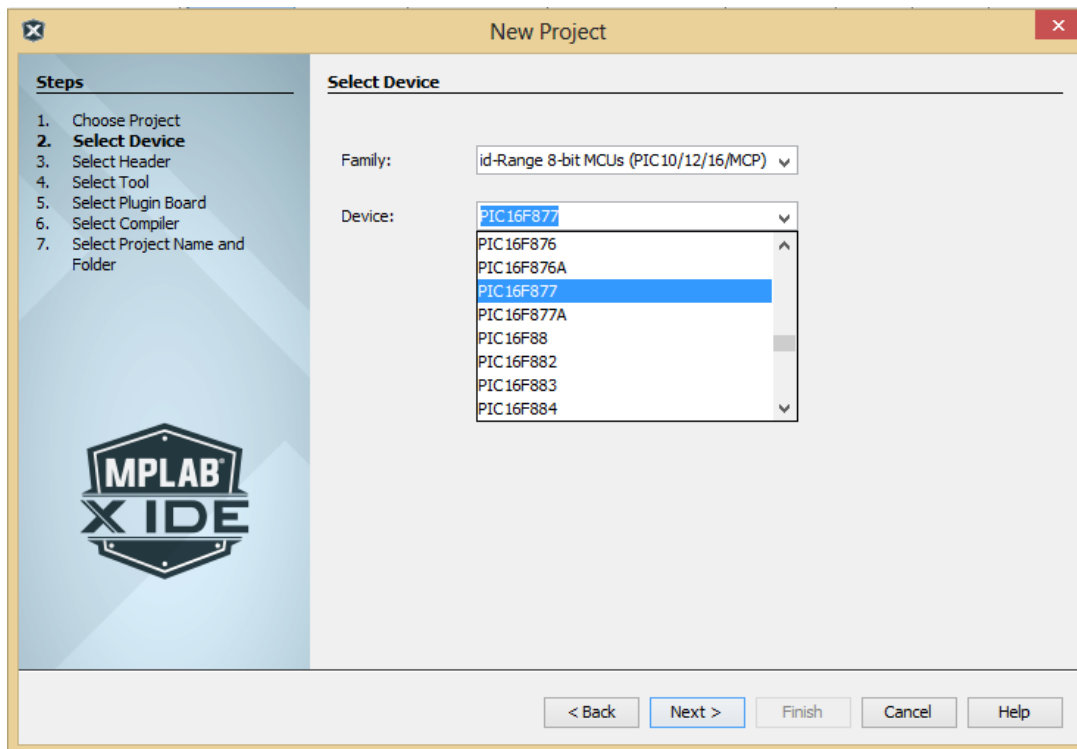
- **Nouvelle fenêtre de projet**

Sélectionnez « Microchip Embedded » (**“Microchip Embedded”**) puis « Standalone Project » (**“Standalone Project”**).



- **Sélection du processeur**

Sélectionnez «Mid-range 8-bits (PIC10/12/16/MCP)» dans le menu déroulant «Famille», puis sélectionnez «PIC16F877» dans le menu «Périphérique».

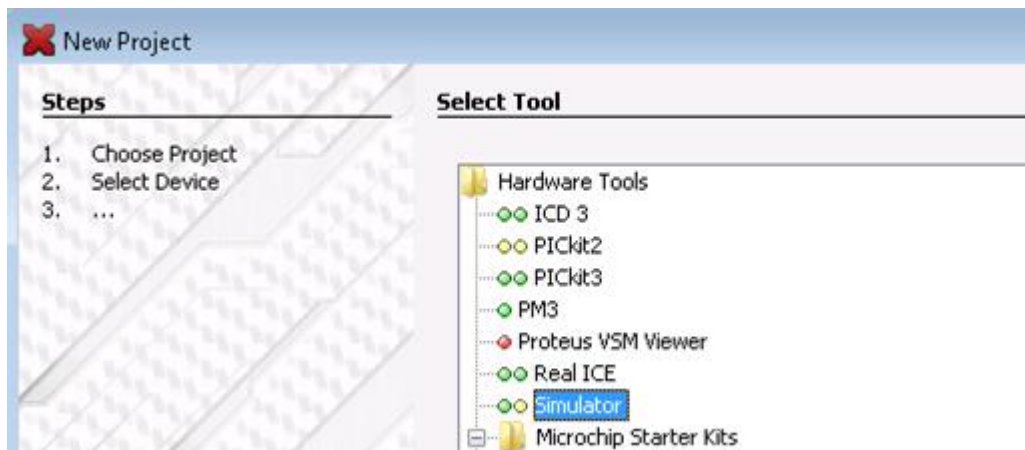



Nous utilisons le PIC16F877 comme microcontrôleur pour ce TP.

Cliquez sur Next 

- **Sélection du matériel**

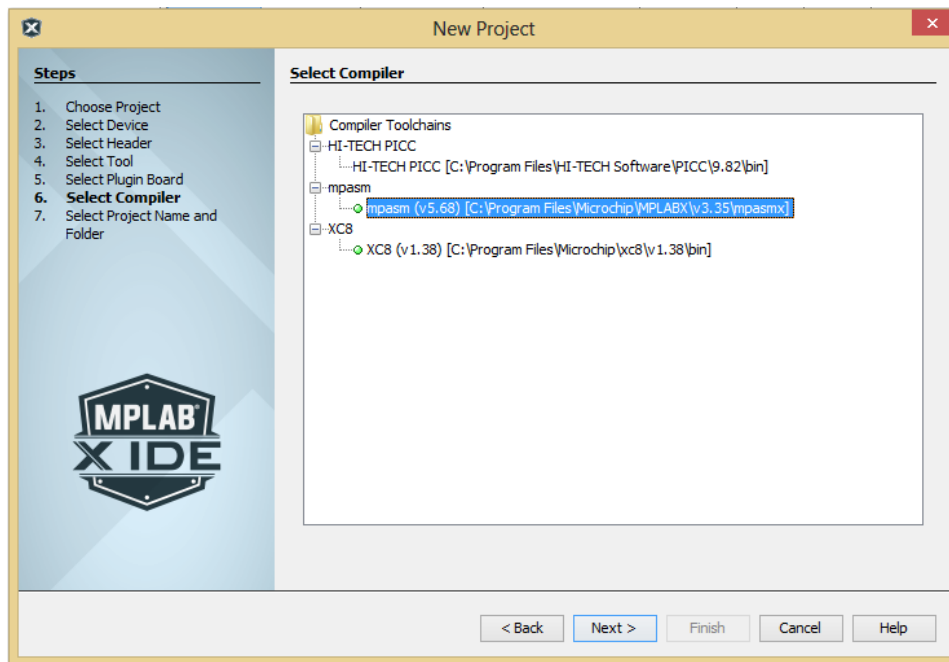
Sélectionnez "Simulateur" sous Outils matériels lorsque vous êtes invité à sélectionner un outil.



Cliquez sur Next 

- **Sélection du compilateur**

Sélectionnez une version installée du compilateur mpasm(v5.xx)



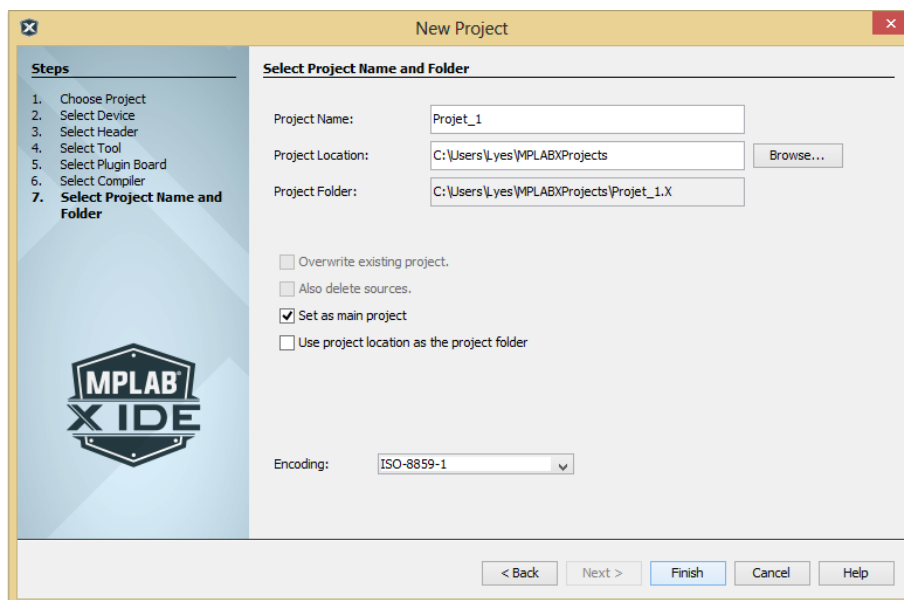
Cliquez sur Next

Next >

- **Nom du projet et sélection du dossier**

Cliquez sur le bouton Parcourir et accédez au dossier que vous désirez, ou bien laisser le dossier par défaut.

Sur la ligne Project location, taper le nom du projet (par exemple : Projet_1).



Remarque MPLAB® X remplissant la ligne du dossier de projet ;

Cliquez sur Finish

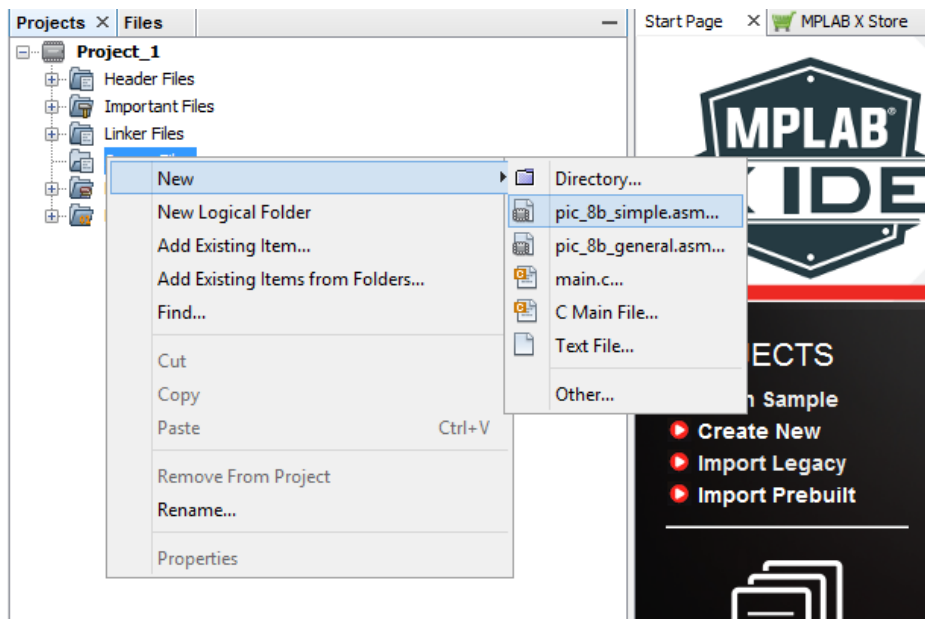
Finish

Toutes nos félicitations ! Vous venez de créer un projet MPLAB® X

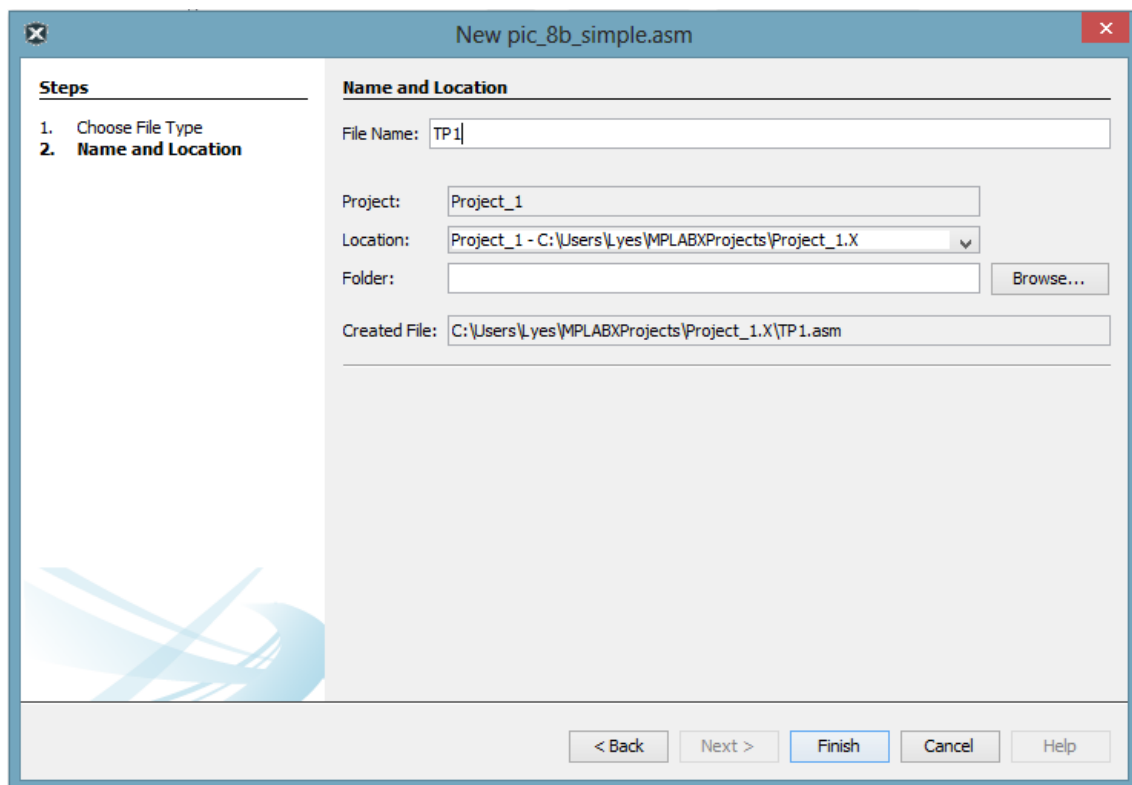
Nous allons maintenant ajouter des fichiers de code source au projet.

- **Éditer le fichier source**

Appuyez en utilisant le bouton droit sur « source files », -> New -> pic_8b_simple.asm.



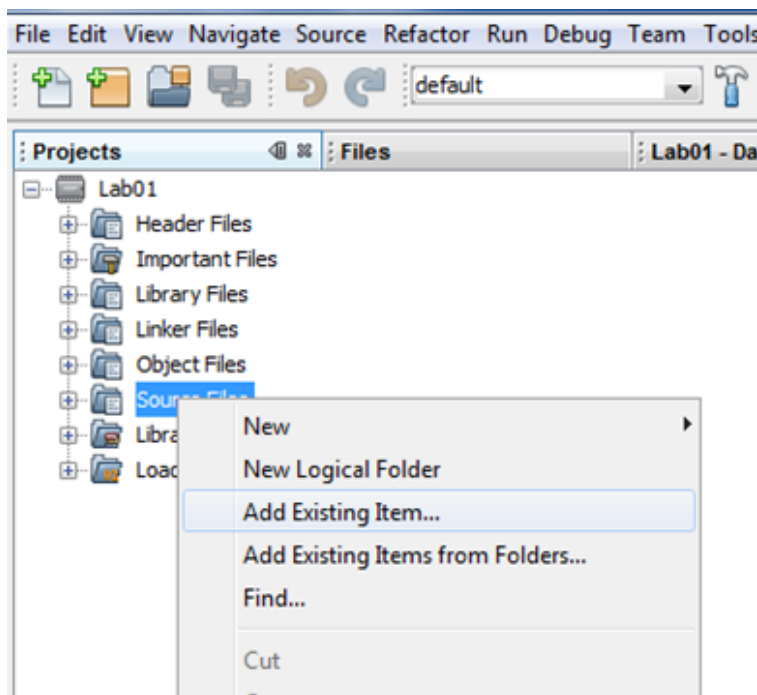
Dans la fenêtre New pic_8b_simple.asm, taper le nom de fichier « TP1 »



- **Ajout de fichiers au projet**

Cliquez avec le bouton droit sur le dossier Fichiers source dans la fenêtre du projet

Sélectionnez Ajouter un élément existant



- Ouvrir l'éditeur

Double-cliquez sur le fichier TP1.asm pour ouvrir l'éditeur. Effacer ce qui existe et taper le code suivant :

```

1      ;*****
2      ; Ce fichier est la base de départ pour une programmation avec      *
3      ; le PIC 16F877. Il contient les informations de base pour          *
4      ; démarrer.                                                         *
5      ;                                                                     *
6      ;*****
7      ;                                                                     *
8      ; NOM: TP1                                                           *
9      ; Date: 29/01/2017                                                  *
10     ; Version: 1.0                                                       *
11     ; Circuit: platine d'expérimentation                                *
12     ; Auteur: L. SAAD SAOUD                                              *
13     ;                                                                     *
14     ;*****
15     ;                                                                     *
16     ; Fichier requis: P16F877.inc                                         *
17     ;                                                                     *
18     ;*****
19     ; Exercice de manipulation de base en utilisant le microcontrôleur 16F876 *
20     ; La LED s'allume quand en appuyant sur le bouton poussoir et elle    *
21     ; s'éteint quand le bouton poussoir est relâché                     *
22     ;*****
23     LIST p=16F877                ; Définition de processeur
24     include <p16F877.inc>         ; fichier include
25     __CONFIG _CP_OFF & _DEBUG_OFF & _WRT_ENABLE_OFF & _CPD_OFF &
        _LVP_OFF & _BODEN_OFF      & _PWRTE_ON & _WDI_OFF & _HS_OSC
26     ; '__CONFIG' précise les paramètres encodés dans le processeur au moment
27     ; de la programmation du processeur. Les définitions sont dans le fichier
28     ; include. Choisir une valeur par groupe.
29     ; Voici les valeurs et leurs définitions :

```

```

30 ; Protection du programme
31 ; -----
32 ; _CP_ALL           protection totale
33 ; _CP_HALF         protection de 1000 à 1FFF
34 ; _CP_UPPER_256    protection de 1F00 à 1FFF
35 ; _CP_OFF           Pas de protection
36
37 ; Debuggage
38 ; -----
39 ; _DEBUG_ON         RB6 et RB7 utilisées pour debugger
40 ; _DEBUG_OFF        RB6 et RB7 en utilisation normale
41
42 ; Accès à la flash programme
43 ; -----
44
45 ; _WRT_ENABLE_ON     Le programme peut écrire dans la flash
46 ; _WRT_ENABLE_OFF    Le programme ne peut pas écrire dans la flash
47
48 ; Protection de la EEprom
49 ; -----
50
51 ; _CPD_ON           Mémoire EEprom protégée
52 ; _CPD_OFF          Mémoire EEprom déprotégée
53
54 ; Programmation sur circuit
55 ; -----
56
57 ; _LVP_ON           RB3 permet la programmation série de la PIC
58 ; _LVP_OFF          RB3 en utilisation normale
59
60 ; Reset de la PIC si tension <4V
61 ; -----
62 ; _BODEN_ON         Reset tension en service
63 ;                   Valide PWRTE_ON automatiquement
64 ; _BODEN_OFF        Reset tension hors service
65
66 ; Retard à la mise sous tension
67 ; -----
68
69 ; _PWRTE_OFF        Démarrage rapide
70 ; _PWRTE_ON         Démarrage temporisé
71
72 ; Watchdog
73 ; -----
74
75 ; _WDT_ON           Watchdog en service
76 ; _WDT_OFF          Watchdog hors service
77
78 ; Oscillateur
79 ; -----
80 ; _LP_OSC           Oscillateur basse vitesse (32<F<200Khz)
81 ; _XT_OSC           Oscillateur moyenne vitesse (200Khz<F<4Mhz)
82 ; _HS_OSC           Oscillateur haute vitesse (4Mhz<F<20Mhz)
83 ; _RC_OSC           Oscillateur à réseau RC
84 ; CBLOCK 0x20
85 ; Pour déclarer des variables globales
86 ; ENDC
87 ; org 0x0000 ; Adresse de reset

```

```

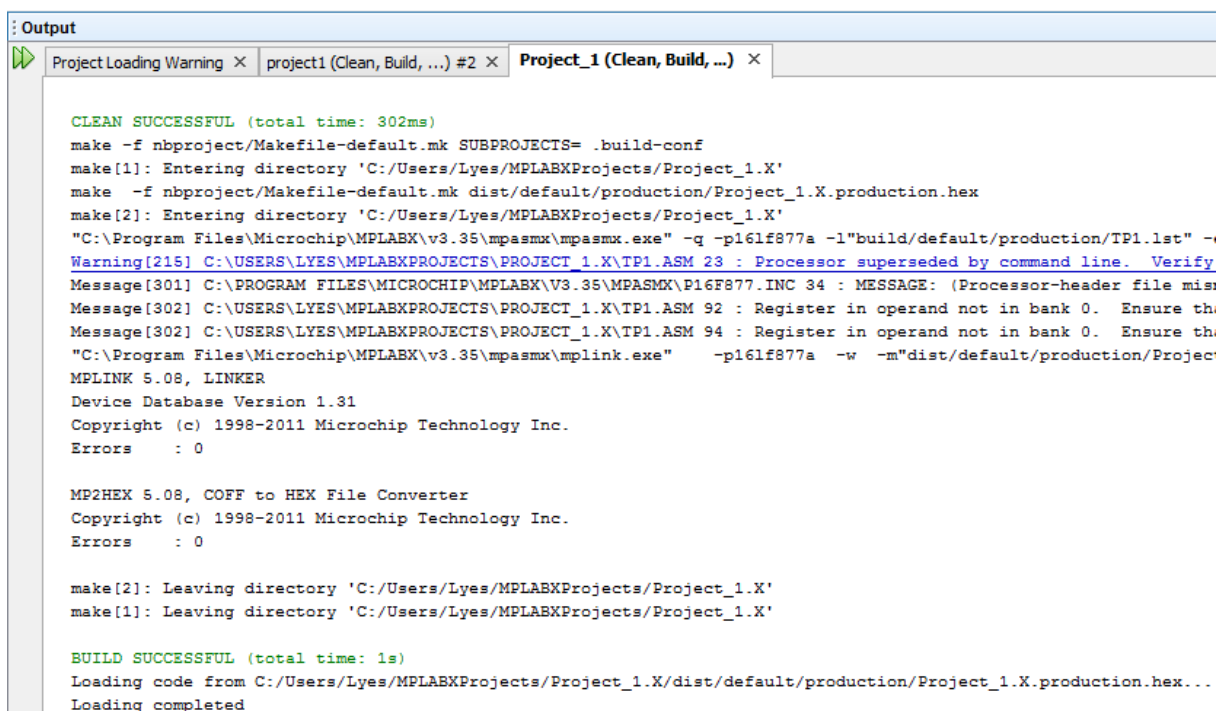
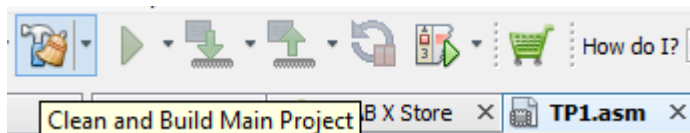
88      banksel TRISD      ; Aller à la banque 2 qui contient le register
89                        ; TRISD (les banques ont été détaillés au
90                        ; chapitre 2
91      movlw 0xFF         ; Toutes les pin PORTD sont des entrées
92      movwf TRISD        ; Charger dans le registre TRISD
93      movlw 0x00         ; Toutes les pin PORTB sont des sorties
94      movwf TRISB        ; Charger dans le registre TRISD
95      banksel PORTB      ; Aller à la banque 1 qui contient le register
96                        ; PORTB
97      clrf PORTB         ; Effacer le PORTB
98
99      Boucle1
100      btfss PORTD,0      ; Lire pin RD0
101      goto B_Off         ; Si RB0 is '0', donc aller B_Off
102
103      B_On
104      bsf PORTB,0        ; Mis à 1 RB0
105      goto Boucle1       ; Aller au début Boucle1
106
107      B_Off
108      bcf PORTB,0        ; mis à zéro RB0
109      goto Boucle1       ; Aller au début Boucle1
110
111      ;=====
112      END                ; Fin du programme

```

C'est un programme très simple conçu pour renforcer la confiance dans votre capacité à construire un projet et à l'exécuter en MPLAB® X.

• Construire le projet

Cliquez sur l'icône Nettoyer et construire pour créer le projet



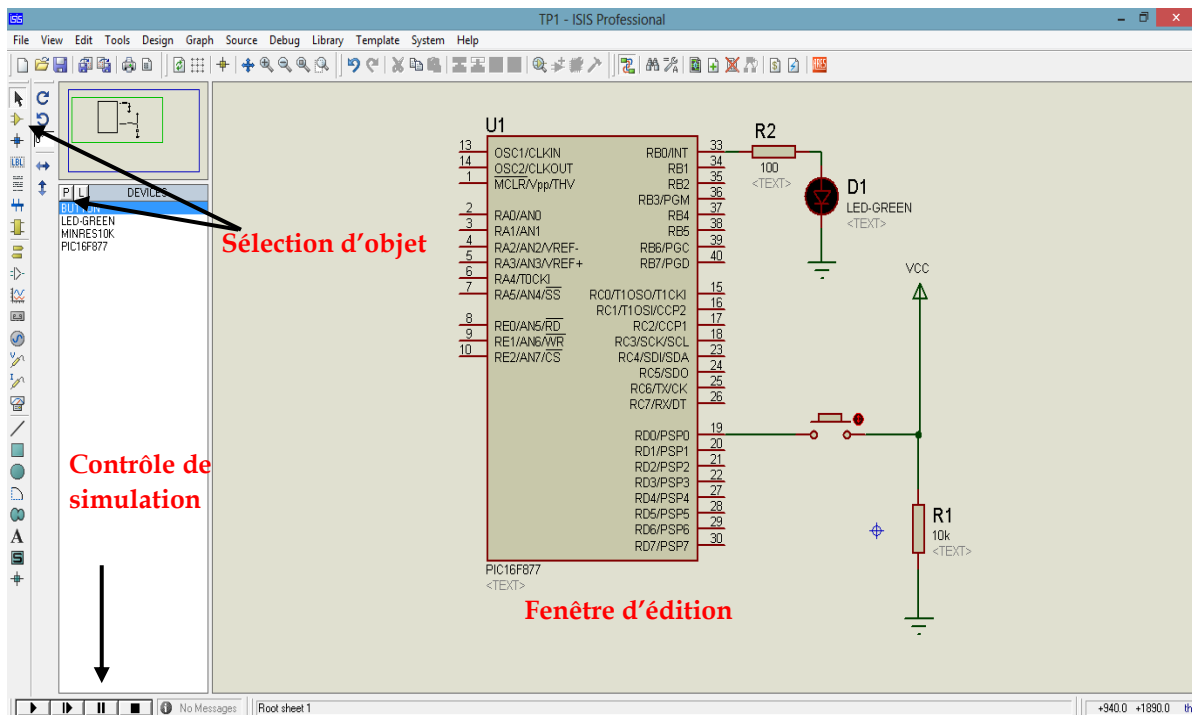
1.1.4. Réalisation du circuit électronique sur Proteus

Comme on peut le voir, ISIS permet également de présenter facilement des diagrammes de circuit à un standard professionnel.

Une capture d'écran de la capture schématique ISIS et de l'environnement de simulation interactif est illustrée à la Figure 7.1. La fenêtre d'édition du schéma principal est accompagnée d'une fenêtre d'aperçu montrant le dessin entier et une fenêtre de sélection d'objet, qui contient normalement une liste de composants. Toutefois, il affiche également des listes d'autres périphériques disponibles pour une utilisation dans la fenêtre d'édition lorsque des modes spécifiques sont sélectionnés.

La fenêtre d'édition principale comprend un contour de la feuille, qui montre le bord de la zone de dessin, dans lequel les composants doivent être placés. Le bouton Composant est normalement sélectionné par défaut dans la barre d'outils Mode. Avec ce mode sélectionné, les composants sont récupérés pour être placés sur le schéma en appuyant sur le bouton P (pick devices) et en sélectionnant la catégorie de composants requise. Le type d'appareil individuel peut alors être choisi dans une liste (Figure 7.2).

Les composants sont catégorisés comme des microprocesseurs (comprend des microcontrôleurs), des résistances, des condensateurs, etc. avec des variantes dans chacun d'eux. Les sous-catégories peuvent être sélectionnées. Les composants interactifs, tels que les boutons poussoirs sont regroupés dans la bibliothèque ACTIVE.



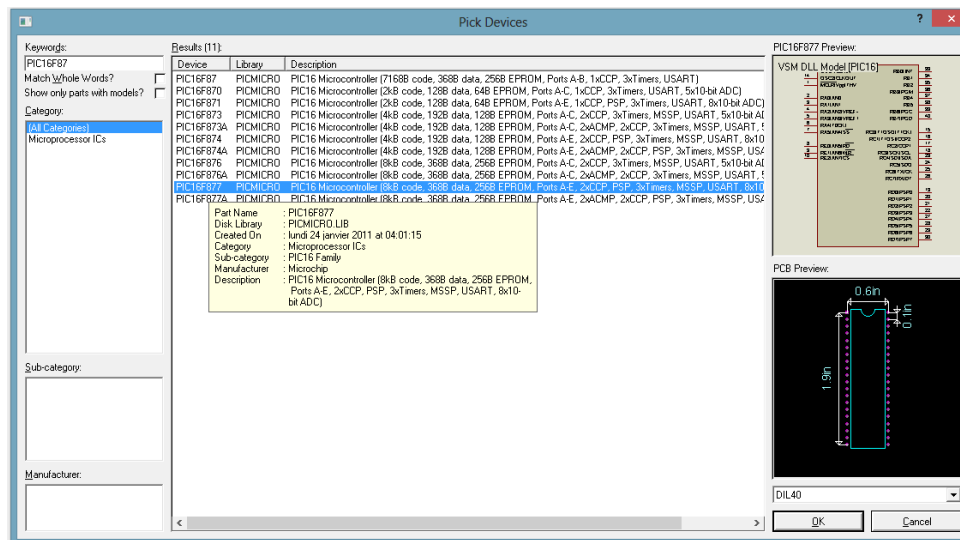


Fig. 7.2.

Un circuit pour démontrer le fonctionnement du programme TP1 est représenté à la figure 7.3.

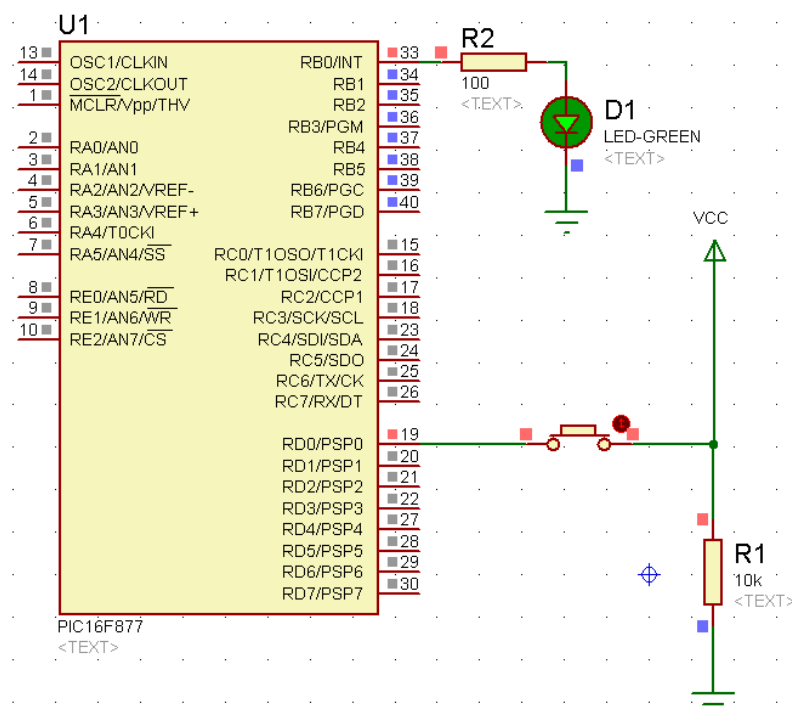


Fig. 7.3.

Le microcontrôleur est un PIC 16F877, notre dispositif de référence ; D'autres puces PIC seront décrites plus loin. Une diode électroluminescente LED est connecté au port B (RB0), avec un bouton poussoir sur RD0. Il faut noter que pour les simulations, le circuit d'horloge externe ne contrôle pas la fréquence de fonctionnement du PIC. Ceci doit être défini dans la boîte de dialogue des propriétés pour le composant MCU (voir ci-dessous). De même, l'entrée MCLR

(Master Clear) ne doit pas être connectée pour que le programme s'exécute en mode simulation, alors que cela est essentiel dans le circuit réel.

Le port n'a pas besoin d'être initialisé pour l'entrée, car c'est également la condition par défaut. D'autre part, les sorties doivent être initialisées par le programme MCU en chargeant le registre de direction de données avec des zéros. Les sorties PIC peuvent généralement fournir jusqu'à 25 mA, ce qui est suffisant pour allumer les LED sans aucun pilote supplémentaire. 100 Ω limite le courant dans la LED à environ 20 mA.

Lors de la création d'une nouvelle application, un dossier approprié doit être prévu pour contenir les fichiers de projet, car le fichier de conception du schéma sera accompagné par plusieurs fichiers associés au programme joint (code source, code hexadécimal, fichier de liste, etc.).

1. Réaliser le circuit de la figure 7.3, en utilisant le logiciel Proteus.
2. Double clic sur le PIC16F877, la fenêtre de la figure 7.4 doit s'apparaître.
3. Appuyer sur l'icône « Program file » et charger le fichier **Project_1.X.production.hex**, pour mon cas :
C:/Users/Lyes/MPLABXProjects/Project_1.X/dist/default/production/Project_1.X.production.hex
4. Appuyer sur le bouton poussoir, la LED doit s'allumer.
5. Modifier le code et le circuit pour que une autre LED en plus reliée à la pin RB1 soit éteint quand la première LED est allumée et vis-versa.
6. Modifier le code et le circuit pour que 7 autres LEDs en plus reliées au PORTB. Les LEDs seront allumées 4 par 4 en alternance (paire et impaire).

Références :

1. Lyes Saad Saoud and Irfan Hussain, "TempoNet: Empowering long-term Knee Joint Angle Prediction with Dynamic Temporal Attention in Exoskeleton Control," Accepted for presentation at the 2023 IEEE-RAS International Conference on Humanoid Robots, Austin, USA, on December 12-14, arXiv: 2310.01795, 2023.
2. Lyes Saad Saoud, Zhenwei Niu, Atif Sultan, Lakmal Seneviratne and Irfan Hussain, "ADOD: Adaptive Domain-Aware Object Detection with Residual Attention for Underwater Environments," Accepted for presentation at the 21st International Conference on Advanced Robotics (ICAR 2023), Abu Dhabi, UAE , 5 to 8 December
3. Muhayyuddin Ahmed, Ahmed Humais, Waseem Akram, Mohamed Alblooshi, Lyes Saad Saoud, Abdelrahman Alblooshi, Lakmal Seneviratne, and Irfan Hussain, "Marine X: Design and Implementation of Unmanned Surface Vessel for Vision Guided Navigation," Accepted for presentation at the 21st International Conference on Advanced Robotics (ICAR 2023), Abu Dhabi, UAE, 5 to 8 December 2023.
4. Lyes Saad Saoud, Lakmal Seneviratne, Irfan Hussain, MARS: Multi-Scale Adaptive Robotics Vision for Underwater Object Detection and Domain Generalization, arXiv:2312.15275, 2023
5. Zhenwei Niu and Lyes Saad Saoud and Irfan Hussain, "Collision Detection for Robots with Variable Stiffness Actuation by Using MAD-CNN: Modularized-Attention-Dilated Convolutional Neural Network, " arXiv: 2310.02573, 2023.
6. Lyes Saad Saoud, Humaid Ibrahim, Ahmad Aljarah, and Irfan Hussain, "Improving Knee Joint Angle Prediction through Dynamic Contextual Focus and Gated Linear Units," arXiv:2306.06900, 2023.
7. Lyes Saad Saoud and Irfan Hussain, "TempoNet: Empowering long-term Knee Joint Angle Prediction with Dynamic Temporal Attention in Exoskeleton Control," 2023 IEEE-RAS International Conference on Humanoid Robots, Austin, USA, on December 12-14, arXiv: 2310.01795, 2023.
8. Lyes Saad Saoud, Zhenwei Niu, Atif Sultan, Lakmal Seneviratne and Irfan Hussain, "ADOD: Adaptive Domain-Aware Object Detection with Residual Attention for Underwater Environments," Accepted for presentation at the 21st International Conference on Advanced Robotics (ICAR 2023), Abu Dhabi, UAE , 5 to 8 December 2023, arXiv:2312.06801.
9. Muhayyuddin Ahmed, Ahmed Humais, Waseem Akram, Mohamed Alblooshi, Lyes Saad Saoud, Abdelrahman Alblooshi, Lakmal Seneviratne, and Irfan Hussain, "Marine X: Design and Implementation of Unmanned Surface Vessel for Vision Guided Navigation," 21st International Conference on Advanced Robotics (ICAR 2023), Abu Dhabi, UAE, 5 to 8 December 2023, arXiv:2311.17197.
10. Ahsan Baidar Bakht, Zikai Jia, Muhayy ud Din, Waseem Akram, Lyes Saad Soud, Lakmal Seneviratne, Defu Lin, Shaoming He, Irfan Hussain, MuLA-GAN: Multi-Level Attention GAN for Enhanced Underwater Visibility, arXiv:2312.15633 , 2023
11. Waseem Akram, Muhayyuddin Ahmed, Lyes Saad Saoud, Lakmal Seneviratne, Irfan Hussain, Autonomous Underwater Robotic System for Aquaculture Applications, arXiv:2308.14762, 2023

12. Hasan AlMarzouqi, Lyes Saad Saoud, Semantic Labeling of High Resolution Images Using EfficientUNets and Transformers, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1-13, 2023.
13. Lyes Saad Saoud, Hasan AlMarzouqi, Ramy Hussein, Cascaded Deep Hybrid Models for Multistep Household Energy Consumption Forecasting, arXiv:2207.02589, 2022.
14. L. Saad. Saoud and H. AlMarzouqi, "Forecasting Solar Irradiance using Hybrid Stationary Wavelet Transform- Quaternion Valued Neural Network with a Softplus AMSGrad Learning Algorithm," 2022 IEEE International Conference on Power Systems Technology (POWERCON), 2022, pp. 1-6, doi: 10.1109/POWERCON53406.2022.9929612.
15. Lyes Saad Saoud, Hasan AlMarzouqi, Ramy Hussein, "Household Energy Consumption Prediction Using the Stationary Wavelet Transform and Transformers," in *IEEE Access*, vol. 10, pp. 5171-5183, 2022, doi: 10.1109/ACCESS.2022.3140818.
16. Lyes Saad Saoud, Hasan AlMarzouqi, Forecasting Solar Irradiance Using Hybrid Stationary Wavelet Transform- Quaternion Valued Neural Network with a Softplus AMSGrad Learning Algorithm, 2022 IEEE International Conference on Power Systems Technology (POWERCON), Pullman, Kuala Lumpur, MALAYSIA , 12-14 September 2022.
17. Shadab Shishegar, Reza Ghorbani, Lyes Saad Saoud, Sophie Duchesne & Geneviève Pelletier (2021) Rainfall–runoff modelling using octonion-valued neural networks, *Hydrological Sciences Journal*, 66:13, 1857-1865, DOI: 10.1080/02626667.2021.1962885
18. Lyes Saad Saoud, Hasan Al-Marzouqi, Mohammed Deriche, Wind speed forecasting using the stationary wavelet transform and quaternion adaptive gradient methods, *IEEE Access* 9, 127356-127367, 2021.
19. K. Aimeur, L. Saad Saoud, R. Ghorbani, Short-Term Solar Irradiance Forecasting and Photovoltaic System Management Using Octonion Neural Networks, *Applied Solar Energy* 56 (3), 219-226, 2020.
20. Lyes Saad Saoud and Hasan Al-Marzouqi, Metacognitive Sedenion-Valued Neural Network and Its Learning Algorithm, *IEEE access*, vol. 8, pp. 144823-144838, 2020
21. L. Saad Saoud, R. Ghorbani, "Metacognitive Octonion-Valued Neural Networks as They Relate to Time Series Analysis," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 2, pp. 539-548, Feb. 2020, doi: 10.1109/TNNLS.2019.2905643.
22. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari "A novel method to forecast 24 hours of global solar irradiation", *Energy systems*, Volume 9, Issue 1, pp 171–193, 2018, Springer-Verlag, Berlin Heidelberg.
23. L. Saad Saoud, R. Ghorbani, F. Rahmoune, "Cognitive Quaternion Valued Neural Network and some applications", *Neurocomputing*, Volume 221, Pages 85–93, 2017, Elsevier.
24. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari "Fully Complex Valued Wavelet Network for Forecasting the Global Solar Irradiation", *Neural Processing Letters*, Volume 45, Issue 2, pp 475–505, 2017, Springer-Verlag, Berlin Heidelberg.

25. E. Reihani, M. Motalleb, R. Ghorbani, L. Saad Saoud, "Load peak shaving and power smoothing of a distribution grid with high renewable energy penetration", *Renewable Energy*, 86, 1372-1379, 2016.
26. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "Generalized dynamical fuzzy model for identification and prediction ", *Journal of Intelligent and Fuzzy Systems*, 26, 1771–1785, 2014.
27. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "Complex-valued forecasting of global solar irradiation", *Journal of Renewable and Sustainable Energy*, 5(4), pp. 043124-043145, 2013.
28. L. Saad Saoud, et al., "A Neural Network Based on an Inexpensive Eight Bit Microcontroller" *Neural computing and application*, vol. 20, no3, pp. 329-334 Springer-Verlag London, 2011.
29. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "Prediction of the daily global solar Irradiation of the great Maghreb region using the complex-valued neural networks", *Revue des Energies Renouvelables*, Vol. 17 N°1, pp. 173 – 185, 2014.
30. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "Modeling pH Neutralization Process using Fuzzy Dynamic Neural units Approaches." *International Journal of Computer Applications* 28(4):22-29, 2011. Foundation of Computer Science, New York, USA.
31. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "Wavelet Network Implementation on an Inexpensive Eight Bit Microcontroller," in: *Features of Liquid Crystal Display Materials and Processes*, Natalia V. Kamanina (Ed.), 87-102, Intech publisher, 2011.
32. L. Saad Saoud, F. Rahmoune, V. Tourtchine, K. Baddari, "Complex-valued wavelet neural network prediction of the daily global solar irradiation of the Great Maghreb Region", in *Progress in Clean Energy Analysis and Modeling*, I. Dincer et al. Ed. Springer Berlin Heidelberg, 2015, pp 321-339.
33. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "Short term forecasting of the global solar irradiation using the fuzzy modeling technique: case study of Tamanrasset city, Algeria", in *Progress in Clean Energy Analysis and Modeling*, I. Dincer et al. Ed. Springer Berlin Heidelberg, 2015, pp. 281-289.
34. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "Photovoltaic Maximum Power Modeling using Radial Basis Function Network Based on the PIC Microcontroller", in: *Artificial Intelligence and Hybrid Systems*, iConcept Press, 2013.
35. L. Saad. Saoud and H. AlMarzouqi, "Forecasting Solar Irradiance using Hybrid Stationary Wavelet Transform- Quaternion Valued Neural Network with a Softplus AMSGrad Learning Algorithm," 2022 IEEE International Conference on Power Systems Technology (POWERCON), 2022, pp. 1-6, doi: 10.1109/POWERCON53406.2022.9929612.
36. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "On the generalized dynamical fuzzy model for forecasting complex systems", *Global Conference on Applied Computing in Science and Engineering*, Rome (Italy), 27-29 July 2016.
37. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "Quaternion neural network to forecast the daily solar irradiation", 3ed International Symposium on Innovative Technologies in Engineering and Science, 3-5 June 2015, Valencia –Spain.

38. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "Quaternion neural network to forecast the daily solar irradiation", 3ed International Symposium on Innovative Technologies in Engineering and Science, 3-5 June 2015, Valencia –Spain
39. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "On the Fuzzy Modeling Based Estimation of Maximum Power Generation from Photovoltaic Module", IEEE proceeding of 2013 3rd International Conference on Systems and Control (ICSC) , 29-31 Oct. 2013, pp. 6 – 11, Algiers, Algeria.
40. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "Design of a Dynamic Neural Network with Kalman Filter for the Identification of Nonlinear Systems. Application: prediction of the maximum power generated by a Photovoltaic module", 1ST International Symposium on Innovative Technologies in Engineering and Science, 07 - 09 June 2013, Sakarya, Turkey.
41. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "Identification of Bioprocesses Using Random Search and Simulated Annealing Algorithms", 2012 6th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), Sousse, Tunisia.
42. L. Saad Saoud, F. Rahmoune, V. Tourtchine and K. Baddari, "An Inexpensive Embedded Electronic Continuous Stirred Tank Reactor (CSTR) Based on Neural Networks," IEEE proceedings of International Conference on Multimedia Technology (ICMT), 26-28 July 2011, pp. 6233 - 6237, Hang Zhou, China, 2011.