جامعة محمد خيضر بسكرة
كلية العلوم والتكنولوجيا
قسم الهندسة الكهربائية

مذكرة ماستر

علوم وتكنولوجيا

اِلكترونيك

الأنظمة المضمنة

رقم : .....

من إعداد و مناقشة الطالب :

غربي أكرم عبد الرؤوف

يوم :2019/07/06

دراسة وإنجاز لوحة تطوير اِلكترونية لتطبيقات المتحكم PIC18F4550.

لجنة المناقشة:

بن المير عقبة	أ. مح ب	جامعة محمد خيضر بسكرة	رئيسا
رحماني نصر الدين	أ. مس أ	جامعة محمد خيضر بسكرة	مشرفا ومقررا
هزابرة عادل	أ. مس أ	جامعة محمد خيضر بسكرة	مناقشا

السنة الجامعية: 2019/2018

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي والبحث العلمي

جامعة محمد خيضر بسكرة

كلية العلوم والتكنولوجيا

قسم الهندسة الكهربائية

اِلكترونيك

الأنظمة المضمنة

مذكرة تخرج لنيل شهادة:

ماستر

الموضوع

دراسة وإنجاز لوحة تطوير اِلكترونية لتطبيقات المتحكم PIC18F4550.

إعداد الطالب:

غربي أكرم عبد الرؤوف.

موافقة الأستاذ المشرف:

أ. مس أ. رحماني نصر الدين.

موافقة رئيس لجنة المناقشة

أ. مح ب. بن المير عقبة.

التوقيع و الختم

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي و البحث العلمي

جامعة محمد خيضر بسكرة
كلية العلوم والتكنولوجيا
قسم الهندسة الكهربائية
فرع: الكترونيك
تخصص الأنظمة المضمنة

الموضوع:

دراسة وإنجاز لوحة تطوير اِلكترونية لتطبيقات المتحكم PIC18F4550.

مقترح من طرف أ. مس أ: رحماني نصر الدين.

إشراف أ. مس أ: رحماني نصر الدين.

ملخص:

يعتبر المتحكم الدقيق من أهم العناصر في الأنظمة المضمنة، فهو المسؤول عن المعالجة والتحكم في كل وظائف النظام، ومع انتشار اللوحات التطويرية لمختلف المتحكمات أصبح العمل على مشاريع الأنظمة الإلكترونية أكثر سهولة وبساطة.

الهدف من مشروعنا هو إنجاز لوحة تطويرية للمتحكم PIC18F4550 باعتبار النوع PIC أكثر الأنواع انتشارا بين طلبة وأساتذة وهواة الالكترونيك؛ هذه اللوحة التطويرية متوافقة مع الملف البرمجي المكتوب بلغة MikroC وتعتمد على إحدى طرق البرمجة المشهورة المتمثلة في البرمجة بجهدٍ عالٍ والبرمجة بجهدٍ منخفض وتقنية برنامج التحميل.

بِسْمِ اللّٰهِ الرَّحْمَنِ الرَّحِيمِ

الإهداء:

إلى أمي الغالية وأبي العزيز.

إلى إخوتي وأصدقائي.

إلى كل من علمني ولو حرفا.

إلى الأمة الإسلامية.

أهدي عملي المتواضع هذا.

أكرم عبد الرؤوف غربي.

تشكر

أتقدم بالشكر الكبير للأستاذ نصر الدين رحماني على قبوله الإشراف على مذكرتي، وعلى مجهوده الكبير في توجيهي ونصحي وإرشادي، كما أشكر صديقي وزميلي وليد علوي الطالب في كلية الأدب العربي على مجهوده في مراجعة وتصحيح المذكرة لغويا ونحويا، وأتقدم بخالص الشكر للجنة المناقشة لقبولهم مناقشة مذكرتي، كما لا أنسى شكر كل من قدم لي الدعم خاصة والداي العزيزين وأصدقائي.

أكرم عبد الرؤوف غربي.

ملخص:

يعتبر المتحكم الدقيق من أهم العناصر في الأنظمة المضمنة، فهو المسؤول عن المعالجة والتحكم في كل وظائف النظام، ومع انتشار اللوحات التطويرية لمختلف المتحكمات أصبح العمل على مشاريع الأنظمة الإلكترونية أكثر سهولة وبساطة.

الهدف من مشروعنا هو إنجاز لوحة تطويرية للمتحكم PIC18F4550 باعتبار النوع PIC أكثر الأنواع انتشارا بين طلبة وأساتذة وهواة الالكترونيك؛ هذه اللوحة التطويرية متوافقة مع الملف البرمجي المكتوب بلغة MikroC وتعتمد على إحدى طرق البرمجة المشهورة المتمثلة في البرمجة بجهدٍ عالٍ والبرمجة بجهدٍ منخفض وتقنية برنامج التحميل.

The microcontroller is one of the most important elements in embedded systems. It is responsible for handling and controlling all functions of the system. Now the work on the electronic systems projects become easier and simpler; that is because commonness of the development boards of the various controllers.

The goal of our project is to completion a PIC18F4550 development board; that is for the PIC is the most popular type among students, professors and electronic hobbyists; this development board is compatible with the MikroC Hex code and is based on one of the famous programming methods witch is High Voltage ICSP or Low Voltage ICSP or Bootloader.

قائمة الجداول حسب الفصل:

الفصل الثاني:

جدول II-1- جدول لبعض أنواع المتحكمات.....27

جدول II-2- جدول مقارنة لبعض متحكمات العائلة PIC18F.....29

الجدول II-3- قيم المكثفات عند استعمال هزاز كريستال.....38

الفصل الثالث:

الجدولIII-1-جدول للبيانات المحمولة على الواصف.....74

الجدولIII-2- جدول لبعض أصناف الأجهزة.....74

الفصل الرابع:

الجدول IV-1- جدول يوضح وظائف الأطراف الخاصة ببرمجة المتحكم 78

الجدول IV-2- سجلات مؤشر عنوان الذاكرة.....80

الجدولIV-3- أوامر التحكم الخاصة بالبرمجة.....83

الجدولIV-4- خوارزمية عملية المسح الكلي للذاكرة.....84

الجدولIV-5- خوارزمية عملية كتابة البرنامج على الذاكرة.....86

الجدولIV-6- خوارزمية برمجة الإعدادات.....87

الجدولIV-7- خوارزمية عملية قراءة خانة واحدة من الذاكرة.....88

قائمة الأشكال حسب الفصل:

الفصل الأول:

الشكلI-1-مراحل عملية ترجمة البرنامج.....5

الشكلI-2-لوحة أردوينو UNO مع توضيح لمكوناتها.....13

الشكلI-3-لوحة LaunchPad مع توضيح لمكوناتها.....14

الشكلI-4-لوحة Raspberry Pi مع توضيح لمكوناتها.....16

الشكلI-5-اللوحة التطويريةuC32 18

الشكلI-6-اللوحة التطويرية Ready for PIC مع توضيح لمكوناتها.....19

الفصل الثاني:

الشكلII-1- مخطط مبسط للبنية الداخلية للمتحكم الدقيق.....25

الشكلII-2- أرجل Pins المتحكم PIC18F4550.....30

الشكلII-3- البنية الداخلية للمتحكمPIC18F4550.....33

الشكل II-4- تقسيم ذاكرة البيانات SRAM.....34

الشكل II-5- عملية الكتابة على ذاكرة البرنامج.....36

الشكل II-6- توصيل هزاز الكريستال أو السيراميك بالمتحكم.....38

الشكل II-7- مدخل الساعة في النمطين EC و ECPLL.....39

الشكل II-8- مدخل الساعة في النمطين ECIO و ECPIO.....39

الشكل II-9- مخطط عمل دارة PLL.....40

الشكل II-10- مخطط عمل الهزاز الداخلي.....42

الشكلII-11- مخطط عملي للمحول ADC.....50

الشكلII-12- دورة التحويل للمحولADC.....52

الشكلII-13- وحدة الاتصال USB.....53

الشكلII-14- الذاكرة RAMUSB.....54

الشكلII-15- مقاطعات وحدة USB.....54

الفصل الثالث:

الشكلIII-1-النقل التسلسلي المتزامن للبيانات.....57

الشكل III-2-النقل التسلسلي الغير متزامن للبيانات.....58

الشكلIII-3-مستويات الجهد في البروتوكول RS-232.....59

الشكلIII-4-المُوَصِّل DB25 للبروتوكول RS-232.....59

الشكلIII-5-المُوَصِّل DB9 للبروتوكول RS-232.....60

الشكل III-6-الإتصال أحادي الإتجاه Simplex.....60

الشكل III-7-مخطط إنتقال البيانات للإتصال Simplex.....61

الشكلIII-8-الإتصال النصف ثنائي Half duplex.....61

الشكلIII-9- مخطط إنتقال البيانات للإتصال Half duplex.....61

الشكلIII-10-الإتصال الثنائي الكامل Full duplex.....62

الشكلIII-11- مخطط اِنتقال البيانات للاِتصال Full duplex.....62

الشكلIII-12- مخطط بياني للبروتوكول RS-232.....62

الشكلIII-13- مخطط بياني لتقنية المصافحة Handshaking.....64

الشكلIII-14- الدارة المدمجة MAX232 بين الحاسوب و المتحكم.....65

الشكل III-15-الأسلاك داخل خيط الناقل USB.....67

الشكلIII-16- تشفير NRZI.....68

الشكلIII-17-إضافة بت '0' بعد 6 قيم متتالية للواحد.....68

الشكلIII-18-التصميم الهندسي لخط الناقل USB.....69

الشكل III-19-بنية إطار البيانات.....70

الشكلIII-20- موصلان من النوع A و B.....70

الشكلIII-21- موصلان من النوعMini B و Micro B.....70

الشكلIII-22- تحديد سرعة الناقل USB.....71

الشكلIII-23- رسالة التعرف على جهاز جديد متصل بالمضيف.....72

الفصل الرابع:

الشكل IV-1- توضيح للأطراف المستعملة في برمجة متحكمات العائلة PIC18F4xxx 78

الشكل IV-2- المساحة المخصصة للملف البرمجي من ذاكرة البرنامج.....79

الشكل IV-3- أجزاء ذاكرة البرنامج Flash Memory.....80

الشكلIV-4- الدخول إلى وضع البرمجة عالية الجهد 81

الشكلIV-5- الخروج من وضع البرمجة عالية الجهد 81

الشكلIV-6- الدخول إلى وضع البرمجة بالجهد المنخفض 82

الشكلIV-7- الخروج من وضع البرمجة بالجهد المنخفض 82

الشكلIV-8- تمثيل بياني لإرسال أمر معين عبر PGD بالتزامن مع نبضات PGC.....84

الشكلIV-9- مخطط لمراحل عملية المسح الكلي للذاكرة.....85

الشكلIV-10- مخطط لمراحل عملية كتابة البرنامج على الذاكرة.....86

الشكلIV-11- مخطط لمراحل عملية برمجة الإعدادات.....87

الشكلIV-13- مخطط لمراحل عمل تقنية برنامج التحميل.....89

الشكلIV-14-تقسيم ذاكرة البرنامج بين برنامج التحميل و التطبيق البرمجي البرمجي.....90

الشكلIV-15- نافذة إدارة المشروع ضمن بيئة تطوير MikroC.....91

الشكلIV-16-واجهة الأداة USB mikroBootloader.....92

الشكلIV-17-نموذج لدارة اللوحة التطويرية الأولى على Proteus.....93

الشكلIV-18-مخطط للدارة المطبوعة الخاصة باللوحة التطويرية على برنامج Proteus.....94

الشكل IV-19-صورة للوحة التطويرية الأولى.....94

الشكلIV-20-تحديد المتحكم الدقيق من طرف برنامج mikroBootloader.....95

الشكلIV-21-تحديد المتحكم الدقيق من طرف برنامج mikroBootloadr.....95

الشكلIV-22-البحث عن الملف البرمجي باستعراض المجلدات.....96

الشكلIV-23-بدأ التحميل.....96

الشكلIV-24-نهاية عملية التحميل وتصفير المتحكم وإعادة تشغيله.....96

الشكلIV-25-دارة المبرمجة JDM.....98

الشكلIV-26-عزل أطراف المبرمجة JDM بواسطة مفتاح.....99

الشكلIV-27-نموذج للوحة التطويرية الثانية بواسطة برنامج Proteus.....100

الشكلIV- 28-صورة للوحة التطويرية الثانية.....100

الشكلIV -29-واجهة برنامج WinPic800. 101

الشكل IV-30- فتح نافذة إعدادات.....101

الشكلIV-31-نختار المبرمجة JDM من قائمة الأجهزة.....101

الشكلIV-32-نختار المنفذ COM Port الموصول بالمبرمجة.....102

الشكلIV-33- أداة تفقد الجهاز من شريط الأدوات.....102

الشكلIV-34- أداة استعراض المجلدات.....102

الشكلIV-35- عرض محتويات الملف البرمجي.....103

الشكلIV-36-مسح ذاكرة المتحكم.....103

الشكلIV-37- برمجة المتحكم.....103

الشكلIV-38- نموذج للوحة التطويرية الثالثة على Proteus.....105

الشكلIV-39- عزل أطراف البرمجة بواسطة مفتاح.....105

الشكل IV-40-صورة للوحة التطويرية الثالثة.....105

الشكلIV-41- مخطط مراحل عملية البرمجة.....106

الشكلIV-42- مخطط عملية تحديد المتحكم.....106

الشكلIV-43- مخطط لعمل التطبيق البرمجي.....108

الشكل IV-44-واجهة التطبيق البرمجي.....109

الشكل IV-45-قراءة Winpic800 للملف البرمجي.....109

الشكل IV-46-قراءة Notpad للملف البرمجي.....109

الشكلIV-47- تحليل أحد أسطر الملف البرمجي.....110

الشكلIV-48- تحليل أحد سطر الإعدادات.....110

الشكلIV-49- سطر بيانات الإعدادات.....110

قائمة الاختصارات مرتبة أبجديا:

ADC: Analog-To-Digital Converter

BOR: Brown-out Reset.

CCS: Code Composer Studio

CTS: Clear To Send.

DCE: Data Communication Equipment

DMA: Direct Memory Access

DRAM: Dynamic RAM

DSR: Data Set Ready

DTE :Data Terminal Equipment

DTR: Data Terminal Ready.

EEPROM: electrically erasable programmable read-only memory

GPIO : General Purpose Input Output

GPRs: General Purpose Registers

GUI: Graphical User Interface

Hex: Hexadecimal

HID: Human Interface Device

I2C: Inter-Integrated Circuit

ICSP: In Circuit Serial Programming

IDE: Integrated Development Environment

JVM: Java Virtual Machine

Lsb: Least Significant Bit

LVP: Low Voltage Programming

Msb: Most Significant Bit

NRZI: Non-Return-To-Zero-Inverted

PCB: Printed Circuit Boards

PIC : Programmable Interrupt Controller

PID: Product ID

PLL : Phase Locked Loop

POR: Power-On Reset

PWM: Pulse-Width Modulation

RS-232: Recommended Standard

RTS: Ready To Send.

SFRs: Special Function Registers

SIE: Serial Interface engine

SOF: Start Of Frame

SPI: Serial Peripheral Interface

SPP: Streaming Parallel Port

SRAM: Static Random-Access Memory

USART: Universal Synchronous/Asynchronous Receiver/Transmitter

USB: Universal Serial Bus

VID : Vendor ID

WDT: Watchdog Timer

قائمة المصطلحات المترجمة مرتبة أبجديا:

الإتصال أحادي الإتجاه Simplex

الإتصال التسلسلي Serial Communication

الإتصال الثنائي الكامل Full Duplex

الإتصال المتوازي Parallel Communication

الإتصال النصف ثنائي Half Duplex

أجهزة مخاطبة الإنسان Human Interface Device

الأدوات Toolboxes

إشارة تعرف Acknowledge

أصناف الأجهزة Device Classes

إطار البيانات Data Frame

أطراف التحكم العامة GPIO

الآلات Machine

آلة جافا الافتراضية Java Virtual Machine

الأمر Instruction

أنظمة الإتصالات Communication Systems

أنظمة التحكم Control Systems

الأنظمة المضمنة Embadded Systems

أنظمة معالجة الإشارة Signal Processing Systems

أولوية عالية High Priority

أولوية منخفضة Low Priority

بت للبداية Start bit

بت للتكافؤ Parity bit

بت للنهاية Stop bit

بتات الإعدادات Configuration Bits

بداية الاطار Start Of Frame

البرمجة الهيكلية Structured Language

البرمجة بجهد منخفضLow Voltage ICSP

البرمجة داخل الدارة بجهد عالي High Voltage ICSP

برمجيات Software

برنامج التطبيق Application Program

برنامج تحميل Bootloader

بمحرك واجهة الإتصال التسلسلي Serial Interface Engine

بنية الحاسب Computer Architecture

البوابة (المنفذ) Port

بيئة التطويرية Integrated Development Environment Ide

بيئة تطوير واجهة المستخدم التخطيطية Graphical User Interface Environment

تجديد Refreshing

التجهيز Device

التصميم الهندسي Topology

التطبيق البرمجي Application

تعديل عرض النبضة Pulse-Width Modulation

التقريب المتتالي Successive Approximation

تقنية المصافحة Handshaking

التكافؤ الزوجي The Even Parity

التكافؤ الفردي The Odd Parity

التوازي Parallel

جهاز بيانات الإتصال Data Communication Equipment

جهاز نهاية البيانات Data Terminal Equipment

جهد التغذية Power Supply

الحافة الصاعدة لنبضة الساعة The Rising Edge Of Clock

الحافة النازلة لنبضة الساعة The Falling Edge Of Clock

خط البيانات Data Bus

الخطية Lineare

دارات مطبوعة Printed Circuit Boards

دفتر المواصفات Data Sheet

ذاكرات Memories

ذاكرة الإستقبالReceiver Buffer

ذاكرة البرنامج Flash Memory

ذاكرة البرنامج Program Memory

ذاكرة القراءة القابلة للمسح كهربائيا Electrically Erasable Programmable Read-Only Memory

ذاكرة الوصول العشوائي Random-Access Memory

الرقمية Digital

ركب وشغل Plug & Play

رمز Character

زمن التحصيل Acquisition Time

زمن التحويل Conversion Time

زمن تأخير Time Delay

الساعة الخارجية External Clock

سجلات Registers

سجلات الأغراض عامة General Purpose Registers

سجلات الوظائف الخاصة Special Function Registers

السرعة العالية High Speed

السرعة الفائقة Super Speed

السرعة الكاملة Full Speed

السرعة المنخفضة Low Speed

سرعة تدفق محددة Baud Rate

عالية المستوى High Level

عبر قناة اِفتراضية Default Pipe

عتاد إلكتروني Hardware

العداد Counter

العلامة Label

الغلاف السفلي Inner Shield

الغلاف العلوي Outer Shield

الغير الخطية Nonlinear

القلاب Flip-Flop

قناة الاتصال Pipe

لغة الآلة Machine Language

لغة التجميع Assembly Language

لغة منخفضة المستوى Low Level

لوحات التطوير الالكترونية Development Boards

لوحة تجريبية Bread Board

المترجم Compiler

المجمع Hub

المحاكاة Simulation

المحاكي Simulator

محول التماثلي الرقمي Analog-To-Digital Converter

مداخل تماثلية Analog Input

مدخل جهد إعادة التعيين Reset Input

مدخل جهد البرمجة Vpp

مدخل جهد التغذيةVdd

المرجع الأرضي Vss / Gnd

المستوى المنخفض Low Level

المصافحة البرمجية Software Handshaking

المضيف Host

معالج دقيق Microprocessor

المعامل Operand

معامل القسمة Prescaler

معلومات التعريف Identification Information

المفردات البرمجية Syntax

مقارن Comparator

مقاطعات Interrupts

مقاومة شدٍّ داخلية إلى الجهد V_{dd} Weak Internal Pull-Up

المكتبات Libraries

المكثفات الطفيلية Parasitic Capacitors

الملف البرمجي Hex Code

المنافذ Ports

منافذ الإدخال /الإخراج I/O Ports

مُنظم الجهد Regulator

مؤشر عنوان الذاكرة Memory Address Pointer

المُوَصِّل Connector

المؤقت Timer

مؤقت المراقبة Watchdog Timer

مؤقت بداية التشغيل Power-Up Timer

الميكروكنترولر، المتحكم الدقيق Microcontroller

الناقل التسلسلي العام Universal Serial Bus USB

ناقلة Transaction

النظام الثنائي Binary

النظام الثنائي Binary System

النظام الستا عشري Hexadecimal

نقطة الاتصال Endpoint

نقطة إلى نقطة Point To Point

النقل التفاضلي Differential Transmission

الهزاز الخارجي External Oscillator

الهزاز Oscillator

الهواتف المحمولة Mobile

واجهة خاصة بالتنقيح والبرمجة Debugging And Programming Interface

الواصف Descriptor

واقع زمني حقيقي Real Time

الوحدات الملحقة بالمعالج Peripherals

وصلة التحكم الافتراضية Default Control Pipe

وصلة المقاطعة Interrupt Pipe.

وضع العُطل Idle Mode

وضع النوم Sleep Mode

قائمة المحتويات:

مقدمة عامة: 1

الفصل الأول: مقدمة حول أنظمة التطوير

I-1-مقدمة : 4

I-2- لغة البرمجة : 4

I -2-1- لغة التجميع Assembly Language : 5

I -2-2- لغة C: 6

I -2-3- لغة الجافا java : 6

I -3- المحاكاة Simulation : 7

I -3-1- المحاكاة الحاسوبية : 7

I -3-2- برنامج Matlab & Simulink : 8

I -3-3- برنامج Proteus : 8

I -4- المترجم البرمجي Compiler : 9

I -4-1- بيئة أردوينو التطويرية Arduino IDE : 10

I -4-2- بيئة تطوير MikroC PRO for PIC : 10

I -4-3- بيئة تطوير MPLAB IDE: 11

I -5- لوحات التطوير الالكترونية Development Boards : 11

I -5-1- لوحة أردوينو Arduino : 12

I -5-2- لوحة التطويرية LaunchPad MSP430 TI: 14

I-5-3- لوحة راسبيري باي Raspberry Pi: 15

I-5-4-لوحة التطوير Kit Pic32 Microchip: 17

I -5-5- اللوحة التطويرية Ready for PIC: 19

I- 6- خاتمة: 20

الفصل الثاني: تقديم المتحكم الدقيق PIC18F4550.

II-1- المقدمة : 22

II-2- تعريف المعالج الدقيق Microprocessor : 22

II-2-1- الوحدات الملحقة بالمعالج Peripherals : 22

II-2-1-1- الهزازOscillator: 22

II-2-1-2-الذاكرات Memories: 23

II-2-1-3- المؤقت Timer: 23

II-2-1-4- وحدة منافذ الإدخال /الإخراج I/O Ports: 23

II-2-1-5- وحدة الاتصال التسلسلي Communication USART: 23

II-2-1-6- المحول التماثلي الرقمي A/D Converter: 24

II-2-2- أنواع الذاكرات: 24

II-3- المتحكم الدقيق Microcontroller: 25
II-3-1- تعريف المتحكم الدقيق: 25
II-3-2- محاسن المتحكم الدقيق : 26
II-3-3- اختيار المتحكم الدقيق المناسب : 26
II-3-4- أنواع المتحكم الدقيق: 27
II -4- المتحكم PIC18f4550: 27
II-4-1- الخصائص العامة للمتحكم PIC18F4550 : 28
II-4-2- تعريف المتحكم PIC :..... 29
II-4-3- الشكل الخارجي للمتحكم PIC18F4550 : 29
II-4-4- أهم وظائف أطراف المتحكم PIC18F4550:..... 30
II-4-5- المخطط الداخلي لبنية المتحكم الدقيق PIC18F4550: 33
II -5- الذاكرات Memories:..... 34
II-5-1- ذاكرة البيانات المؤقتة Data Memory:..... 34
II -5-2- ذاكرة البرنامج(فلاش) : 35
II -5-3- ذاكرة البيانات EEPROM: 36
II-6- مصادر الساعة Clock Sources: 37
II -6-1- الهزاز الخارجي External Oscillator: 37
II-6-2- أنماط المصادر الخارجية للساعة External Clock وخصائصها : 38
II -6-3- دارة مضاعفة التردد PLL : 39
II -6-4- قسم الهزاز الداخلي Internal Oscillator:..... 41
II -6-5- أنماط الهزاز الداخلي:..... 41
II -7- منافذ الدخل والخرج I/O Ports:..... 42
II -7-1- المنفذA:..... 43..
II -7-2- المنفذB:..... 43.... .....
II -7-3- المنفذC:..... 44 .....
II -7-4- المنفذD:..... 44 .....
II -7-5- المنفذE:..... 45
II -8- المؤقت TIMER:..... 45
II-8-1- المؤقت TIMER0 : 45
II -8-2- المؤقت TIMER1: 45
II-8-3- المؤقت TIMER2: 46
II-8-4- المؤقت TIMER3: 46
II -9- إعادة التعيين Reset:..... 47
II-9-1- إعادة التعيين بواسطة الطرف MCLR:..... 47

II-9-2- إعادة التعيين عند وصل التغذية POR: 47
II-9-3- إعادة التعيين عند اِنخفاض الجهد BOR: 47
II-9-4- إعادة التعيين بواسطة مؤقت المراقبة WDT: 48
II -10- المقاطعة Interrupt : 48
II-10-1- السجلات INTCON: 49
II-10-2- السجلات PIR: 49
II-10-3- السجلات PIE: 49
II-10-4- السجلات IPR: 49
II-10-5- السجل RCON: 49
II -11- المحول التماثلي – الرقمي Analog To Digital Convertor: 49
II-11-1- خوارزمية عملية التحويل: 50
II-11-2- آلية عمل المحول ADC: 51
II -12- وحدة الاِتصال USB: 52
II-12-1- الذاكرة USB RAM: 53
II-12-2- إعدادات الهزاز في الاتصال USB: 54
II-12-3- مقاطعة USB: 54
II 12-4- الواصف USB Descriptor: 55
II-13- الخاتمة: 55
الفصل الثالث: تقنيات الاِتصال التسلسلي
III-1- المقدمة: 57
III-2- النقل التسلسلي المتزامن: 57
III-3- النقل غير متزامن: 57
III-4- بروتوكول الاتصال التسلسلي Recommended Standard RS-232: 58
III -4-1-تعريف: 58
III -4-2- خصائص المعيار RS-232: 58
III-4-3- الخصائص الفيزيائية: 59
III-4-4- الخصائص الوظيفية للمعيار RS-232: 60
III-4-5- أنواع الاتصال في البروتوكول RS-232: 60
III-4-5-1- الاِتصال أحادي الإتجاه Simplex: 60
III-4-5-2- الاِتصال النصف ثنائي Half duplex: 61
III-4-5-3- الاِتصال الثنائي الكامل Full duplex: 62
III-4-6- شرح البروتوكول RS-232: 62
III-4-7- تقنية المصافحة Handshaking: 63
III-4-8- بروتوكول الاِتصال التسلسلي UART في المتحكمات الدقيقة: 65

III-5- الناقل التسلسلي العام Universal Serial Bus USB: 65
III-5-1- تعريف: 65
III-5-2- أسلاك الناقل USB: 66
III-5-3- تشفير المعلومة: 68
III-5-4- التصميم الهندسي للخط USB: 68
III-5-5- إصدارات البروتوكول USB: 69
III-5-6- إطار البيانات Data Frame في الناقل USB: 69
III-5-7- أنواع الموصلات USB: 70
III-5-8- تحديد سرعة الناقل: 71
III-5-9- عملية إتصال المضيف مع الجهاز: 72
III-5-9-1-إعداد الإتصال: 72
III-5-9-2-مراحل الإعداد: 72
III-5-10- الواصف Descriptor: 73
III-5-11- أصناف الأجهزة Device Classes: 74
III-6- خاتمة: 75
الفصل الرابع : الجانب العملي.
IV-1- المقدمة : 77
IV-2- بروتوكولات البرمجة: 77
IV-2-1- طرق البرمجة داخل الدارة ICSP: 77
IV -2-2- تقسيم مساحة ذاكرة البرنامج Flash Memory: 78
IV-2-3- مؤشر عنوان الذاكرة Memory Address Pointer: 80
IV2-4- بدء وإنهاء البرمجة بالجهد العالي : 81
IV-2-5- بدء و إنهاء وضع البرمجة بالجهد المنخفض: 82
IV-2-6- نقل البيانات خلال عملية البرمجة: 83
IV-2-7- عملية مسح الذاكرة ICSP Erase: 84
IV-2-8- كتابة البرنامج على الذاكرة Code Memory Programing: 85
IV-2-9- برمجة الإعدادات Configuration Bits Programing: 87
IV-2-10- قراءة ذاكرة البرنامج Reading Flash Memory: 88
IV- 3- اللوحات التطويرية: 88
IV-4- اللوحة التطويرية الأولى: 88
IV-4-1- برنامج التحميل: 88
IV-4-2- برنامج USB HID Bootloader: 90
IV-4-3- تقسيم ذاكرة البرنامج : 90
IV-4-4- حرق برنامج التحميل USB HID Bootloader على ذاكرة المتحكم PIC18F4550: 91

IV-4 -5- واجهة التطبيق mikroBootloader: 92
IV-4-6- دارة اللوحة التطويرية: 92
IV-4-7- اِستعمال اللوحة التطويرية : 95
IV-5- اللوحة التطويرية الثانية: 97
IV-5-1- المبرمجة JDM: 97
IV-5-2- دارة المبرمجة JDM : 98
IV-5-3- دمج المبرمجة داخل اللوحة التطويرية: 99
IV-5-4- استعمال اللوحة التطويرية: 100
IV-6- اللوحة التطويرية الثالثة: 103
IV-6-1- دارة اللوحة التطويرية: 104
IV -6-2- آلية برمجة المتحكم على اللوحة التطويرية: 106
IV-6-3- التطبيق البرمجي The Application: 107
IV-6-4- تحليل شفرات الملف البرمجي: 109
IV-7- خاتمة: 111
خاتمة عامة: 113
قائمة المراجع 114
الملاحق 118

مقدمة عامَّة

مقدمة عامَّة:

مع اِنفتاح علم الإلكترونيات على مجال البرمجيات وظهور الإلكترونيات الرقمية أصبحت عجلة التطور والإختراع في تسارع مستمر، وظهرت تخصصات تُعنى بدراسة الإلكترونيك مُدمجةً مع علوم الحاسوب والبرمجيات مثل الأنظمة المضمنة Embedded Systems.

النظام المضمن هو نظام لمعالجة المعلومات يقوم بوظائف محددة ضمن جهاز اِلكتروني ما؛ كقياس درجة الحرارة أو التحكم في سرعة محرك...، تحتوي المنتجات الإلكترونية كالهواتف النقالة، ومكيف الهواء، والتلفاز، وحتى السيارة على العديد من الأنظمة المضمنة التي تُمَكِّنها من العمل بصفة آلية أو ما يطلق عليه اِسم التحكم الذكي.

أهم مكون للنظام المضمن هو وحدة المعالجة والمتمثلة غالبا في متحكم دقيق. بداية كان يتطلب العمل على دراسة وتطوير تطبيقات الأنظمة المضمنة بواسطة المتحكم تركيب دارات اِلكترونية وتوفير العناصر اللازمة لعمله ومعرفة واسعة بلغات البرمجة المستخدمة وبالبنية الداخلية للمتحكم وهذا كان معيقا للتقدم بسرعة في هذا المجال.

لكن مع ظهور اللوحات التطويرية كلوحة أردوينو أصبح العمل على تطوير تطبيقات الأنظمة المضمنة والإلكترونيات التفاعلية سريعا وأكثر سهولة؛ وذلك بِاستخدام دارة اِلكترونية واحدة ثابتة موصلة مباشرة مع أدوات الإدخال كالحساسات Sensors، وأدوات الإخراج كالشاشات. إضافة إلى ميزة البرمجة البسيطة من خلال بيئة تطوير تحتوي على مجموعة من المكتبات الجاهزة.

يعتبر المتحكم الدقيق من نوع PIC أكثر المتحكمات اِستخداما وشيوعًا بين هواة وطلبة وأساتذة الإلكترونيك لِمَا يوفره من ميزات وخصائص دون غيره من المتحكمات إضافة إلى سعره المنخفض ووفرته. لكن اللوحات التطويرية لمتحكمات PIC غير متوفرة وغالية الثمن مقارنة بمثيلاتها.

ومن هنا عملنا على مشروع اِنجاز لوحة تطوير تطبيقات للمتحكم الدقيق PIC18F، لتوفير الوسيلة اللازمة لتطوير أفضل وأسهل لتطبيقات الأنظمة المضمنة بواسطة هذا المتحكم وبإستغلال بيئة تطويرMikroC التي توفر بدورها سهولة وبساطة في البرمجة من خلال مجموعة من المكتبات الجاهزة والخاصة بمختلف ملحقات هذه المتحكمات.

الفصل الأول من مذكرتنا يتحدث عن الأدوات المستعملة في تطوير التطبيقات والوسائل اللازمة لتطوير نظام تطبيقي معين سواء أكان هذا النظام برمجيا أم إلكترونيا (عتاد).

في الفصل الثاني قمنا بإعطاء دراسة فزيائية مفصلة للمتحكم الدقيق PIC18F4550 الرقاقة المحورية في مشروعنا.

وقد تطرقنا في الفصل الثالث لتقنيتي الاتصال التسلسلي الأكثر استعمالا في عصرنا USB و UART من ناحية البنية الفيزيائية والمنطقية.

وكان الفصل الرابع مخصصا لشرح تفصيلي حول الإنجاز العملي لمشروعنا من ناحية العتاد والبرمجيات.

وقد توجنا هذه المذكرة بخاتمة عامة.

الفصل الأول : مقدمة حول أنظمة التطوير

I-1-مقدمة :

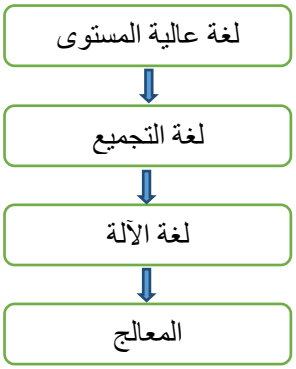
أنظمة تطوير التطبيقات هي الوسائل اللازمة لإنجاز نظام تطبيقي معين، تنقسم هذه الوسائل إلى برمجيات Software (لغة البرمجة، المترجم Compiler والمحاكي Simulator) وعتاد إلكتروني Hardware؛ الذي يمكن أن يكون لوحة إلكترونية لمعالج أو متحكم.

I-2-لغة البرمجة :

هي خوارزميات تتكون من مجموعة أوامر مكتوبة بشكل مميز موجهة لبرمجة الحاسوب [1]، أو بالأحرى المعالجات والمتحكمات الدقيقة وبعض الرقاقات الإلكترونية. يكتب برنامج الحاسوب بلغة برمجة عالية المستوى High Level، وهي لغة قريبة من لغة الإنسان يسهل عليه فهمها وكتابتها [2]، ومن أمثلتها : لغة C، C++، java و Python...

البرامج المكتوبة بلغة عالية المستوى لا تصلح أن تكون ضمن ذاكرة المعالج ليتم تنفيذها، فالمعالج لا يمكنه تمييز أي أمر من أوامر هذه البرامج – فهي لا معنى لها بالنسبة له– لذا يتم ترجمة البرامج المكتوبة بلغات عالية المستوى إلى لغة الآلة Machine Language، والتي يفهمها المعالج وأساسها النظام الثنائي Binary System [3]. ولصعوبة التعامل مع لغة الآلة بالنسبة إلينا كبشر تم اِبتكار لغة لتكون وسيطا بين اللغة العالية المستوى ولغة الآلة هذه اللغة هي لغة التجميع Assembly Language وهي لغة منخفضة المستوى Low Level مصممة على أساس الهندسة الداخلية للمعالج.

يمثل الشكل (I-1) مراحل ترجمة البرامج [2].



الشكلI−1 مراحل عملية ترجمة البرامج.

I-2-1-لغة التجميع Assembly Language :

يحتوي أي حاسوب على معالج دقيق Microprocessor يقوم بعمليات مختلفة (حسابية ومنطقية وتحكم). كل عائلة من المعالجات لها مجموعتها الخاصة من الأوامر والتي تكون بلغة الآلة أي أنها مكتوبة وفق النظام الثنائي Binary (0،1) [4]، وهذا يوجد صعوبة في فهمها وكتابتها من طرف المبرمجين، لذا تم تطوير لغة قريبة جدا من لغة الآلة لا تستعمل النظام الثنائي حتى تسهل ترجمتها إلى أوامر على الحاسوب، وتسهل البرمجة بها من قبل المبرمجين، هذه اللغة هي لغة التجميع؛ وهي عبارة عن أوامر مكونة من أحرف مختصرة للدلالة عليها(Mnémonique).

لكل معالج لغة تجميع خاصة به، وذلك حسب بنيته الداخلية بما فيها من سجلات Registers، وذاكرات Memories، فهي لغة مبنية على أساس بنية الحاسب Computer Architecture.

تتكون أسطر برامج لغة التجميع من ثلاثة أجزاء [5] :

الأمر Instruction : هو العملية التي سيقوم المعالج بتنفيذها، مثل : MOV BX.

المعامل Operand : هو ما سيتم تطبيق العملية عليه ،مثل : العدد 0F h.

العلامة Label : هي إشارة إلى أحد أسطر البرنامج، مثل : الاسم TOTO.

مثال : TOTO : MOV BX ,0F h

I-2-2-لغة C :

هي لغة برمجة عامة تعتبر ذات مستوى عالٍ، مصممة من طرف B.W.KERNIGHAN وD.M.RITCHIE، تم تصميمها في الأصل لتستعمل في التطوير، والعمل على نظام UNIX، وتعتبر أشهر لغات البرمجة وأكثرها اِستعمالا في تطوير التطبيقات وبرمجة الآلات [6].

ميزاتها [6] :

- سهلة التعلم ومفهومة.
- تدعم البرمجة الهيكلية Structured Language.
- البرمجة بكفاءة عالية.
- غير مرتبطة بمجال برمجي معين (تستعمل لبرمجة أنظمة التشغيل والآلات Machine).

تمتلك لغة C تعليمات برمجية تسمح بإنشاء برامج مبنية بشكل متناسق وسهلة الفهم والكتابة، ومن هذه التعليمات أمر الشرط if-else، أمر اختيار الاحتمالات Switch، وأوامر الحلقات while-for.

I-2-3-لغة الجافا java :

هي لغة برمجة للكائنات الموجهة Object Orient، تم إنشاؤها من طرف J.GOSLING عام 1992 في مختبرات شركة Sun Microsystems والتي اشترتها شركة Oracle بعد ذلك. تم تطوير لغة الجافا لتنافس لغة C++ [7].

تستعمل لغة الجافا نفس المفردات البرمجية Syntax للغة C مع اختلاف بسيط [8]. تعتبر هذه اللغة مستقلة عن بيئة التشغيل وقد عبّرت شركة Sun المطوِّرة للّغة عن هذا بالمبدأ القائل "اُكتب مرة واحدة وشغّل أينما تريد، وهذا يتطلب تعريف وتحقيق ما نسميه آلة جافا الافتراضية Java Virtual Machine (JVM) التي تحاكي عمل معالج له تعليماته الخاصة حيث يسمح تواجد هذه الآلة على

منصات عمل متنوعة بنقل برنامج جافا من بيئة تشغيل إلى أخرى وتشغيلها دون مشاكل [9]. حيث إن البرنامج المكتوب بلغة الجافا يعمل آليا على أنظمة Windows وMac وUnix...

تستخدم لغة الجافا في برمجة [10] :

- التطبيقات المختلفة والواجهات على الحواسيب.
- صفحات الويب.
- تطبيقات الهواتف المحمولة Mobile المختلفة.
- الألعاب (2D، 3D).

I-3-المحاكاة Simulation :

تعرف المحاكاة على أنها عملية تقليد لأداة حقيقة أو عملية فيزيائية أو حيوية، وهي تحاول أن تمثل وتقدم الصفات المميزة لسلوك نظام ما بواسطة نظام آخر يحاكي الأول، وهي محاولة إعادة عملية ما في ظروف اِصطناعية مشابهة إلى حد ما الظروف الحقيقية [11].

الهدف من المحاكاة [11]:

- تصميم وبناء نماذج مماثلة للأنظمة الحقيقية.
- دراسة النتائج المتوقعة.
- تجريب واختبار الأنظمة.

I-3-1-المحاكاة الحاسوبية :

هي إحدى طرق المحاكاة، حيث يتم اِستخدام تطبيقات معينة تسمح بإنشاء أنظمة اِفتراضية مماثلة للحقيقية، ويتم تشغيل هذه الانظمة في ظروف برمجية تحاكي الظروف الواقعية، كما تسمح هذه التطبيقات بدراسة النتائج المتحصل عليها أثناء عمل الأنظمة الافتراضية [11].

من أشهر برامج المحاكاة Matlab & Simulink الذي يستعمل في محاكاة أنواع مختلفة من الأنظمة، وبالنسبة للبرامج الخاصة بمحاكاة الأنظمة الالكترونية يوجد برنامج Proteus، و NI Multisim...

I-3-2-برنامج Matlab & Simulink :

مطور من طرف شركة Math Works والبرنامج الخاص بالمحاكاة Simulink - المدمج مع Matlab – هو لغة برمجة تخطيطية تسمح بتصميم ومحاكاة وتحليل العديد من الأنظمة الديناميكية تحت ما يسمى ببيئة تطوير واجهة المستخدم التخطيطية Graphical User Interface (GUI) environment.

يضم برنامج Simulink أقسام شاملة من المكتبات Libraries والأدوات Toolboxes للقيام بالمحاكاة الخطية Lineare والغير الخطية Nonlinear للأنظمة [12].

من الأنظمة التي يقوم البرنامج بمحاكاتها وتحليلها [13] :

- أنظمة التحكم Control Systems.
- أنظمة معالجة الإشارة Signal Processing Systems.
- أنظمة الاتصالات Communication Systems.
- الأنظمة المضمنة Embedded Systems.
- الأنظمة الفيزيائية وغيرها من الأنظمة الديناميكية.

I-3-3-برنامج Proteus :

هو بيئة تطوير خاصة بتصميم ومحاكاة الأنظمة الإلكترونية منتج من طرف شركة Labcenter Electronics، حيث يعتبر برنامج كامل لمحاكاة الدارات الالكترونية ونمذجة وتصميم

الدارات المطبوعة، إذ إنّه ينقسم إلى قسمين، قسم خاص بتصميم ومحاكاة الدارات الالكترونية وهو Schematic Capture وقسم خاص بتصميم وإنجاز الدارات المطبوعة وهو PCB Layout [14].

يتضمن برنامج Proteus العديد من الأدوات البرمجية لتوفير أفضل محاكاة ليكون بين يدي الطالب والمهندس بيئة متكاملة من أجل عملية محاكاة واقعية، فهو يجمع بين نظام SPICE لمحاكاة الدارات والعناصر الالكترونية (يعرضها على شكل مقاطع صور متحركة) وبين نماذج المعالجات والمتحكمات الدقيقة، تسير عملية المحاكاة وفق واقع زمني حقيقي (أو مقارب للحقيقة) Real Time Simulation وبشروط ومتطلبات بسيطة للحاسوب المستخدمة [15].

يملك Proteus أيضا خاصية مهمة جدا تميزه عن باقي برامج المحاكاة المعروفة، وهي أنه يستطيع أن يحاكي مختلف التفاعلات والعمليات المتبادلة ما بين برنامج مخزن في المتحكم الدقيق وبين كافة العناصر الالكترونية التماثلية والرقمية الموصولة به، وبإمكانه أيضا أن يحاكي أي تصميم يحتوي على أكثر من وحدة معالجة مركزية حيث يوفر سهولة في اِستخدام واحد أو أكثر من المعالجات الدقيقة وربطها ببعضها لتتفاعل فيما بينها [15].

I-4-المترجم البرمجي Compiler :

بدايةً كانت البرامج تكتب أصلا بلغات متدنية المستوى يفهمها الحاسوب مباشرة، الآن أصبحت تكتب بلغات عالية المستوى، والسبب في هذا أن البرامج زادت تعقيداتها وحساباتها، مما جعل البرمجة المباشرة بلغة دونية أمرا صعبا، كما أن انتشار عدة بنى للحاسوب (كل شركة لها بنية خاصة بمعالجاتها) جعل من اللازم كتابة البرامج عدة مرات بلغة دونية لكي يعمل على حواسيب مختلفة، لغات البرمجة عالية المستوى تركز على المهم من الأمور في البرمجة وتسهّل من إدارة المشاريع البرمجية وتجعل اِستهداف عدة أجهزة ممكنا بملف مصدري واحد مهما تعددت البنى (مع مراعاة بعض

التفاصيل الصغيرة عند ذلك وضبطها)، هذا الأخير يتم ترجمته إلى ملف بلغة دونية بواسطة مترجم برمجي وفق البنية المراد العمل عليها[16].

المترجم هو برنامج حاسوبي يقوم بقراءة البرنامج المكتوب بإحدى لغات البرمجة العالية المستوى (لغة الملف المصدري) وترجمته إلى برنامج يماثله لكن بلغة منخفضة (لغة الملف المستهدف)، أهم جزء من عملية الترجمة هو تحديد الأخطاء في الملف المصدري وإعلام المستخدم بها [17].

I-4-1-بيئة أردوينو التطويرية Arduino IDE :

هي برنامج مفتوح المصدر لكتابة وترجمة الملفات البرمجية الخاصة بأردوينو وتحميلها على لوحة أردينو، تم تطويره باستعمال لغة الجافا وهو متوافق مع أنظمة تشغيل متعددة (Windows، وMac، وLinux) [18].

الملف المصدري لبرنامج Arduino IDE تم إصداره تحت ترخيص GNU للمصادر المفتوحة، يدعم البرنامج كل من لغة C و++C، يعتمد في برمجيته على مكتبات متعددة للمداخل والمخارج بمختلف أنواعها يمكن للمستخدم كتابة الملف البرمجي ضمن دالتين أساسيتين، دالة الاعدادات void setup ودالة الحلقة التكرارية void loop، يقوم البرنامج أيضا بعملية الترجمة البرمجية حيث يحول الملف البرمجي المكتوب إلى ملف Hex بالنظام Hexadécimal هذا الأخير هو الذي يتم تحميله على لوحة الأردوينو عن طريق أداة ضمن Arduino IDE [19].

I-4-2-بيئة تطوير MikroC PRO for PIC :

هو برنامج لكتابة وترجمة وإدارة الملفات البرمجية الخاصة بالمتحكمات من النوع PIC، مطور من طرف شركة MikroElektronika ، يحتوي على العديد من الأدوات البرمجية التي تسمح ببرمجة سهلة وسريعة للمتحكم PIC كما أنه يقوم بترجمة البرنامج المكتوب بلغة السي C إلى ملف برمجي HEX ويوفر العديد من الحلول لتطوير مختلف تطبيقات الأنظمة المضمنة بحيث يقدم

للمستخدم مجموعة من المكتبات الجاهزة والمتنوعة : مكتبة للتعامل مع الذاكرات بأنواعها، مكتبة لشاشات العرض، مكتبة التحويلات، مكتبة الاتصال...الخ [20].

يحتوي أيضا على مجموعة من الأدوات للبرمجة السريعة باستعمال برنامج تحميل Bootloader، إضافة إلى توافقه مع مجموعة من اللوحات الإلكترونية التطويرية المنتجة من قبل نفس الشركة (MickroElektronika) حيث يوفر لها خاصية تحميل البرنامج عبر وصلة USB.

I-4-3-بيئة تطوير MPLAB IDE [21] :

هو برنامج مجاني يعتبر بيئة تطوير متكاملة لتطبيقات الأنظمة المضمنة التي تحتوي على المتحكمات PIC وdsPIC المنتجة من طرف شركة Microchip Technology.

تم تطوير MPLAB باستعمال لغة C++ يوفر البرنامج العديد من الأدوات لكتابة الملفات البرمجية وتغييرها وتصحيحها وتنقيحها وترجمتها قصد برمجة المتحكمات ذات 8 بت من النوع PIC وAVR والمتحكمات ذات 16بت (PIC24 & dsPIC) والمتحكمات 32 بت من النوع SAM (ARM) وPIC32.

يدعم MPLAB مجموعة من الأجهزة البرمجية المنتجة من طرف نفس الشركة (Microchip) مثل MPLAB ICD 3 وMPLAB ICE والمبرمجة PICKIT.

I-5-لوحات التطوير الالكترونية Development Boards :

للعمل على تطوير الأنظمة المضمنة يجب توفر شيئين أساسيين : لوحة تطوير الكترونية وبيئة تطوير مدمجةIDE .

اللوحات التطويرية الالكترونية هي دارات مطبوعة PCB تحتوي على متحكم أو معالج، تكمن أهميتها في كونها توفر اِستعمال سهل وسريع للمتحكمات أو المعالجات من جانب البرمجة والتشغيل والاتصال ضمن مشاريع الأنظمة المضمنة.

تحتوي لوحة التطوير على وحدة معالجة –قد تكون متحكم دقيق أو معالج دقيق مع ملحقاته – محاطة ببيئة تشغيلها (دارة تغذية، مبرمجة، دارة تصفير Reset وهزاز لإعطاء نبضات التوقيت Clock) إضافة إلى وحدات ملحقة مثل وحدات الإدخال كلوحة المفاتيح، الحساسات التماثلية مع منفذ ADC والحساسات الرقمية، منافذ مباشرة لمختلف تقنيات الِاتصال مثل الِاتصال التسلسلي Serial Port، USB، Ethernet إن وجد، ووحدات إخراج كشاشات العرض LCD... [22] .

و إليكم بعض أنواع لوحات التطوير الالكترونية :

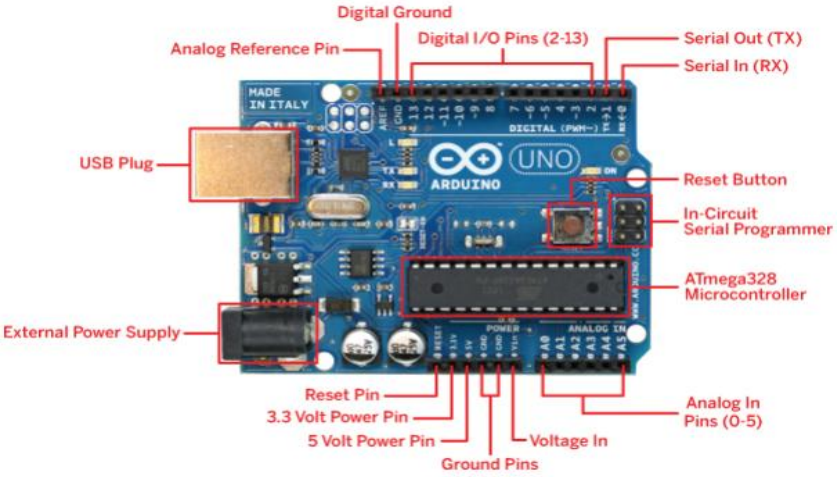
I-5-1-لوحة أردوينو Arduino :

هي لوحة الكترونية تطويرية مفتوحة المصدر للمتحكم الدقيق ATMEGA تعمل مع بيئة تطوير مجانية Arduino IDE وهي الأخرى مفتوحة المصدر.

كمثال عن لوحات الأردوينو نأخذ الإصدار Arduino UNO الذي يحتوي على المتحكم ATMega 328 وهو متحكم دقيق من عائلة AVR ذو خط بيانات 8 بت تتم برمجته بلغة السي C ضمن Arduino IDE مع اِستعمال العديد من المكتبات الجاهزة والتي تسهل اِستخدام المداخل والمخارج المختلفة منها مكتبة : محول تماثلي رقمي ADC، توليد إشارات شبه تماثلية (تعديل عرض النبضة) PWM، الِاتصال التسلسلي بمختلف أنواعه :I2C، SPI، USB وUSART... إضافة إلى العديد من المكتبات الخاصة بحساسات متنوعة [22].

عملية تحميل الملف البرمجي على ذاكرة المتحكم سهلة جدا وبسيطة وذلك عن طريق برنامج تحميل موجود مسبقا في الذاكرة يعمل على اِستقبال الملف البرمجي من نوع Hex ووضعه ضمن ذاكرة البرنامج flash memory.

الشكل(I–2) يمثل لوحة أردوينو.



الشكلI-2 لوحة أردوينو UNO مع توضيح لمكوناتها[22].

خصائص لوحة الأردوينو UNO [22] :

- متحكم دقيقATMega328 ذو خط بيانات 8 Bits.
- ذاكرة برنامج Flash Memory بسعة 32 Kbytes.
- ذاكرة بيانات مؤقتة SRAM بسعة 2 Kbytes.
- ذاكرة بيانات دائمة EEPROM بسعة 1Kbytes.
- 14 منفذ للمداخل/المخارج الرقمية.
- المؤقت/العداد Timer/Counter : يوجد 3 وحدات منه Timer0 و Timer1 (يعملان على 8 Bits) Timer2 (يعمل على 16 Bits).
- 6 منافذ للمحول التماثلي الرقميADC بمعيار 10 Bits.
- خط اتصال I2C و SPI.
- منفذ اتصال تسلسلي USART.
- مقارن تماثليAnalog comparator.
- مؤقت المراقبة Watchdog Timer .
- وحدة مقاطعات Interrupts.

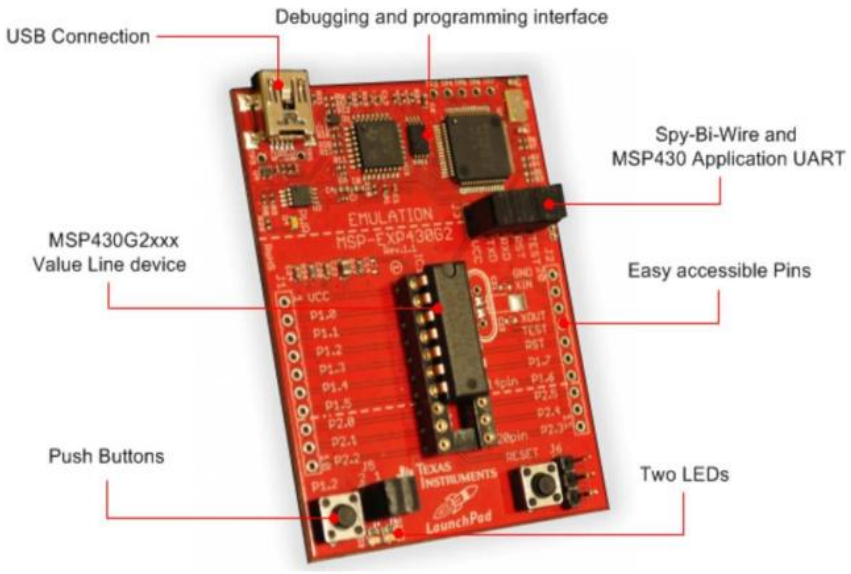
I-2-5-الوحة التطويرية LaunchPad MSP430 TI :

هي لوحة تطوير الكترونية منتجة من طرف شركة Texas Instruments تعتبر من لوحات التطوير الأقل ثمنا، تستخدم في تطوير الأنظمة المضمنة، تحتوي على متحكم دقيق من نوع MSP430 المنتج من طرف نفس الشركة، إضافة إلى ملحقات للِاتصال التسلسلي وواجهة خاصة بالتنقيح والبرمجة Debugging and Programming Interface [22].

لوحة LaunchPad قابلة للبرمجة بواسطة عدة بيئات تطوير مجانية منها : Code Composer Studio (CCS)IDE و IAR Embedded Workbench...

تدعم اللوحة أيضا اِستخدام متحكمات دقيقة مختلفة في عدد الوصلات(DIP 14 أو DIP 20) شرط أن تكون من نفس العائلة MSP430 [23].

الشكل(I-3) يمثل لوحة LaunchPad.



الشكلI-3 لوحة LaunchPad مع توضيحها لمكوناتها[22].

خصائص لوحة LaunchPad [22] :

- واجهة برمجة بمنفذ USB.
- الإتصال التسلسلي بسرعة تدفق قصوى 9600 bit/s.
- متوافقة مع المتحكمات : MSP430G2XX2، MSP430G2XX3، MSP430F20XX.
- المتحكم MSP430G2553 : خط بيانات 16 Bits، 8 قنوات لمدخل المحول التماثلي الرقمي ADC بمعيار 10 Bits، مقارن Comparator، ذاكرة برنامج Flash Memory بسعة 16 Kbytes وذاكرة مؤقتة RAM بسعة 512 Bytes.
- المتحكم MSP430G2452 : خط بيانات 16 Bits، 8 قنوات لمدخل المحول التماثلي الرقمي ADC بمعيار 10 Bits، مقارن، ذاكرة برنامج بسعة 8 Kbytes وذاكرة مؤقتة SRAM بسعة 256 Bytes.
- هزاز كريستال للساعة بتردد 32.768 Khz.

I-5-3-لوحة راسبيري باي Raspberry Pi :

تمثل لوحة الراسبيري باي حاسبا آليا متكاملا ومصنوعا من شريحة الكترونية واحدة، حيث تحتوي على مكونات الحاسب الآلي التقليدي وهي معالج بيانات CPU أحادي النواة بسرعة 700Mhz ومعالج الرسوميات GPU ثنائي النواة بسرعة 250Mhz، وذاكرة عشوائية RAM تصل إلى 512 Mbytes، وبالإضافة إلى وصلات تحكم رقمية بإمكانها التحكم في القطع الالكترونية والكهربائية المختلفة [24].

من هذا التعريف نستنتج أن الراسبيري باي يستخدم كأي حاسوب تقليدي، وأيضا كلوحة تطوير سريعة ومتطورة لتطبيقات الأنظمة المضمنة [24].

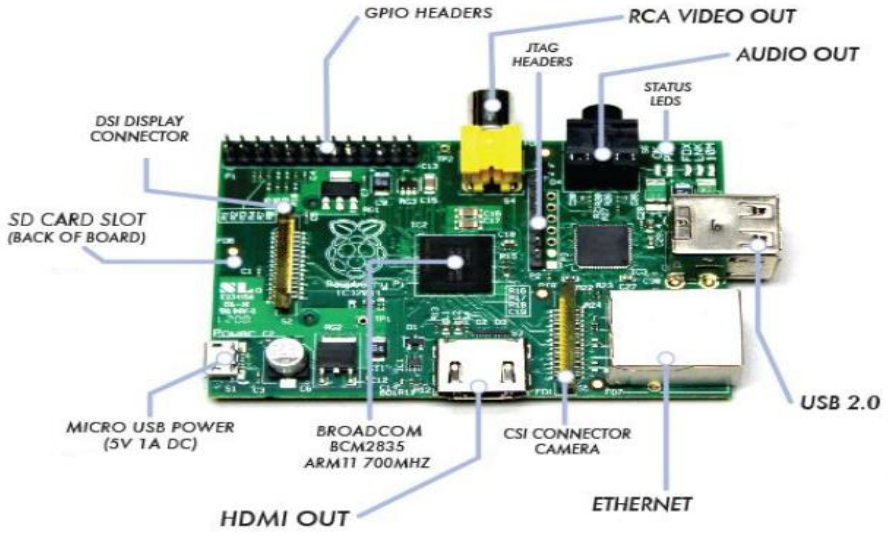
كل هذه الإمكانيات متواجدة على رقاقة صغيرة فقط وهو ما يعرف باسم "النظام على شريحة System on chip SOC" ويتم تشغيل هذا الحاسوب الصغير بأنظمة لينيكس Linux مفتوحة المصدر[24].

تتعدد أنظمة التشغيل الخاصة بالراسبيري باي كل منها له ميزات مختلفة لكن معظمها مشتق من النظام الأصلي Linux، ومن هذه الأنظمة [24]:

نظام راسبيان Raspbian : وهو نظام يتميز بوفرة البرامج لجميع الأغراض، حيث يحتوي على مجموعة ضخمة جدا منها وهو من أشهر أنظمة التشغيل للراسبيري.

نظام Occidentalis : وهو مشتق من الراسبيان ومن تعديل شركة الالكترونيات الشهيرةَ Adafruit حيث تم تخصيص النظام لتطبيقات التحكم والالكترونيات المتقدمة لتسهيل اِستخدام بروتوكولات التحكم المختلفة.

الشكل(I-4) يمثل لوحة راسبيري باي.



الشكلI-4 : لوحة Raspberry Pi مع توضيح لمكوناتها[24].

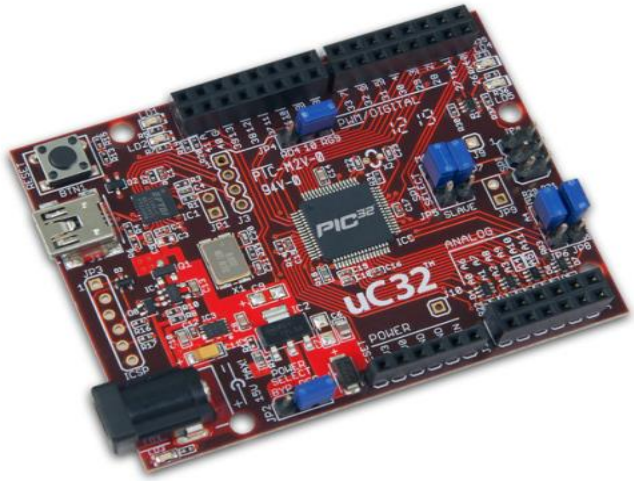
خصائص لوحة Raspberry Pi Model B [24]:

- معالج بيانات ARM1176JZF (700 Mhz Low Power).
- معالج رسوميات GPU (Dual Core 280 Mhz with shared memory).
- ذاكرة مؤقتة RAM بسعة 512 Mbytes.
- مخرجان للفيديو HDMI و RCA-Video.
- مخرج للصوت Audio stéreo Jack 3.5 mm.
- وحدة قراءة/كتابة من وعلى بطاقة الذاكرة الخارجية.
- منفذان للإتصال USB.
- منفذ الشبكة Ethernet.
- وصلات مداخل/مخارج GPIO للتحكم الالكتروني، حيث يمكن إستعمال الراسبيري كمتحكم دقيق.
- واجهات إتصال مختلفة : SPI، I2C و S2C.
- واجهة ربط مع الكاميرا CSI.
- واجهة ربط مع شاشة لمس DSI.
- مدخل Micro USB للتغذية بجهد 5V.

I-5-4-لوحة التطوير Kit Pic32 Microchip :

لوحة Kit Pic32 أو uC32 هي لوحة تطويرية من إنتاج شركة Microchip مصممة على أساس المتحكم Pic32، تعتبر مثيل للوحة Arduino UNO المفتوحة المصدر، فهي تحتوي على منفذ USB للإتصال التسلسلي مع بيئة التطوير IDE وللتغذية بجهد 5V من نفس المنفذ [22].

الشكل(I-5) يمثل لوحة uC32.



الشكلI–5 اللوحة التطويرية uC32 [22].

تحتوي اللوحة على متحكم دقيق PIC32MX340F512؛ هذا المتحكم يحتوي معالج 32Bits بسرعة 80Mhz، وذاكرة برنامج بسعة 512 Kbytes، وذاكرة بيانات مؤقتة SRAM بسعة 32 Kbytes [22].

يتم برمجة لوحة uC32 باستخدام بيئة تطوير متكاملة تدعى MPIDE مشتقة من بيئة تطوير أردوينو Arduino IDE، ومعدلة لتتوافق مع المتحكمات PIC32 [22].

كما تدعم لوحة uC32 البرمجة باستخدام بيئة تطوير MPLAB IDE والمبرمجة PICKIT3.

خصائص اللوحة التطويرية uC32 [22]:

- متحكم دقيق PIC32MX340F512H بخط بيانات 32 Bits.
- متماثلة ومتوافقة مع لوحة Arduino UNO.
- 42 وصلة للمداخل/مخارج، 12 منها يمكن أن تستعمل كمداخل تماثلية للمحول التماثليADC.
- محول تماثلي ADC بمعيار 10 Bits.

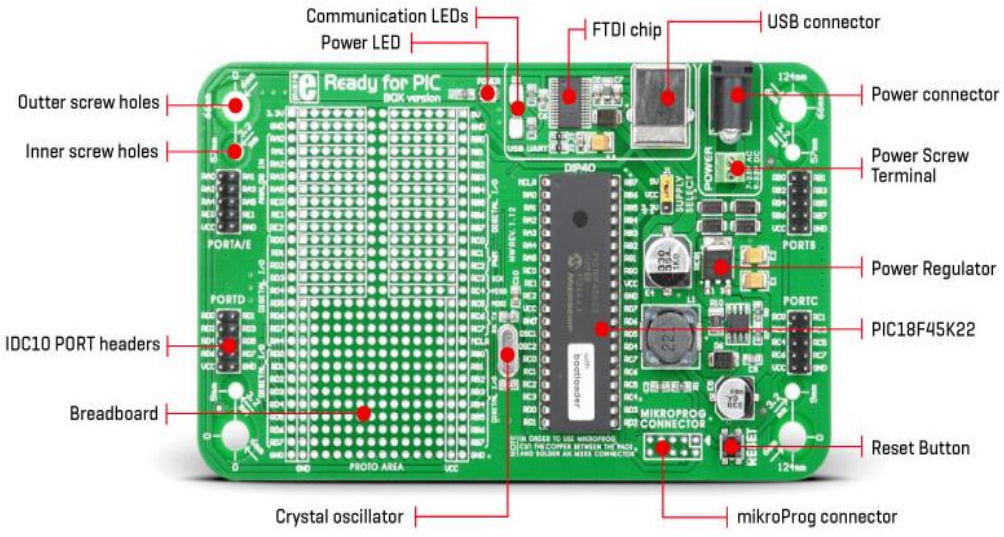
- يعمل المتحكم بجهد [2.3V، 3.6V].
- منافذ اِتصال متنوعة UART، SPI وI2c.
- منافذ للإشارات شبه التماثلية (تعديل عرض النبضة) PWM.

I-5-5-اللوحة التطويرية Ready for PIC :

مطورة من طرف شركة MikroElektronika، مصممة على أساس المتحكم PIC18F45K22، تحتوي وصلات لجميع المنافذ Ports دون تخصيص، قابلة للبرمجة باستعمال خاصية برنامج التحميل بواسطة أداة MikroBootloader ضمن بيئة تطوير MikroC، كما يمكن برمجتها مباشرة باستعمال المبرمج MikroProg المتوافق معها[25].

لوحة Ready for PIC مدعومة أيضا من عدة مترجمات للشركة نفسها: MikroElektronika، MikroC، MikroBasic وMikroPascal [25].

الشكل(I-6) يمثل لوحة Ready for Pic.



الشكلI- 6 : اللوحة التطويرية Ready for PIC مع توضيح لمكوناتها[25].

خصائص لوحة Ready for PIC [25]:

- متحكم دقيق 18F45K22 8 Bits.
- واجهة اِتصال تسلسلي UART بمنفذ USB.
- واجهة برمجة USB.
- وصلات جاهزة لجميع المنافذ Ports.
- هزاز كريستال بتردد 8 Mhz.
- مزودة بمساحة تحتوي لوحة تجريبية Bread Board.

I-6-خاتمة :

في هذا الفصل تحدثنا عن أنظمة التطوير وأنها تنقسم إلى جزأين أساسيين : جزء برمجي وجزء العتاد الالكتروني، كما تحدثنا عن أساسيات كل جزء حسب الترتيب العملي لإنجاز الأنظمة، مع إعطاء أمثلة عن البرمجيات والعتاد الموجود على الساحة العملية لدراسة وتطوير الأنظمة المضمنة.

الفصل الثاني : المتحكم الدقيق PIC18F4550.

II-1-المقدمة :

في هذا الفصل سنتطرق للحديث عن المتحكم الدقيق؛ باعتباره العنصر الأساس في الألواح التطويرية اِنطلاقا من تعريف المتحكمات، ومكوناتها، واِنتهاء بتقديم المتحكم الدقيق المستعمل في المشروع.

II-2-تعريف المعالج الدقيق Microprocessor :

هو دارة متكاملة IC تمثل وحدة معالجة للبيانات، وقد ظهر لأول مرة عام 1971 عندما قامت شركة Intel بتصنيع أول معالج هو 4004 بخط بيانات 4 Bits [1].

يقوم المعالج بأداء عدة وظائف وذلك حسب البرنامج الذي يقوم بتنفيذه، وهو بالتالي يمثل دارة متكاملة متعددة المهام.

يتكون المعالج الدقيق من أربعة أقسام :

- وحدة تحكم CPU.
- وحدة حساب ومنطق ALU.
- السجلات Registers.

II-2-1-الوحدات الملحقة بالمعالج Peripherals :

الدارة المتكاملة للمعالج وحدها لا معنى لها في أي مشروع إن لم يتم توصيلها بملحقات على حسب المشروع المراد إنجازه، ومن هذه الملحقات :

II-2-1-1-الهزاز Oscillator :

هو أهم الوحدات الملحقة بالمعالج ويعتبر بمثابة قلبه النابض، فهو يقوم بتزويد المعالج بإشارات دورية ذات تردد معين يستعملها المعالج لمزامنة وظائفه إضافة إلى استخدامات أخرى، له عدة أنواع : الكريستال، السيراميك، وهزاز نمط مقاومة مكثفة RC [2].

II-2-1-2-الذاكرات Memories :

يحتاجها المعالج لتخزين البيانات وإجراء العمليات المختلفة بسهو لة وهي على نوعين [1]:

ذاكرة الوصول العشوائي Random Access Memory RAM : هي ذاكرة مؤقتة تفقد محتوياتها بمجرد انقطاع التيار الكهربائي عنها.

ذاكرة دائمة Read Only Memory ROM : تمتاز بكونها ذاكرة دائمة قابلة للقراءة فقط، لكن تم استحداث أنواع مطورة منها قابلة للكتابة والقراءة وهي EPROM، وEEPROM.

II-2-1-3-المؤقت Timer :

هو أحد الملحقات الأساسية في عدة مشاريع، يستخدم في العمليات الخاصة بالعد Counter، والتوقيت Timer، مثل ضبط زمن تأخير للمعالج أثناء عمله Time Delay، والعد إلى قيمة معينة باستغلال حدث ما، كما يستخدم في توليد الاشارات شبه التماثلية PWM [3].

II-2-1-4-وحدة منافذ الإدخال /الإخراج I/O Ports :

هي أحد وحدات الاتصال الخارجي للمعالج، تحتوي على منافذ لكل منها مجموعة من الوصلات تأخذ قيمة منطقية 0،1 يمكن التحكم فيها واستخدامها كمخرج ويمكن استخدامها كمدخل وقراءة قيمتها 0 أو 1 من طرف المعالج [3].

II-2-1-5-وحدة الاتصال التسلسلي Communication USART :

هي وحدة للاتصال الخارجي للمعالج، تقوم بعمليات الاتصال التسلسلي وفق عدة أوضاع مختلفة، يمكن اعتبار عملها مع المعالج على أنها تقوم بتحويل البيانات التي تستقبلها منه على التوازي Parallel بواسطة خط البيانات Data bus وتحويلها إلى بيانات في حزمة تسلسلية من البتات موجهة إلى الجهاز المستقبل [3].

II-2-1-6-المحول التماثلي الرقمي A/D Converter :

بما أن المعالج الدقيق لا يتعامل إلا مع الإشارات الرقمية Digital في عملياته المختلفة كان لزاما أن يضاف إليه وحدة ملحقة تقوم بدور الوسيط بينه وبين العالم الخارجي وظواهره ذات الإشارات التماثلية، هذه الوحدة هي المحول التماثلي الرقمي. تقوم هذه الأخيرة بتحويل الإشارات التماثلية إلى إشارات رقمية موجهة للمعالج. لكل محول معيار معين من البتات وجهد مرجعي يتم تحديده [3].

ويوجد عدة ملحقات أخرى مثل وحدة متحكم المقاطعة PIC، وحدة الوصول للذاكرة DMA...

II-2-2-أنواع الذاكرات :

كما أسلفنا الذكر تنقسم الذاكرات المستعملة حاليا إلى نوعين RAM وEEPROM، وهذه الأنواع بدورها تنقسم إلى أنواع أخرى من الذاكرات لها نفس الدور، وتختلف في تركيبتها الصناعية، وفي أدائها وهي على التالي:

الذاكرة RAM : Static RAM (SRAM)، Dynamic RAM (DRAM).

الفرق بينهما هو العنصر الذي تصنع منه الذاكرة، حيث أن SRAM الموجودة في المتحكمات تصنع من القَلَّاب (Flip-Flop) الذي يتميز بالقدرة على الاحتفاظ بالبيانات بأقل تيار كهربي ممكن وبالتالي فهو الخيار الأفضل من ناحية الاستهلاك للطاقة كما أن البيانات الموجودة عليه لا تحتاج لعملية تجديد Refreshing مثل الذاكرة "dynamic RAM" الموجودة في الحواسيب التقليدية والتي تصنع من المكثفات الطفيلية Parasitic Capacitors، والتي تحتاج دائمًا لعملية تجديد وإلا تضيع البيانات المخزنة بداخلها مع مرور الوقت (أكثر من 10مللي ثانية كفيلة باختفاء البيانات من DRAM) كما أنها تستهلك الكثير من الطاقة بسبب هذه العملية [6]. هذا النوع من الذاكرة أبطأ من الـSRAM وأكثر استخداما حتى أصبح المصطلح RAM يدل على ذاكرة الوصول العشوائي الديناميكية DRAM [1].

الذاكرة EEPROM : EEPROM العادية والذاكرة فلاش Flash Memory.

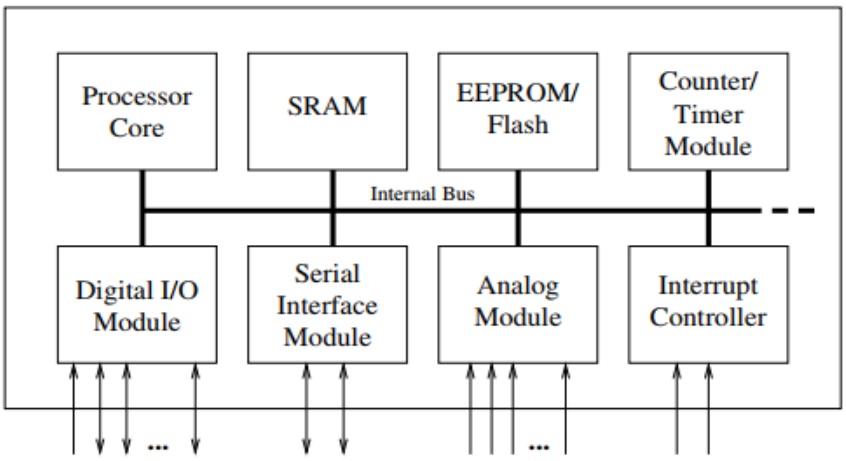
الذاكرة EEPROM تعني الذاكرة التي تستخدم للقراءة فقط وتحتفظ بالبيانات بعد انقطاع التغذية ويمكن الكتابة فيها وبرمجتها كهربيا بطريقة خاصة. المتحكمات الدقيقة غالبًا ما تحتوي على نوعين من EEPROM الأولى تسمى Flash؛ لأنها سريعة جدًا في كتابة البيانات وقد تصل سرعة الكتابة عليها إلى واحد ميغا بايت في الثانية؛ فمثلا قد تكتب 1 بايت بداخل الفلاش في زمن 1ميكرو ثانية فقط بينما EEPROM التقليدية بطيئة للغاية مقارنة بالفلاش حيث إن كتابة 1بايت بداخلها قد يستغرق 1مللي ثانية (يعني أبطأ بنحو 1000مرة من الفلاش) [4].

II-3-المتحكم الدقيق Microcontroller :

II-3-1-تعريف المتحكم الدقيق :

يُعْرّف المتحكم الدقيق على أنه وِحدة معالجة للمعلومات كما هو حال المعالج الدقيق، لكنه يختلف عنه؛ بِكَون المعالج جزءا منه إضافة إلى الذاكرات وباقي الوحدات الملحقة (المنافذ، المؤقت، المحول...). يوجد العديد من العائلات المختلفة للمتحكمات الدقيقة على حسب الشركة المصنعة [4].

الشكل التالي يوضح مخطط البنية الداخلية للمتحكم.



الشكلII-1- مخطط مبسط للبنية الداخلية للمتحكم الدقيق[3].

II-3-2-محاسن المتحكم الدقيق :

- دارة متكاملة تحتوي عدة عناصر ولواحق الكترونية مقارنة بمعالج مع لواحقه على دارة مطبوعة، يعتبر المتحكم الدقيق الخيار الأنسب في عدة مشاريع.
- قليل التكلفة مقارنة بعناصر أخرى في مكانه[5].
- استهلاك قليل للطاقة فهو يعمل على أقصى جهد 5 فولت وتيار صغير جدا.
- صغر حجمه ميزة تمكننا من استخدامه في عدة مشاريع لسهولة تموضعه في الدارات المطبوعة[6].

II-3-3-اختيار المتحكم الدقيق المناسب :

تعتبر مهارة اختيار المتحكم المناسب من أهم ما يجب أن يتعلمه أي مهندس نظم مدمجة، حيث إن الشركات المنتجة للمتحكمات الدقيقة عادة ما تصنع المئات منها وتقسمها إلى عائلات تختلف فيما بينها على حسب السعر والإمكانيات لكل متحكم، لذا يستوجب حسن الاختيار للمتحكم المناسب لأداء أفضل تصميم وبأقل سعر ممكن[4].

تعتبر أهم العوامل المؤثرة في اختيار المتحكمات الدقيقة هي[4]:

- سرعة معالجة البيانات والاستجابة المطلوبة.
- عدد أطراف التحكم العامة GPIO والتي تمثل عدد المداخل والمخارج.
- عدد وإمكانيات ملحقات المعالج التي يحتويها المتحكم.
- مساحة الذاكرة المطلوبة والتي ستحدد حجم البرنامج الذي سيوضع في المتحكم.
- البيئة والمجال التي سيعمل فيها المتحكم الدقيق مثل درجة الحرارة ومعدل استهلاك الطاقة.
- توفر الوسائل البرمجية والاجهزة اللازمة لنقل الملف البرمجي إلى ذاكرة المتحكم بسهولة.
- لغة البرمجة الخاصة بالمتحكم وكذا بيئات التطوير المدمجة IDE المتوافقة معه.

II-3-4- أنواع المتحكم الدقيق :

كما سبق وذكرنا فإن للمتحكمات الدقيقة أنواع مختلفة على حسب الشركة المصنعة لها وتنقسم هذه الأنواع إلى عائلات مختلفة فيما بينها في الإمكانيات التي توفرها للمستخدم في مشاريعه من حجم الذاكرات واتصالات مختلفة وتحويلات وعدد الوصلات المتاحة...الخ. الجدول التالي يوضح أمثلة عن أنواع المتحكمات وكذا العائلات الموجودة في كل نوع :

جدول II-1- جدول لبعض أنواع المتحكمات[4].

اسم الشركة المصنعة	اسم المتحكم	العائلات الموجودة
Microchip	PIC	PIC10،PIC12، PIC16، PIC18 ، PIC24،PIC32
Intel	MCS-51	8051
ATMEL	AVR	ATmega، AT90، AVR32
Zilog	/	Z80،Z180
TOSHIBA	/	TX19A

في مشروعنا هذا اخترنا العمل على المتحكم PIC18F4550 من شركة Microchip نظرا لعدة عوامل منها :

- وفرته في الأسواق وفي المختبرات الجامعية.
- شيوع استخدامه بكثرة في المشاريع لدى طلبة وأساتذة وهواة الإلكترونيك.
- سعره المنخفض مقارنة بباقي الأنواع.
- الإمكانيات العالية التي توفرها هذه العائلة من المتحكمات PIC.

II-4-المتحكم PIC18f4550 :

كل ما سنقوله ينطبق على المتحكمات PIC18F2455/2550/4455/4550 من العائلة 18F نظرا لتشابهها في الخصائص والأوصاف ضمن هذه العائلة التي تضم عدة مجموعات أخرى من المتحكمات. وسنخصص حديثنا عن المتحكم PIC18F4550.

II-4-1-الخصائص العامة للمتحكم PIC18F4550 :

- منفذ اتصال USB 2.0 بسرعة دنيا 1،5 Mbytes/s وسرعة قصوى 12 Mbytes/s.
- ساعة خارجية External Clock تعمل وفق وضعين 2 Modes بتردد أعظمي 48Mhz.
- هزاز داخلي بـ 8 ترددات مختلفة حسب الاختيار من 31 Khz إلى 8 Mhz.
- وحدة PLL من أجل الاتصال USB.
- تيار أعظمي 25 mA.
- 3 مصادر للمقاطعة الخارجية Interruptions.
- وحدة CCP (Capture/Compare/PWM).
- وحدة ECCP (وحدة CCP معدلة Enhanced بميزات إضافية مثل الإغلاق الآلي وإعادة التشغيل بصفة ألية Auto-shutdown & auto-restart).
- وحدة اتصال تسلسلي USART.
- وحدة اتصال تسلسلي MSSP للبروتوكولات SPI وI2C.
- 13 قناة للمحول التماثلي الرقمي ADC بمعيار 10 Bits.
- ذاكرة مؤقتة SRAM بسعة 2 Kbytes.
- ذاكرة دائمة للبيانات EEPROM بسعة 256 Bytes.
- ذاكرة برنامج Flash Memory بسعة 32 Kbytes.
- 4 وحدات للمؤقت Timer، 3 منها بحجم 16 Bits وواحدة بحجم 8 Bits.
- وحدة مؤقت مراقبة Watchdog Timer WDT.
- مجال واسع لجهد التشغيل (2V إلى 5،5V) [7].

وهذا جدول مقارنة بين المتحكم PIC18F4550 وباقي المتحكمات المشابهة له في مجموعته :

جدول II-2- جدول مقارنة لبعض متحكمات العائلة PIC18F [7].

Device	Program Memory		Data Memory		I/O	10-Bit A/D (ch)	CCP/ECCP (PWM)	SPP	MSSP		EAUSART	Comparators	Timers 8/16-Bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)					SPI	Master I^2C™			
PIC18F2455	24K	12288	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F2550	32K	16384	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F4455	24K	12288	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3
PIC18F4550	32K	16384	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3

II-4-2-تعريف المتحكم PIC :

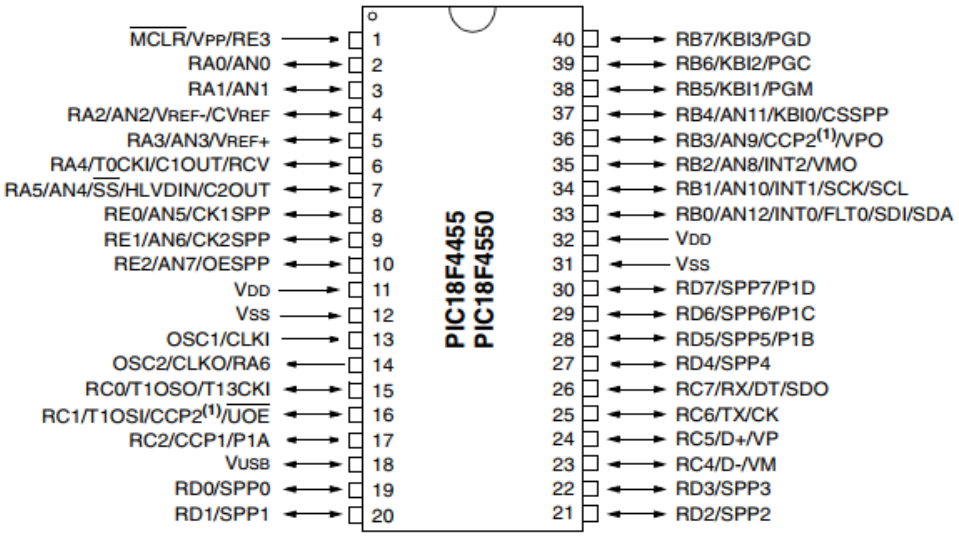
يتم تعريف المتحكم PIC ببساطة بواسطة رقمه، بحيث إن الرقمين الأوليين يحددان صنف المتحكم الدقيق، والحرف الذي يأتي بعدهما يعبر عن خصائص مميزة للمتحكم [5]:

- الحرف L : يعني أن المتحكم يعمل عند جهد مخفض.
- الحرف C : يعني أن المتحكم يحتوي على ذاكرة برنامج EPROM أو EEPROM.
- الحرف CR : يعني أن المتحكم يحتوي على ذاكرة برنامج من نوع ROM.
- الحرف F : يعني أن ذاكرة البرنامج من نوع Flash.

II-4-3-الشكل الخارجي للمتحكم PIC18F4550 :

المتحكم الدقيق PIC18F4550 هو دارة متكاملة في علبة صغيرة تسمى « DIL40 »، تملك هذه الأخيرة 40 طرفا على جهتين متقابلتين، 20 في كل جهة، الأطراف مرقمة من 1 إلى 40 [4]، حيث إن الطرف رقم 1 يوجد تحت النقطة المقعرة الموجودة في الطبقة العلوية للعلبة.

يمثل الشكل(II-2) أرجل المتحكم PIC18F4550.



الشكلII-2- أرجل Pins المتحكم PIC18F4550 [7].

II-4-4-أهم وظائف أطراف المتحكم PIC18F4550 :

RA ، RB ، RD وRE : هي وصلات للمنافذ A، B، C، D وE.

MCLR : مدخل جهد إعادة التعيين Reset Input.

Vpp : مدخل جهد البرمجة.

AN12-AN0 : مداخل تماثلية Analog Input.

V_{REF+} وV_{REF-} : الجهدان المرجعيان للمحول ADC.

CV_{REF} : المخرج المرجعي للمقارن التماثلي.

TOCKI : مدخل الساعة الخارجية External Clock للمؤقت Timer 0.

C1OUT : مخرج المقارن 1.

HLVDIN : مدخل لتحديد الجهد العالي والمنخفض.

C2OUT : مخرج المقارن 2.

Vdd : مدخل جهد التغذية.

Vss : المرجع الأرضي GND.

OSC1 : مدخل الهزاز الكريستال أو أي مصدر خارجي آخر للساعة.

OSC2 : مخرج الهزاز الكريستال.

CLK0 : في النمط RC هذا المخرج يخرج ¼ من التردد الداخل إلى OSC1.

T1OSO : مخرج المؤقت Timer 1.

T13CKI : مدخل ساعة خارجية للمؤقتات 1 و3.

T1OSI : مدخل المؤقت Timer1.

CCP2 : المخرج الثاني للوحدة CCP.

CCP1 : المخرج الأول للوحدة CCP.

CCP1 : القناة A، مخرج PWM للوحدة ECCP.

+D : الخط الموجب للناقل USB.

-D : الخط السالب للناقل USB.

Tx : منفذ الإرسال غير المتزامن لوحدة الاتصال التسلسليEUSART.

Rx : منفذ الإستقبال غير المتزامن لوحدة الاتصال التسلسلي EUSART.

CK : منفذ ساعة المزامنة للاتصال التسلسلي EUSART.

DT : منفذ البيانات للاتصال التسلسلي المتزامن EUSART.

P1A،P1B، P1C و P1D : مخرج PWM لوحدة CCP1 المعدلة.

INT2/INT1/INT0 : مداخل المقاطعة الخارجية.

SDO : مخرج البيانات للاتصال SPI.

SDI : مدخل البيانات للاتصال SPI.

SCK : مدخل ومخرج ساعة المزامنة لوضع الاتصال SPI.

SS : مدخل تحديد التابع Slave في البروتوكول SPI.

SDA : مدخل ومخرج البيانات في وضع الاتصال I2C.

SCL : مدخل ومخرج ساعة المزامنة لوضع الاتصال I2C.

CCP2 : المخرج الثاني لوحدة CCP.

KBI3-KBI0 : مداخل مقاطعة تعمل بتغير الحالة.

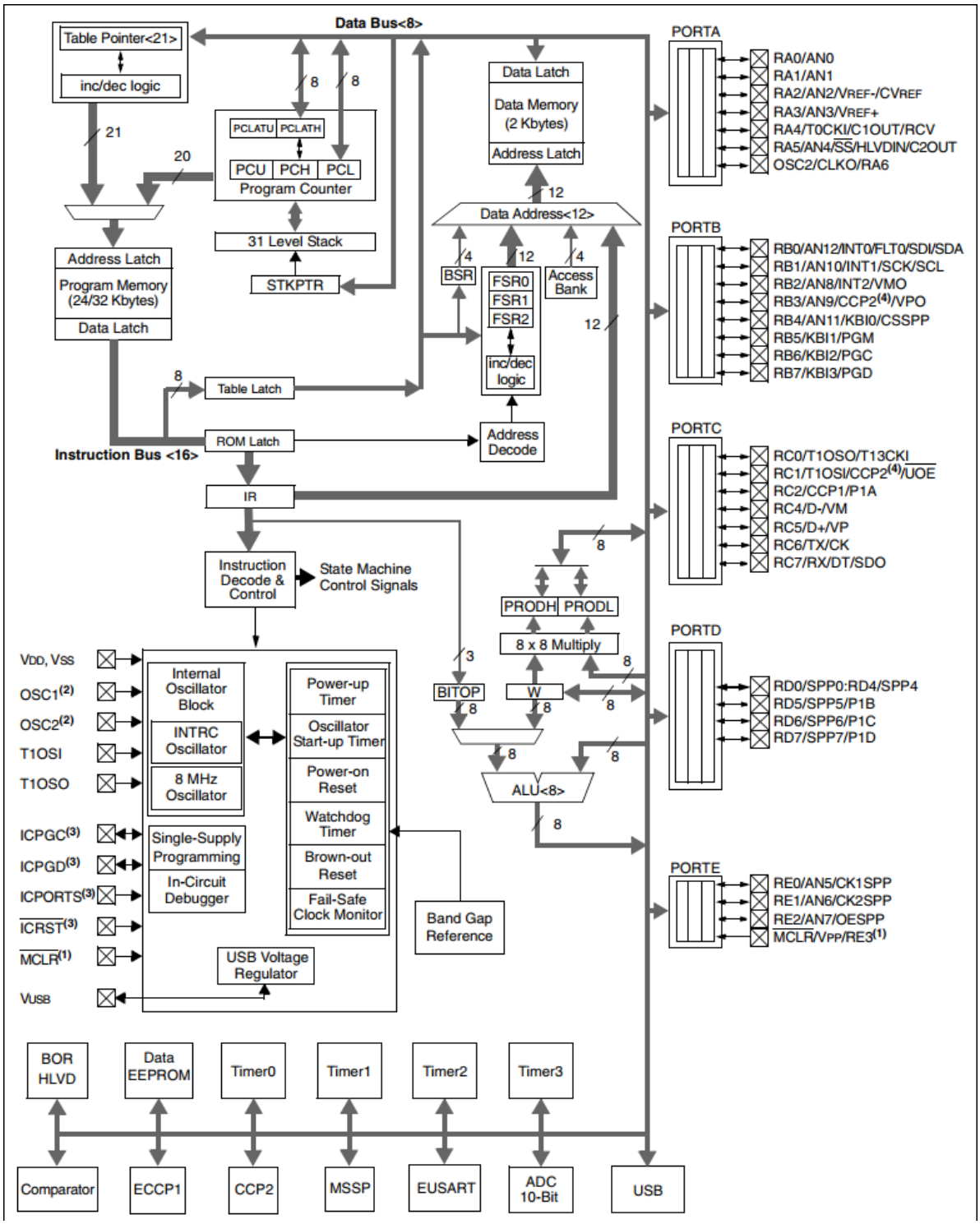
PGM : مدخل تفعيل وضع البرمجة بالجهد المنخفض Low Level ICSP.

PGC : مدخل ساعة المزامنة لوضع البرمجة ICSP.

PGD : مدخل ومخرج البيانات لوضع البرمجة ICSP [7].

II-4-5- المخطط الداخلي لبنية المتحكم الدقيق PIC18F4550 :

يمثل الشكل(II-3) البنية الداخلية للمتحكم PIC18F4550 والتي توضح مكوناته:



الشكلII-3- البنية الداخلية للمتحكم PIC18F4550 [7].

II-5-الذاكرات Memories :

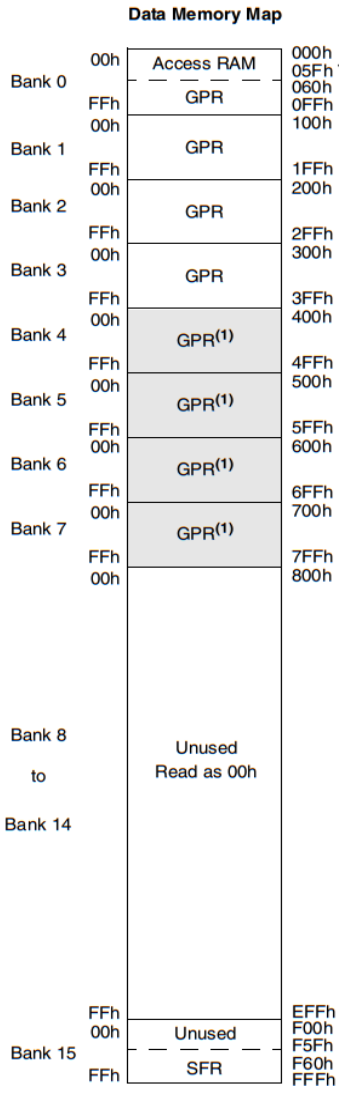
يوجد ثلاثة أنواع من الذاكرات في المتحكم PIC18F [7]:

- ذاكرة البرنامج Program Memory.
- ذاكرة البيانات المؤقتة Data RAM.
- ذاكرة البيانات الدائمة Data EEPROM.

كما هو حال الأجهزة ذات معمارية هارفرد فإن البيانات والبرنامج يستخدمان خطي بيانات منفصلين.

II-5-1-ذاكرة البيانات المؤقتة Data Memory :

الذاكرة العشوائية الموجودة في المتحكمات PIC18F هي من نوع Static RAM كل سجل في الذاكرة لديه عنوان بـ 12 Bits مساحة الذاكرة مقسمة إلى 16 قسما أو بنكا(Bank) حجم كل قسم 256 Bytes. المساحة المستخدمة من الذاكرة في العمل العادي هي 8 بنوك (0-7) وهي بمساحة 2048 Bytes. يمثل الشكل(II-4) تقسيم ذاكرة البيانات SRAM. هذه الأقسام الـ 16 التي تحتويها الذاكرة تعتبر سجلات على نوعين؛ الأول لأغراض عامة General Purpose Registers (GPRs) والثاني سجلات الوظائف الخاصة Special Function Registers (SFRs)[7].



الشكل II-4- تقسيم ذاكرة البيانات SRAM[7].

سجلات SFRs تستعمل للتحكم وإعطاء حالة المتحكم ووظائف الملحقات، سجلات GPRs تستعمل في التخزين. أيُّ قراءة للأماكن غير المستعملة من الذاكرة تَكُون0 [7].

الأقسام من 4 إلى 7 تستعمل في عمليات المنفذ USB. في حالة عدم تفعيل وحدة الاتصال USB فإن هذه الأقسام تستعمل كباقي السجلات[7].

في حالة تفعيل وحدة الاتصال USB فإن هذه الأقسام تستعمل كحامل للبيانات BUFFER في عمليات النقل عبر المنفذ USB[7].

II-5-2-ذاكرة البرنامج (فلاش):

هذه الذاكرة قابلة للقراءة والكتابة والمسح خلال العمل بصفة عادية. القراءة من ذاكرة البرنامج تكون بالبايت عند كل أمر قراءة. الكتابة تكون بقسم كامل يتكون من 32 Bytes مرة واحدة عند كل أمر كتابة. عملية مسح الذاكرة تكون بمسح قسم كامل يتكون من 64 Bytes عند كل أمر مسح، لا يتاح للمستخدم خاصية المسح الكلي للذاكرة[7].

للقيام بالقراءة من ذاكرة البرنامج والكتابة عليها يوجد أمران يسمحان للمعالج بنقل البايتات Bytes بين ذاكرة البرنامج والذاكرة RAM؛ الأمر الخاص بالقراءة وهو TBLRD والأمر الخاص بالكتابة TBLWT (بلغة الأسمبلي)[7].

يتم إجراء العمليتين في نقل البيانات بين الذاكرتين بواسطة السجل TABLAT ذي 8 بتات.

تُسْتَخدم سجلات تحكم للتعامل مع الأمرين TBLRD و TBLWT؛ وهي تشمل[7] :

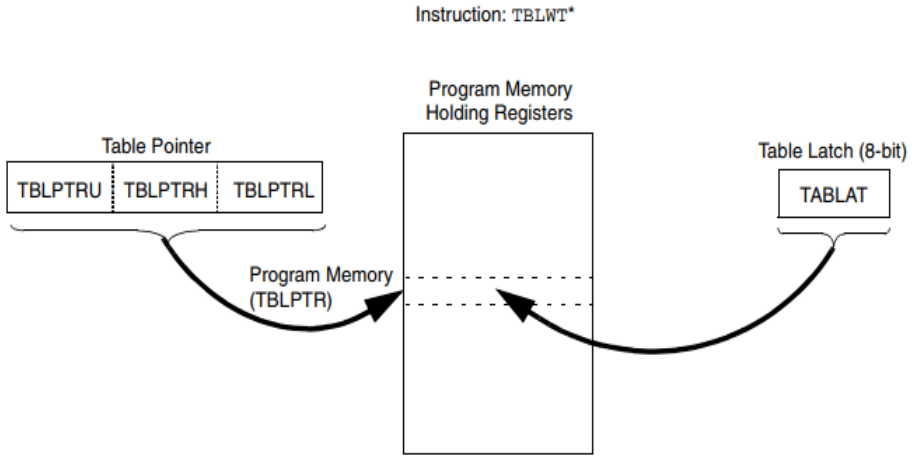
- السجل EECON1.
- السجل EECON2.
- السجل TABLAT.
- السجل TBLPTR.

السجل EECON1 مخصص للتحكم في إعدادات الوصول إلى الذاكرة. السجل EECON2 هو سجل تحكم. وهو ليس سجل فيزيائيا. يستعمل حصرا في عمليتي الكتابة والمسح. يعطي القيمة 0 عند القراءة[7].

السجل TABLAT (Table Latch) هو سجل بطول 8 Bits متواجد ضمن سجلات SFR في RAM. يستعمل لحمل البيانات أثناء اِنتقالها بين ذاكرة البرنامج والذاكرة RAM[7].

السجل TBLPTR (Table Pointer) هو سجل عنونة لذاكرة البرنامج. ينقسم TBLPTR إلى ثلاثة سجلات متواجدة ضمن سجلات SFR في RAM (TBLPTRU، TBLPTRH، TBLPTRL). هذه السجلات في مجملها تشكل سجل بطول 22 Bits[7].

يتم التعامل مع TBLPTR باِستخدام الأمرين TBLRD و TBLWT.



الشكل II-5- عملية الكتابة على ذاكرة البرنامج[7].

II-5-3-ذاكرة البيانات EEPROM :

هي ذاكرة منفصلة عن الذاكرة RAM والذاكرة فلاش، تستعمل في التخزين طويل المدى للبيانات. الذاكرة قابلة لقراءة والكتابة[7].

سجلات SFRs المستعملة في القراءة والكتابة من وعلى الذاكرة EEPROM هي [7]:

- EECON1.
- EECON2.
- EEDATA.
- EEADR.

تكون القراءة والكتابة بالبايت حيث إن السجل EEDATA يحمل ثمانية بتات من البيانات المخصصة الخاصة للقراءة أو الكتابة، والسجل EEADR يحمل عنوان المكان المحدد من الذاكرة EEPROM[7].

السجلان EECON1 وEECON2 يتحكمان في الوصول إلى الذاكرة EEPROM بنفس الآلية المعمول بها في الوصول إلى الذاكرة فلاش[7].

II-6-مصادر الساعة Clock Sources :

حتى يقوم المتحكم بعمله فإنه بحاجة إلى نبضات الساعة التي يمكن أن يكون مصدرها خارجي (كريستال، هزاز سيراميك، شبكةRC) أو داخلي (الهزاز الداخلي) وهي تزود المتحكم بالنبضات كمثابة القلب من الإنسان.[2]

II-6-1-الهزاز الخارجي External Oscillator :

وهذا الهزاز مبني داخل المتحكم وله قطبان يظهران للعالم الخارجي هما OSC1،OSC2. وهو يدعى هزاز خارجي رغم كونه مبنيا داخل المتحكم لأنه يعتمد على دارة خارجية تزوده بالنبضات عن طريق القطبينOSC1 ،OSC2 كما قلنا. الدارة الخارجية يمكن أن تكون كريستال أو هزاز سيراميك أوشبكة RC أو أي دارة أخرى حيث يمكن أن تكون متحكم آخر.[2]

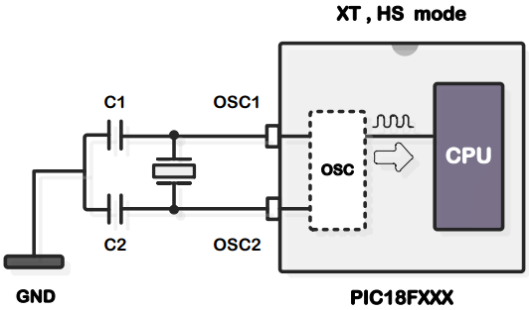
II-6-2-أنماط المصادر الخارجية للساعة External Clock وخصائصها :

الأنماط الخاصة بالمصادر الخارجية للساعة هي :

EC، ECIO، ECPLL، ECPIO، HS، HSPLL، XT، XTPLL.

في الأنماط HS، HSPL، X وXTPLL يتم توصيل أطراف هزاز الكريستال أو هزاز السيراميك بالطرفين OSC1 وOSC2[7].

يتم توصيل مكثفين مع الكريستال (أو مع هزاز السيراميك) كما هو موضح في الشكل (II-6)، وتتوقف قيم هذه المكثفات على قيمة الكريستال ويمكن الحصول على هذه القيم من الداتا شيت الخاصة بالميكروكنترولر[6].



الشكل II-6- توصيل هزاز الكريستال أو السيراميك بالمتحكم.[2]

الجدول التالي يوضح قيم المكثفات المستعملة بالنسبة لكل نمط عند استعمال هزاز الكريستال :

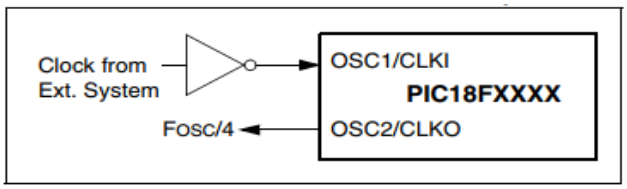
الجدول II-3- قيم المكثفات عند استعمال هزاز كريستال[7].

Osc Type	Crystal Freq	Typical Capacitor Values Tested:	
		C1	C2
XT	4 MHz	27 pF	27 pF
HS	4 MHz	27 pF	27 pF
	8 MHz	22 pF	22 pF
	20 MHz	15 pF	15 pF

هذه القيم للمكثفات غير ثابتة [7]. يمكن استعمال مكثفات سعتها محصورة بين 15 pF و33pF مع كل تردد وكل نمط [6].

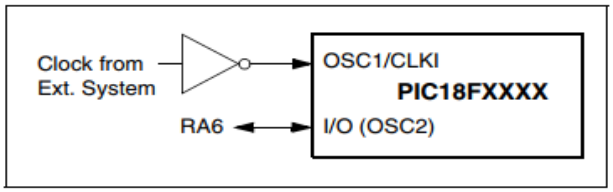
في الأنماط EC، ECIO، ECPLL، ECPIO يتم توصيل المصدر الخارجي للساعة بالطرف OSC1 فقط [7]. في هذه الأنماط المتحكم سيبدأ عمله مباشرة بعد وصول التغذية ولن يكون هناك أي تأخير – من أجل استقرار التردد – [2] بعد التشغيل أو إعادة التعيين أو الخروج من وضع النوم Sleep Mode[7].

في النمط EC وECPLL يُقَسم التردد الداخل على 4 ويكون الطرف OSC2 مخرجا له [7] (أي أن الطرف OSC2 سَيُخْرِج ربع التردد الداخل للطرف OSC1) كما يوضح الشكل(II–7).



الشكل II-7- مدخل الساعة في النمطين EC و ECPLL[7].

النمطان ECIO وECPIO مماثلان للنمطين EC وECPLL ما عدا أن الطرف OSC2 يستعمل كطرف عادي (مدخل/مخرج) للمنفذ A (RA6) [7] وهذا ما يبينه الشكل(II–8).



الشكل II-8- مدخل الساعة في النمطين ECIO و ECPIO[7].

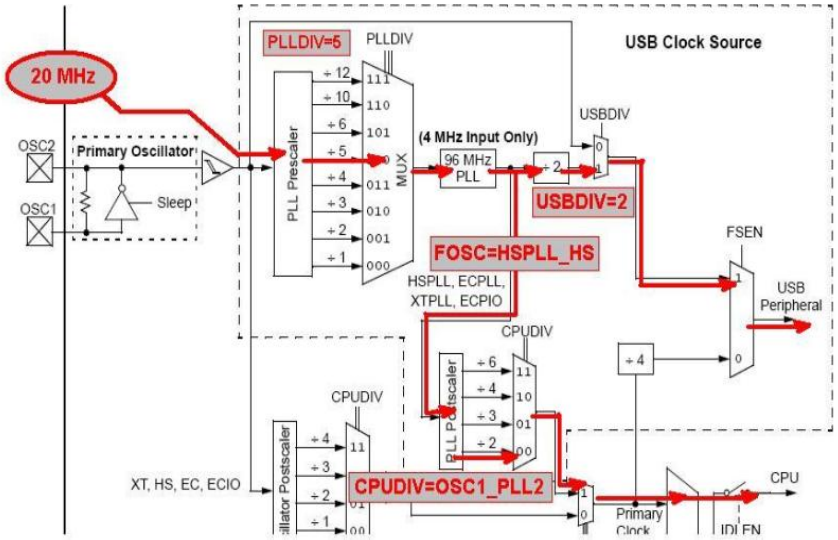
II–6–3–دارة مضاعفة التردد PLL :

يحتوي المتحكم PIC18F4550 على دارة PLL (Phase Locked Loop) وهي خاصة بتطبيقات الاتصال USB. وهي أيضا تستخدم في جعل المتحكم الدقيق يعمل كمصدر للساعة. يتم تفعيل PLL في الأنماط HSPLL، XTPLL، ECPLL، ECPIO للهزاز [7].

دارة PLL هي دارة جديدة في عائلة PIC18F، وتقدم لنا خيار مضاعفة التردد وذلك لزيادة سرعة جميع العمليات. هذه الدارة مهمة جدا لاسيما عند اِستخدام المتحكم من أجل تحقيق اِتصال مع المسرى USB [2].

عند استخدام المتحكم الدقيق للاِتصال من خلال USB يجب استعمال ساعة بتردد 6 ميغا هرتز للسرعة المنخفضة و48 Mhz للسرعة الكاملة. المشكلة هي أننا للسرعة الكاملة سنستخدم هزاز كريستال بتردد 48 Mhz، وهذا التردد عالي جدا يسبب تشويشا(ضوضاء) إضافة إلى كونه غير متاح بوفرة ومكلف. ولحل هذه المشكلة قامت شركة ميكروشيب بعمل دارة مضاعفة التردد PLL، والتي توفر التردد المطلوب والمستخدم للسرعة الكاملة عن طريق استخدام هزاز كريستال تتراوح قيمته من 4 ميغا هرتز إلى 48 Mhz. كما أن الدارة تقوم بمعالجة مشكلة الضوضاء [1].

في حالة ما إذا استخدمنا كريستال بتردد 4 Mhz ونريد العمل على السرعة القصوى للاِتصال USB، فإننا نختار سلم حساب Prescaler قيمته 1 فيبقى التردد نفسه 4 Mhz. يدخل هذا التردد على المضاعف لينتج تردد 96 Mhz وبعد القسمة على 2 يصبح التردد 48 Mhz. وإذا استخدمنا كريستال بتردد 20 Mhz عندئذ يجب أن يكون سلم الحساب هو 5 وهكذا. أما بالنسبة لتردد عمل المعالج CPU يمكن استخدام تردد مخرج PL أو استخدام مخرج الكريستال مباشرة [1] (الشكلII-9).



الشكل II-9- مخطط عمل دارة PLL [1].

II-6-4-قسم الهزاز الداخلي Internal Oscillator :

يحتوي المتحكم PIC18F4550 على هزاز داخلي يولد إشارتين مختلفتين لمصدر الساعة. إذا لم يستخدم الاتصال USB فإنه يمكن الاستغناء عن الهزاز الخارجي[7].

المخرج الرئيس للهزاز INTOSC يوفر تردد 8Mhz يمكن استخدامه مباشرة كتردد لساعة الجهاز. كما يمكن قسمة التردد الخارج من INTOSC للحصول على قيم مختلفة للتردد من 31Khz إلى 4Mhz[7].

المصدر الآخر للساعة هو هزاز RC الداخلي (INTRC) والذي يوفر تردد بقيمة 32 Khz. يتم تفعيل INTRC إذا تم تحديده كمصدر لساعة الجهاز. كما يتم تفعيله آليا عند تفعيل بعض الخصائص كمؤقت بداية التشغيل Power-up Timer ومؤقت المراقبة... [7]

II-6-5-أنماط الهزاز الداخلي :

في حالة اِستعمال الهزاز الداخلي كمصدر لساعة المتحكم الدقيق؛ فإنه يجب اِستخدام أحد أنماط الهزاز الخارجي كمصدر للساعة بالنسبة لوحدة الاتصال USB. يتم اختيار نمط الهزاز المناسب للUSB بما يتوافق مع الهزاز الداخلي[7].

يوجد أربعة أنماط متاحة وهي [7]:

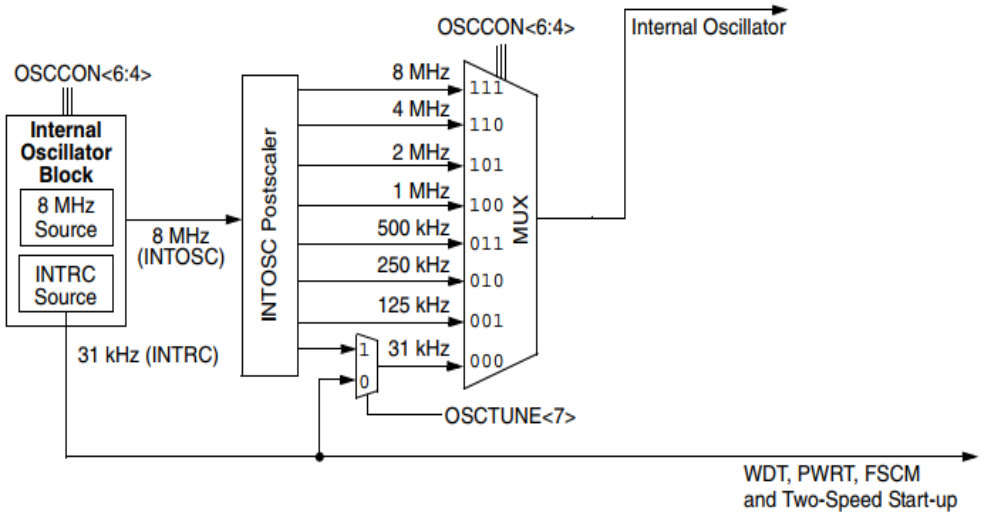
1/ النمط INTHS : نمط ساعة الوحدة USB هو HS.

2/ النمط INTXT : نمط ساعة الوحدة USB هو XT.

3/ النمط INTCKO : مصدر ساعة الوحدة USB يأتي من المدخل OSC1/CLKI، ويستعمل الطرف OSC2/CLKO كمخرج يوفر تردد يساوي ربع ¼ تردد دخل OSC1 (Fosc/4).

4/ النمط INTIO : هو نفسه النمط السابق (INTCKO) لكنَّ الطرف OSC2/CLKO/RA6 يستخدم كمدخل ومخرج للمنفذ A (RA6).

يمثل الشكل(II-10) مخطط عمل الهزاز الداخلي.



الشكل II-10- مخطط عمل الهزاز الداخلي[7].

II-7-منافذ الدخل والخرج I/O Ports :

يحتوي المتحكم الدقيق PIC18F4550 على 5 منافذ متاحة للمخارج والمداخل كل منفذ لديه 3 سجلات تتحكم في عمله وهي [7] :

- السجل TRIS : وهو سجل اِتجاه البيانات.
- السجل PORT : وهو سجل حالة أطراف المنفذ.
- السجل LAT : وهو سجل تثبيت المخرج.

كل بت في السجل TRISx يحدد ما إذا كان الطرف مدخلا أو مخرجا. حيث إن : 0 تعني مخرجا و 1 تعني مدخلا[8].

كل بت من السجل PORTx يعكس حالة المدخل المرتبط بالطرف المقابل له(إذا تم برمجة المنفذ كمدخل وإلا فإنه يكون هو نفسه السجل LATx)[8].

كل بت من السجل LATx يمثل حالة المخرج المرتبط بالطرف المقابل له(إذا تم برمجة المنفذ كمخرج وإلا فإنه لا يدل على حالة المنفذ)[8].

II-7-1- المنفذA :

هو منفذ ثنائي الإتجاه بطول 7بت. سجل اِتجاه البيانات الخاص به هو TRISA. سجل القراءة هو PORTA وسجل الكتابة هو LATA [7] وهو يحوي نفس معطيات السجل PORTA والكتابة إلى أي منهما تعني الكتابة إلى الآخر. لكن القراءة من LATA لا تعبر عن القيمة الحقيقية للأرجل وإنما القيمة المرسلة[2].

البتات 0، 1، 2، 3، 5 من هذه النافذة (المنفذ) تستخدم أيضا من أجل الدخل التماثلي. بعد تصفير المتحكم، تصبح هذه الأرجل مبرمجة كمداخل تماثلية أما RA4 وRA6 مهيأة كمداخل رقمية وأما لبرمجة المداخل التماثلية لتصبح رقمية فإن السجل المسؤول عن ذلك هو السجل ADCON1[2].

الرجل RA4 مرتبطة مع المؤقت0 (TOCKI)، RA6 يمكن أن تستخدم من أجل وظائف الإدخال والإخراج العامة أو كمدخل للساعة OSC2 أو كمخرج للساعة Fosc/4 [2].

II-7-2- المنفذ B :

هو منفذ ثنائي الإتجاه بطول 8 بت. السجلات الخاصة به هي :TRISB، PORTB، LATB. كل طرف من هذا المنفذ لديه مقاومة شدٍّ داخلية إلى الجهد V_{DD} Internal Pull-up Weak [7] والتي يُمْكِنُ أن تُمَكَّنَ عن طريق البت RBPU من السجل INTCON2[2].

مقاومات الشدِّ هذه سوف يُلْغَى تمكينها في حالة تهيئة أرجل البورت كمخارج. مقاومات الشدِّ ستسمح بوصل تجهيزات إدخال مثل لوحة المفاتيح والأزرار اللحظية بدون مقاومات شدٍّ خارجية[2].

أرجل البورت من RB4 وحتى RB7 يمكن أن تستخدم من أجل المقاطعة بتغير المدخل على الأقطاب من RB4 وحتى RB7. من أجل تمكين هذه المقاطعة فإن البت المسؤول عن ذلك هو RBIE أما العَلَم Flag الذي يدل على حدوث المقاطعة فهو RBIF وهذان البتان موجودان في السجل INTCON[2].

II-7-3- المنفذC :

هو منفذ مزدوج الإتجاه بطول 7 بت. السجلات الخاصة به هي TRISC، وPORTC، وLATC. الطرف RC3 غير مستعمل تماما. المنفذ مرتبط مع وحدات الإتصال التسلسلي والتي تتضمن EUSART وMSSP وUSB. جميع أطراف المنفذC عدا RC4 وRC5 تستعمل كمداخل من نمط قلاب شميت Schmitt Trigger[7].

الطرفان RC4 وRC5 مرتبطان بوحدة الإتصال USB. وعلى اِختلاف باقي أطراف المنفذ C في حالة استخدامه كمنفذ رقمي الطرفان RC4 وRC5 لا يعملان إلا كمداخل رقمية[7]. تستعمل الأطراف RC6/Tx وRC7/Rx للإتصال EUSART.

II-7-4- المنفذD :

هو منفذ مزدوج الاتجاه بطول 8بت. سجلاته هي TRISD، وPORTD، وLATD. كل أطرافه تستعمل كمداخل من نمط قلاب شميت. وكلها أيضًا لديها مقاومة شدٍّ داخلية[7].

يتم تفعيل مقاومات الشدّ عن طريق البت RDPU (البت 7 من السجل PORTE). تلغى هذه الخاصية عند إعداد المنفذ كمخرج رقمي أو إعداد أحد أطرافه كطرف لأحد الملحقات، كما تثبط مقاومة الشدّ بعد إعادة تشغيل الجهاز بعد إعادة التعيين Power-on Reset[7].

ثلاثة أطرف من المنفذD مرتبطة مع المخارج P1B وP1C وP1D لوحدة CCP المعدلة. يمكن أيضا إعداد المنفذD كمنفذ متواز لثمانية بتات 8بت(Streaming Parallel Port (SPP [7].

II-7-5- المنفذ E :

بالنسبة للمتحكم PIC18F4550 فإن المنفذ E بطول 4بت. ثلاثة أطراف من المنفذ وهي RE0 وRE1 وRE2 تستخدم كمداخل أو مخارج (مخارج من نمط قلاب شميت). كما يمكن استخدامها كمداخل تماثلية للمحول ADC. سجلات المنفذ هي TRISE وLATE وPORTE والذي يضم أيضا البت الخاص بتفعيل مقاومات الشدّ للمنفذD[7].

الطرف RE3 يعمل كمدخل فقط. يستخدم للقيام بعملية إعادة التعيين الخارجي MCLR، كما يستخدم كمدخل لجهد البرمجة Vpp [7].

II-8-المؤقت TIMER :

يحتوي المتحكم الدقيق PIC18F4550 على أربع وحدات للمؤقت؛ كل منها له خصائص [7]:

II-8-1-المؤقت TIMER0 :

- يعمل كمؤقت أو عداد بطول 8 Bits أو16 Bits.
- سجلاته قابلة للقراءة الكتابة (TMR0 في حالة 8بت وTMR0L : TMR0H في حالة 16 Bits).
- يستخدم سُلّم قسمة لإشارة الهزاز Prescaler بثمانية معاملات.
- يعمل على مصدر ساعة داخلي أو خارجي.
- حدوث مقاطعة عند وجود فائض Overflow.

II-8-2-المؤقت TIMER1 :

- يعمل كمؤقت أو عداد بطول 16 Bits.
- لديه سجلان (TMR0وTMR1) كل منهما بطول 8 Bits قابلان للقراءة والكتابة.
- يعمل على مصدر ساعة داخلي أو خارجي.

- يعمل كعداد متزامن Synchronous وغير متزامن Asynchronous.
- حدوث مقاطعة عند وجود فائض.
- يملك خاصية تصفير سجلاته عند تولد إشارة من وحدة المقارن CCP2.

II-8-3-المؤقت TIMER2 :

- يعمل كمؤقت على سجلين بطول 8 Bits ؛ أحدهما للحساب TMR0 والآخر للمقارنة PR2.
- كلا السجلان قابلان للقراءة والكتابة.
- سُلّم قسمة لإشارة الهزاز بثلاثة معاملات 1/1، 1/4، 1/16.
- سُلّم قسمة للإشارة الخارجة منه بمجال قيم من 1/1 إلى 1/16.
- حدوث مقاطعة عند تساوي السجل TMR2 مع السجل PR2.
- يستخدم في توليد إشارات PWM لوحدة CCP. كما يستخدم في الإتصال SPI ضمن وحدة MSSP.

II-8-4-المؤقت TIMER3 :

- يعمل كمؤقت أو كعداد بطول 16 Bits.
- لديه سجلان (TMR3Hو TMR3L) قابلان للقراءة والكتابة.
- تحديد مصدر الساعة (مصدر داخلي أو خارجي).
- حدوث مقاطعة عند وجود فائض.
- يعمل كعداد متزامن Synchronous وغير متزامن Asynchronous.
- يملك خاصية تصفير سجلاته عند تولد إشارة من وحدة المقارن CCP2.

II-9-إعادة التعيين Reset :

هو قيام المتحكم الدقيق بإيقاف جميع وظائفه ومسح جميع سجلاته؛ وهذا بسبب تَوَلُّد إشارة من أحد مصادر إعادة التعيين في المتحكم الدقيقPIC18F4550.

مصادر إعادة التعيين في المتحكم PIC18F4550 هي [7] :

- إعادة التعيين عند توصيل جهد التغذية Power-on Reset (POR).
- إعادة التعيين الخارجي بواسطة الطرف MCLR.
- إعادة التعيين بواسطة مؤقت المراقبة Watchdog Timer.
- إعادة التعيين عند انخفاض الجهد Brown-out Reset.
- إعادة التعيين عن طريق الأمر RESET.
- إعادة التعيين عند اِمتلاء المكدس Stack Full Reset.
- إعادة التعيين عند فراغ المكدس Stack Underflow Reset .

II-9-1-إعادة التعيين بواسطة الطرف MCLR :

وذلك بجعل الجهد الداخل إلى الطرف MCLR للمتحكم الدقيق في المستوى المنخفض Low Level. يمكن تثبيط هذه الخاصية عن طريق البت الخاص بالإعدادات MCLRE [7].

II-9-2-إعادة التعيين عند وصل التغذية POR :

هي عملية تحدث للجهاز عند بداية تصاعد جهد التغذية V_{DD} نحو العتبة. الهدف من هذه العملية هو السماح للمتحكم الدقيق بتهيئة وحداته قبل البدء في العمل[7].

II-9-3- إعادة التعيين عند اِنخفاض الجهد BOR :

تتم عملية إعادة التعيين هذه عند اِنخفاض الجهد V_{DD} إلى قيمة أقل من العتبة V_{BOR} لمدة أكبر من المدة T_{BOR} [7].

II-9-4-إعادة التعيين بواسطة مؤقت المراقبة WDT :

تحدث هذه العملية عند دخول المتحكم في حلقة مفرغة نتيجة حدوث خطأ أثناء تنفيذ البرنامج. حينها يقوم مؤقت المراقبة WDT بتصفير الجهاز بعد إكماله العد حسب الإعدادات المضبوطة.

II-10-المقاطعة Interrupt :

يوجد العديد من مصادر خدمة المقاطعة في المتحكم الدقيق PIC18F4550؛ تصنف هذه المقاطعات حسب مصدرها إلى نوعين [8]:

- مقاطعات خارجية المصدر وهي الأطراف RB0/INT0 وRB1/INT1 وRB2/INT2، إضافة إلى استعمال الأطراف RB4-RB7 كمصادر للمقاطعة بتغير قيمتها المنطقية.
- مقاطعات داخلية المصدر وهي المقاطعات الخاصة بالمتحكم والمقاطعات الخاصة بالوحدات الملحقة بالمعالج Peripherals.

يوجد عشرة سجلات تستعمل للتحكم في المقاطعات وهي : RCON، INTCON، INTCON2، INTCON3، PIR1، PIR2، PIE1، PIE2، IPR1 وIPR2 [7].

كل مقاطعة لديها ثلاثة بتات للتحكم في عمليتها، هذه البتات هي[7] :

- بت علم المقاطعة Interrupt Flag bit البت الخاص بالإعلام بحدوث المقاطعة.
- بت التفعيل Interrupt Enable bit وهو البت الخاص بتفعيل المقاطعة وبالتالي تفعيل انتقال المعالج من تنفيذ البرنامج الرئيسي إلى تنفيذ برنامج المقاطعة الفرعي عند حدوثها.
- بت تحديد الأولوية Interrupt Priority bit وهو البت الخاص بتحديد ما إذا كانت هذه المقاطعة ذات أولوية عالية High Priority أو أولوية منخفضة Low Priority.

II-10-1-السجلات INTCON :

هي ثلاثة سجلات INTCON، INTCON2 وINTCON3. كل منها قابل للقراءة والكتابة. تحتوي على بتات تفعيل وبتات أعلام وبتات الأولوية لبعض المقاطعات. كما يحتوي على بتات حجب المقاطعات [7].

II-10-2-السجلات PIR:

وهي تشمل السجلين PIR1 وPIR2، تحتوي على البتات الخاصة بأعلام مقاطعات الوحدات الملحقة [7].

II-10-3-السجلات PIE:

وهي تشمل السجلين PIE1 وPIE2، تحتوي على بتات تفعيل مقاطعات الوحدات الملحقة[7].

II-10-4-السجلات IPR:

وتشمل السجلين IPR1 وIPR2، تحتوي على بتات تحديد أولوية مقاطعات الوحدات الملحقة. لِاستعمال هذين السجلين يجب تفعيل إمكانية تحديد أولوية المقاطعة عن طريق البت IPEN من السجل RCON [7].

II-10-5-السجل RCON:

يحتوي على بتات أعلام؛ تُسْتَعْمَلُ لتحديد سبب(مصدر) آخِرِ عملية تصفير أو سبب عودة الجهاز للعمل بعد أن كان في وضع النوم Sleep Mode أو وضع العُطل Idle Mode [7].

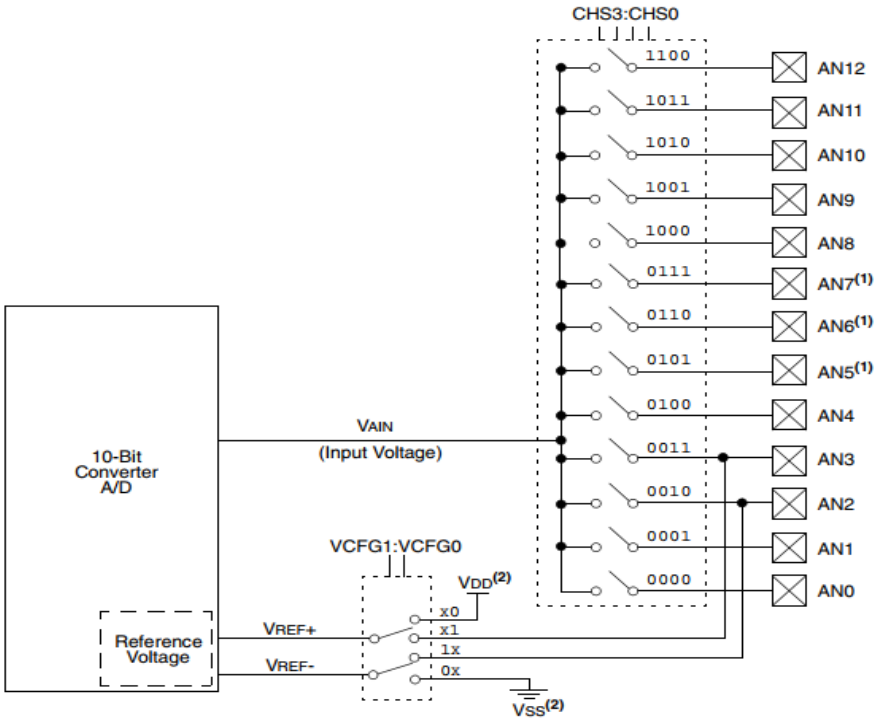
II-11-المحول التماثلي – الرقمي Analog To Digital Convertor :

هو أحد الوحدات الملحقة المتواجدة داخل المتحكم الدقيق. يعمل على تحويل الإشارات التماثلية إلى إشارات رقمية ليتم معالجتها على مستوى معالج المتحكم الدقيق.

تعتمد المتحكمات PIC على تقنية التقريب المتتالي Successive approximation حيث تتصف هذه التقنية بالسرعة الكبيرة والتصميم العالي والكلفة المنخفضة[9].

تملك وحدة المحول التماثلي الرقمي للمتحكم PIC18F4550 13 مدخلا للإشارات التماثلية. يتم تحويل الإشارة الداخلة إلى عدد رقمي بطول 10بتات. تعمل وحدة المحول على 5 سجلات هي : ADRESL و ADRESH و ADCON0 و ADCON1 و ADCON2[7].

يمثل الشكل(II-11) مخطط لعمل المحول ADC.



الشكلII-11- مخطط عملي للمحول[7].

II-11-1-خوارزمية عملية التحويل :

1/ إعداد وحدة المحول ADC :

- إعداد الأطراف التماثلية والجهد المرجعي والمداخل/المخارج الرقمية.
- تحديد أحد مداخل الإشارة التماثلية من أحد مداخل قنوات المحول من خلال السجل ADCON0.

- تحديد زمن التحصيل Acquisition Time من السجل ADCON2.
- تحديد مصدر ساعة المحول من السجل ADCON2.
- تشغيل وحدة المحول ADC من السجل ADCON0.

2/ إعداد مقاطعة المحول ADC :

- مسح بت علم المقاطعة ADIF.
- رفع بت تفعيل مقاطعة المحول ADIE.
- رفع بت التفعيل العام للمقاطعات.

3/ اِنتظار مرور زمن التحصيل.

4/ بدء التحويل :

- رفع البت GO/DONE من السجل ADCON0.

5/اِنتظار اِنتهاء التحويل وذلك بمسح البت GO/DONE تلقائيًا أو عند حدوث المقاطعة بارتفاع البتADIF.

6/ قراءة النتيجة من السجلين(ADRESH : ADRESL) [7].

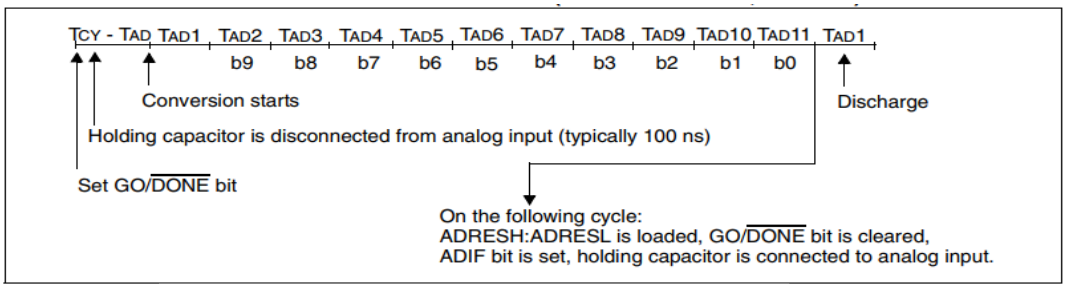
II-11-2-آلية عمل المحول ADC :

أولا تتم تهيئة نظام المحول ADC برمجيا من خلال سجلات التحكم ADCON0 ADCON1 وADCON2. حيث يتم إعداد الأطراف التماثلية للمحول وتحديد القناة التي سيتم قراءة الإشارة التماثلية منها وتحديد مصدر نبضات ساعة المحول[9].

تبدأ عملية الشحن وتستمر فترة زمنية هي زمن التحصيل Acquisition Time (وهو الزمن الضروري لشحن مكثف المسك إلى مستوى الجهد الخارجي المراد تحويله إلى رقمي) [9].

بعد انقضاء زمن التحصيل يفصل المكثف عن الجهد التماثلي لتبدأ عملية التحويل التماثلي/الرقمي الفعلية برفع البت GO/DONE. تدوم هذه العملية زمنا هو زمن التحويل Conversion Time الذي يساوي تقريبا $11T_{AD}$؛ حيث T_{AD} هو زمن تحويل كل خانة أوكل بت. يمثل الشكل (II–12) دورة التحويل للمحول ADC[9].

بعد إتمام عملية التحويل يتم مسح البت GO/DONE تلقائيًا ورفع البت ADIF[7].



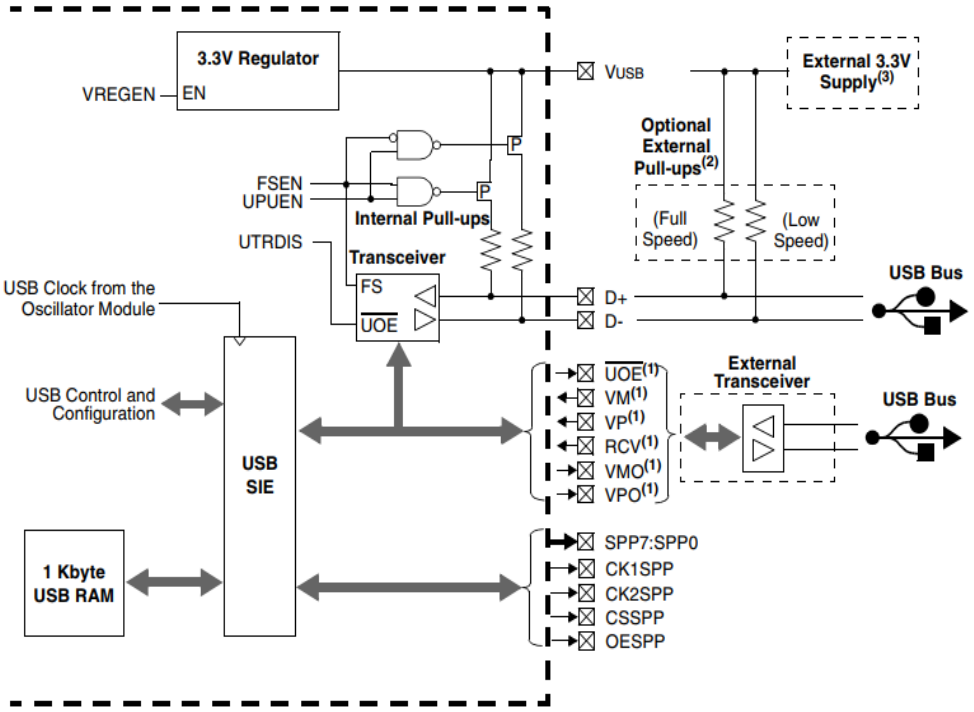
الشكلII-12- دورة التحويل للمحول ADC[7].

II–12–وحدة الإتصال USB :

يحتوي المتحكم PIC18F4550 على وحدة داخلية لإجراء الإتصال USB بالسرعة العالية أو بالسرعة المنخفضة. يسمى مركز هذه الوحدة بمحرك واجهة الإتصال التسلسلي (SIE) Serial Engine Interface. تسمح هذه الوحدة بإجراء إتصال بين أي مضيف USB والمتحكم PIC [7].

يمكن لهذه الوحدة الإتصال المباشر عبر المنفذ USB باستعمال الناقل الداخلي. أو يمكنها الإتصال غير المباشر عبر ناقل خارجي. يُستعمَل مُنظم الجهد الداخلي إلى3.3V Internal Regulator.3V3، لتغذية الناقل الداخلي أثناء عمل المتحكم بالجهد 5 فولت. ذلك أنَّ البروتوكول USB يعمل على جهد 3.3V [7].

البنية الداخلية لوحدة الاتصال USB يبينها الشكل(II–13).



الشكلII-13- وحدة الاتصال USB[7].

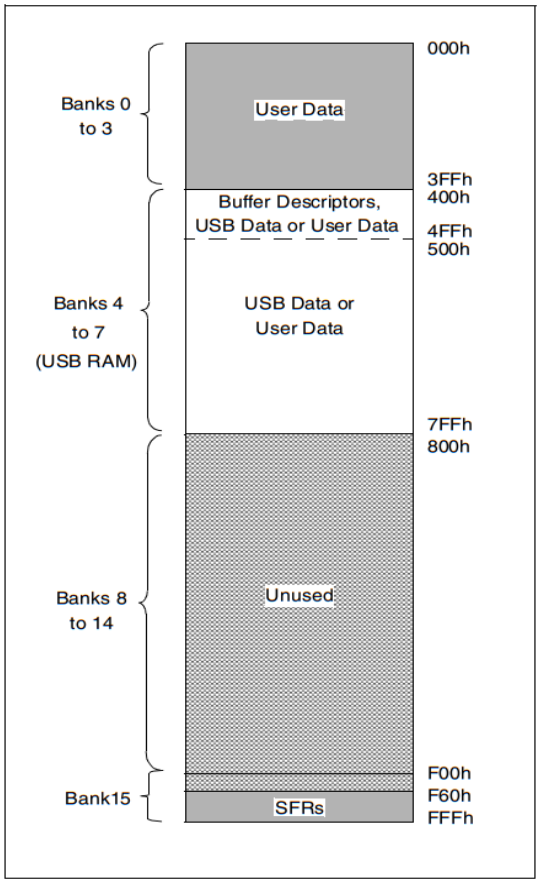
يتم إدارة وإعداد عمليات وحدة USB فعليًا من خلال 22 سجل[7] :

- سجل التحكم UCON.
- سجل الإعدادات UCFG.
- سجل حالة النقل USTAT.
- سجل عنوان الجهاز UADDR.
- سجلا رقم الإطار (UFRMH : UFRML).
- سجلات تفعيل نقطة الإتصال من السجل 0 إلى السجل 15 (UEPn).

II-12-1-الذاكرة USB RAM :

تنتقل البيانات أثناء الإتصال USB بين قلب المتحكم وSIE عبر مساحة من الذاكرة تسمى بـUSB RAM. تتواجد هذه المساحة ضمن ذاكرة البيانات المؤقتة RAM؛ حيث تشمل البنوك من 4 إلى 7 بمساحة 1Kbytes [7].

البنك 4 مخصص لحفظ بيانات التحكم لنقطة الإتصال. بينما تخصص المساحة الباقية من البنك 5 إلى 7 للبيانات المنقولة عبر USB. الشكل(II-14) يوضح تقسيم ذاكرة USB RAM[7].



الشكلII-14- الذاكرة USB RAM[7]

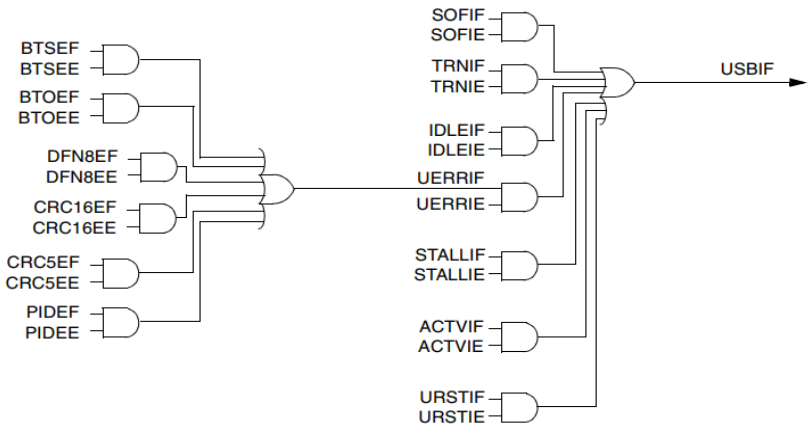
II-12-2-إعدادات الهزاز في الاتصال USB :

سبق وأن تحدثنا عن هذا الأمر بالتفصيل في شرحنا لدارة مضاعفة التردد PLL.

II-12-3-مقاطعة USB :

يمكن لوحدة USB أن تُجْرَى بداخلها عدة مقاطعات متعددة المصادر. تقوم وحدة USB بإدارة هذه المقاطعات وتجميعها لتَخرُج في النهاية مقاطعة واحدة USBIF موجهة لوحدة مقاطعات المتحكم[7].

يمثل الشكل(II-15) تجميع مقاطعات USB في مقاطعة واحدة.



الشكلII-15- مقاطعات وحدة USB [7].

II-12-4-الواصف USB Descriptor :

عندما يتم توصيل جهاز جديد بالناقل USB للمضيف Host فإن هذا الأخير سيحتاج إلى معرفة معلومات حول هذا الجهاز[2].

عند اِتصال المتحكم الدقيق مع الحاسوب الذي يعمل كمضيف يجب على المتحكم تقديم تعريف بنفسه للمضيف. فلو أن عملية التعريف هذه لم تتم سَيَحدث خطأ يُسبب اِنقطاع الاِتصال بين الطرفين.

يخضع صنف من الأجهزة المرتبطة عبر المنفذ USB للحاسوب لبروتوكول يسمى Human Interface Device (HID)، والمتحكم الدقيق ضمن هذا الصنف من الأجهزة[1].

II-13-الخاتمة :

المتحكم الدقيق يحتوي بداخله على جميع مكونات الحاسوب لذا يمكن اِعتبارُهُ كحاسوب مُصغّر يعمل ضمن شروط معينة؛ فهو يضم بداخله معالج دقيق، ومجموعة من الوحدات الملحقة إضافة إلى واجهات اِتصال مختلفة تمكنه من التفاعل مع العالم الخارجي بسهولة.

الفصل الثالث: تقنيات الاِتصال التسلسلي

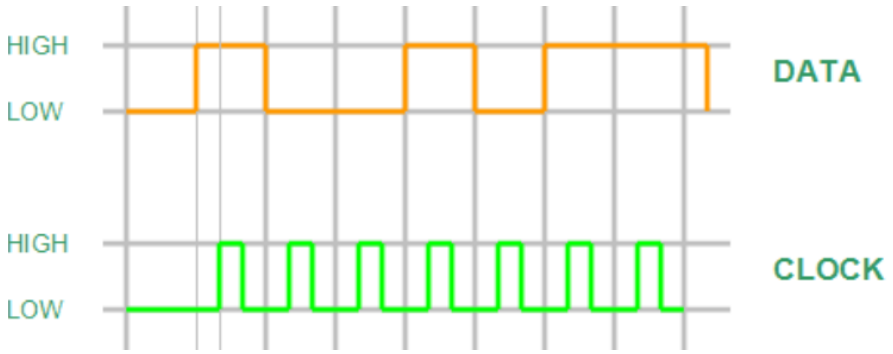
III-1-المقدمة:

في زمننا هذا تعتمد أغلب إن لم نقل كل الأجهزة على تقنيات الإتصال التسلسلي Serial Communication، هذا النوع من الإتصال الذي جاء ليحل محل الإتصال المتوازي Parallel Communication. وقد تعددت أنواع البروتوكولات المستخدمة لتقنية الإتصال التسلسلي نذكر منها: I2C، SPI، RS232 وUSB.

في هذا الفصل سنتحدث عن البروتوكولين RS-232 وUSB بإعتبارهما أهم البروتكولات التسلسلية.

III-2-النقل التسلسلي المتزامن:

تنتقل البيانات بت وراء بت في خط النقل ويتم إضافة خط لإشارة ساعة المزامنة The Clock Signal؛ هذه الإشارة هي التي تحدد نهاية كل بت وبداية البت الذي يليه. يتم نقل البيانات عند الحافة الصاعدة لنبضة إشارة الساعةّ Rising Edge [1]. الشكل(III-1) يمثل رسم بياني للنقل التسلسلي المتزامن.

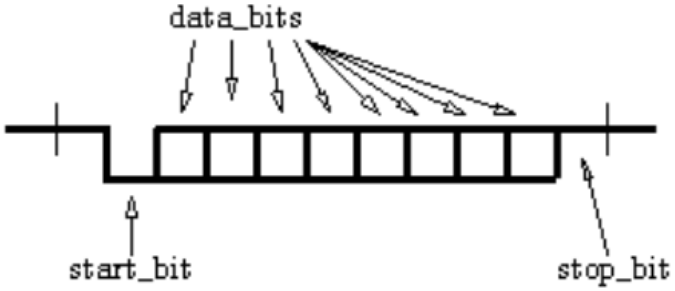


الشكلIII-1- النقل التسلسلي المتزامن للبيانات [1].

III-3-النقل غير المتزامن:

يتم فيه نقل البيانات بدون الإستعانة بساعة مزامنة؛ حيث إنّ كلّا من المرسل والمستقبل يتفقان على بنية محددة لإطار البيانات Data Frame وسرعة تدفق محددة Baud Rate[1].

الشكل(III-2) يمثل رسم بياني للنقل التسلسلي الغير المتزامن.



الشكل III-2- النقل التسلسلي الغير متزامن للبيانات[1].

III-4-بروتوكول الاتصال التسلسلي Recommended Standard RS-232:

III-4-1-تعريف:

المعيار RS-232 ويسمى أيضاً EIA-232-D هو معيار لنقل البيانات تسلسليا في إتجاهين وبنمط غير متزامن Asynchronous[2]. صُمم من طرف شركة EIA الأمريكية عام 1962 للعمل على ربط جهاز نهاية البيانات Data Terminal Equipment (DTE) كالحاسوب مع جهاز بيانات الإتصال(Data Communication Equipment (DCE مثل المودم. كما يمكن ربط جهازين من نوع DTE مباشرة[3].

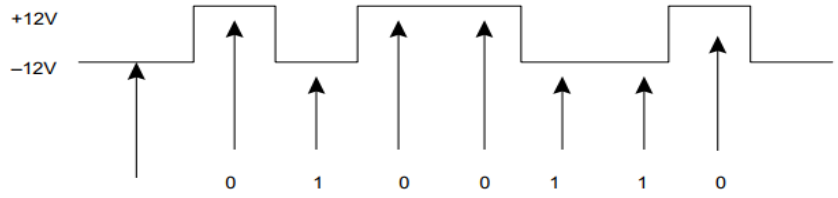
يعمل البروتوكول وفق أنماط مختلفة يستخدم فيها عدد مختلف من خطوط الإتصال؛ لكنّ أهم 3 خطوط والتي لا يمكن أن يُجْرَى أي إتصال بدونها هي خط الإرسال Tx وخط الإستقبال Rx والخط الأرضي GND.

III-4-2-خصائص المعيار RS-232:

يحدد المعيار RS-232 المقاييس الفيزيائية والوظيفية المعمول بها في تقنية الإتصال التسلسلي هذه.

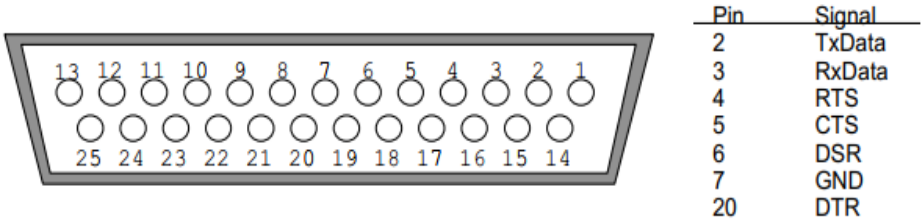
III-4-3-الخصائص الفيزيائية:

تشمل الخصائص الكهربائية وهي مجال الجهد الذي يعمل عليه البروتوكول RS-232؛ حيث أن القيمة المنطقية 1 يقابلها مجال جهد سالب من 3V- إلى 25V-، والقيمة القياسية هي 12V-. بينما يقابل القيمة المنطقية 0 المجال الموجب 3V إلى 25V، والقيمة القياسية هي 12V. أي جهد بين 3V- و3V لا يؤخذ بعين الاعتبار. يبين الشكل(III-3)مستويات الجهد المذكورة آنفا[4].



الشكلIII-3- مستويات الجهد في البروتوكول RS-232[4].

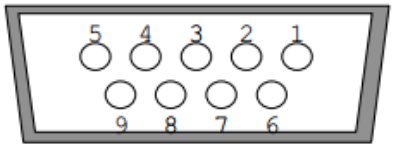
إضافة إلى الخصائص الكهربائية تشمل المقاييس الفيزيائية أيضًا المُوَصِّلات المستعملة في الإتصال. حيث يوجد الموصل DB25 الذي يحتوي على 25 طرف ويسمح بإستعمال كل الخصائص المتاحة للبروتوكول RS-232 [4]. الشكل(III-4) يوضح المُوَصِّل DB25 وأطرافه.



الشكلIII-4- المُوَصِّل DB25 للبروتوكول RS-232 [4].

بالنسبة للأجهزة الإلكترونية الصغيرة فإنها تحتاج إلى موصل صغير يفي بغرض الإتصال التسلسلي لذا يوجد الموصل DB9 [4].

الشكل(III-5) يوضح المُوَصِّل DB9 وأطرافه.



Pin	Signal
2	RxData
3	TxData
4	DTR
5	GND
6	DSR
7	RTS
8	CTS

الشكلIII-5- المُوَصِّل DB9 للبروتوكول RS-232 [4].

III-4-4-الخصائص الوظيفية للمعيار RS-232:

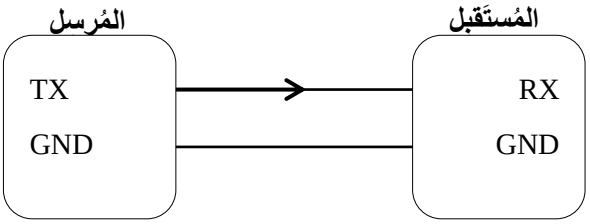
- يمكنه إرسال كلمة تتكون من 5 إلى 8 بت.
- تأطير البيانات بوضع بت للبداية Start بقيمة 0 منطقي ووضع 1 أو 1,5 أو 2 بت للنهاية Stop بقيمة 1 منطقي.
- 1 بت للتكافؤ Parity يضاف من أجل تحديد أخطاء النقل.
- نقل بسرعات تدفق مختلفة من 75 Bits/s إلى 115200 Bits/s.
- إمكانية الإتصال النصف ثنائي Half duplex والثنائي الكامل Full duplex[5].

III-4-5-أنواع الاتصال في البروتوكول RS-232:

III-4-5-1-الإتصال أحادي الإتجاه Simplex:

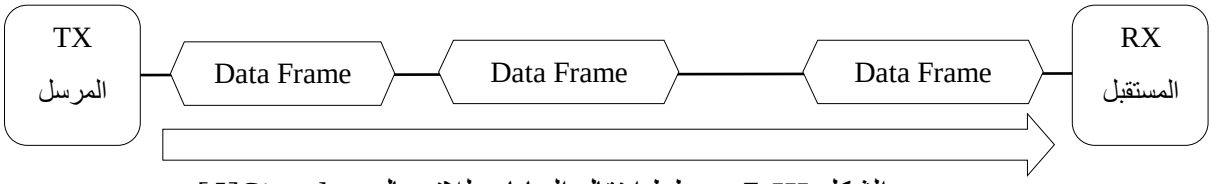
في هذا النمط يكون الإتصال بين المرسل والمستقبل فقط في إتجاه واحد؛ حيث يمكن لطرف واحد فقط أن يكون هو المرسل دائما والطرف الآخر يبقى مستقبلا[6].

الشكل (III-6) يوضح ربط لجهازين على نمط الإتصال أحادي الإتجاه:



الشكل III-6-الإتصال أحادي الإتجاه Simplex[5].

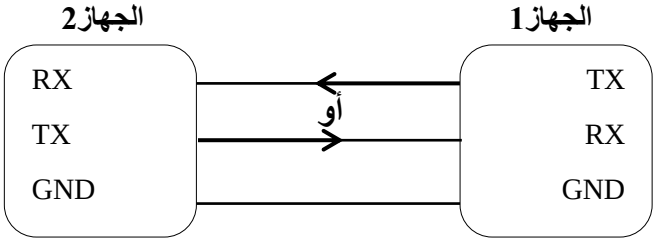
الشكل (III–6) يوضح مخططا بيانيا لانتقال البيانات أثناء الاتصال Simplex.



الشكل III-7-مخطط إنتقال البيانات للإتصال Simplex[5].

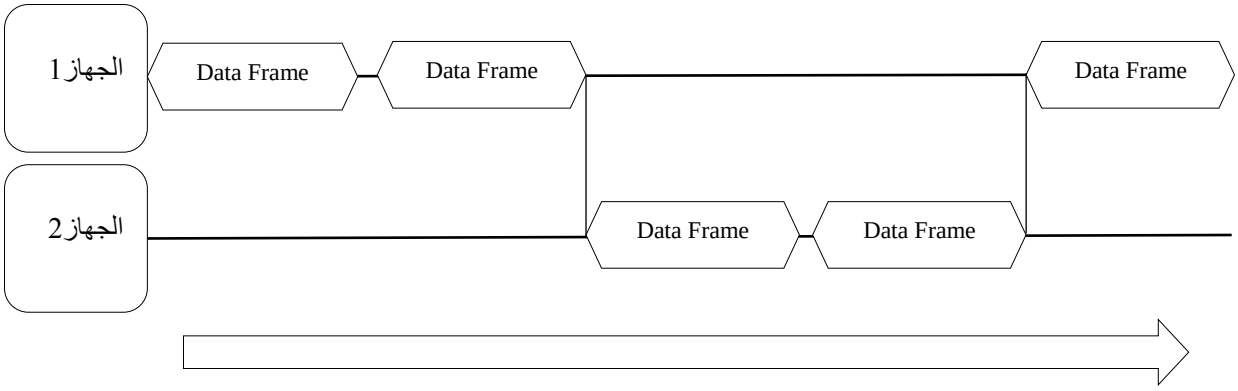
III–4–5–2–الإتصال النصف ثنائي Half duplex:

في هذا النوع يكون الإتصال بين المرسل والمستقبل بإتجاهين لكن على التناوب أي مرة بمرة؛ بمعنى أن كل طرف من طرفي الإتصال يمكنه الإرسال والإستقبال لكن ليس في نفس الوقت[6]. الشكل التالي يمثل طريقة ربط جهازين على النمط Half duplex:



الشكلIII-8-الإتصال النصف ثنائي Half duplex[5]

الشكل (III–9) يوضح مخططا بيانيا لانتقال البيانات أثناء الاتصال Half duplex.

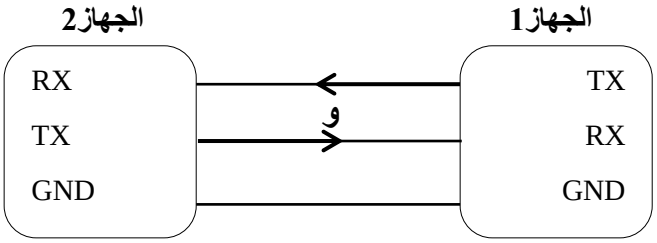


الشكلIII-9- مخطط إنتقال البيانات للإتصال Half duplex[5].

III-4-5-3-الإتصال الثنائي الكامل Full duplex:

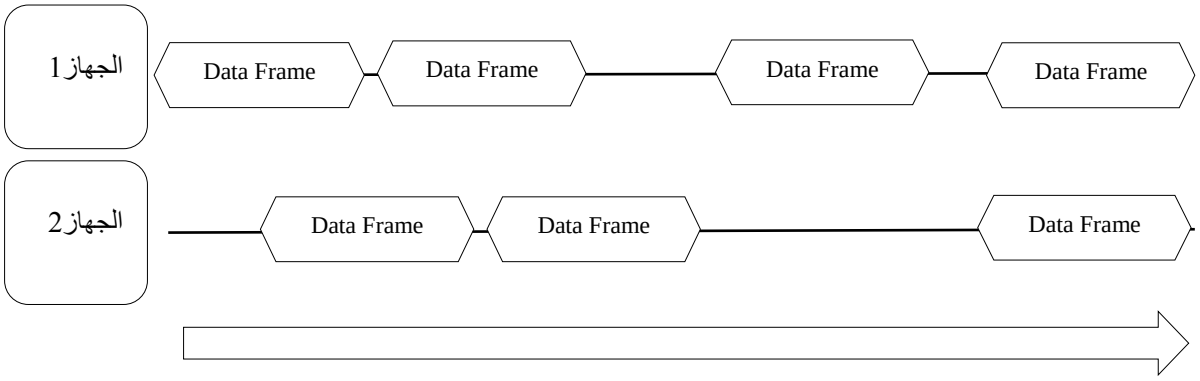
في هذا النوع يكون الإتصال بين الطرفين بشكل متوازي وآنيّ حيث يمكن لكليهما أن يرسل وأن يستقبل في نفس الوقت. هذا النوع من الإتصال غير متزامن نهائيا[6].

الشكل التالي لربط جهازين على النمط Full duplex:



الشكلIII-10-الإتصال الثنائي الكامل Full duplex[5].

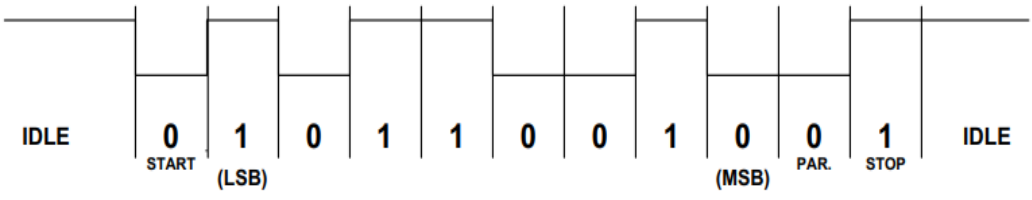
الشكل (III-11) يوضح مخططا بيانيا لانتقال البيانات أثناء الاتصال Full duplex.



الشكلIII-11- مخطط إنتقال البيانات للإتصال Full duplex[5].

III-4-6-شرح البروتوكول RS-232:

يوضح الشكل التالي مخططا بيانيا للبروتوكول RS-232:



الشكلIII-12- مخطط بياني للبروتوكول RS-232[7].

ينتقل كل بت وراء البت الذي يسبقه بمدة معينة تُحَدِّدُ سرعة تدفق البيانات. يكون خط النقل في وضع الراحة عند القيمة المنطقية '1'، ويقوم المرسل ببعث بت البداية لإعلام المستقبِل بقدوم حزمة بيانات إليه بعد هذا البت الذي يكون دائما بالقيمة المنطقية '0' [4].

الـ5 بتات أو 6 أو 7 أو 8 (الكلمة مكونة من 5 إلى 8 بتات) التي تَلِي بت البداية ترسل على شكل رمز أسكي ASCII متبوعة بِبت التكافؤ. في النهاية – بعد بت التكافؤ– يتم إرسال بت أو 1,5 بت أو 2 بت للتوقف لإعلام المستقبِل بنهاية إطار البيانات[4].

بت التكافؤ هو بت يضاف من قبل المرسل وعلى حسب الإتفاق مع المستقبل يكون هذا البت '0' أو '1'. تستعمل هذه التقنية لكشف أو تصحيح الخطأ الحادث أثناء عملية نقل البيانات. يوجد نوعان لبت التكافؤ[5]:

أ– التكافؤ الزوجي The Even Parity: يضاف في هذه الحالة بت التكافؤ ليكون عدد القيم المنطقية '1' في حزمة البيانات زوجيا.

مثال: حزمة البيانات 1010 0111 تحتوي على 5 قيم للواحد. في هذه الحالة يضاف بت تكافؤ بقيمة '1' ليكون عدد قيم الواحد في الحزمة 6 وهو عدد زوجي.

ب– التكافؤ الفردي The Odd Parity: يضاف بت التكافؤ ليكون عدد قيم الواحد في حزمة البيانات فرديا.

مثال: حزمة البيانات 1001 0010 تحتوي على 3 قيم للواحد وهذا عدد فردي وبالتالي فإن بت التكافؤ يضاف بقيمة '0'.

III–4–7–تقنية المصافحة Handshaking:

في حالة النقل العادية يجب على المستقبل أن يُتِمَّ قراءة البيانات المرسلة له قبل أن يَتِمَّ إرسال بيانات أخرى إليه. يقوم المستقبل بتخزين البيانات المرسَلة إليه في ذاكرة خاصة تسمى ذاكرة الإستقبال

Receiver Buffer. نظريا لا تتسع هذه الذاكرة سوى لرمز واحد One Character. يتم تفريغ ذاكرة الإستقبال بعد أن تتم قراءتها. وإذا تم إرسال رمز جديد قبل أن يتم تفريغها يحدث ضياع للبيانات السابقة للرمز الأول الذي حل محله الرمز الجديد [4].

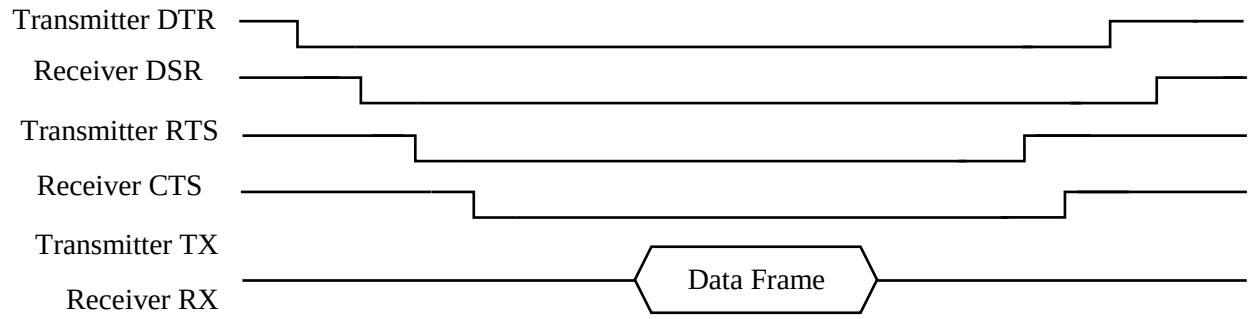
مثال: يقوم المرسل ببعث أربعة رموز متتالية إلى المستقبِل الذي يُتِم قراءة الرمزين الأولين على التوالي لكنه لم يُتِمّ قراءة الرمز الثالث فأدّى هذا لضياعه وحلول الرمز الرابع مكانه[4].

المصافحة البرمجية Software Handshaking تتضمن اِرسال رموز إعدادات خاصة. بينما في تقنية المصافحة الفيزيائية عن طريق العتاد يقوم المرسل بسؤال المستقبل ما إذا كان جاهزا لِإستقبال بيانات أم لا. إذا كانت ذاكرة الإستقبال فارغة يتم إعلام المرسل أن المستقبل جاهز لِإستقبال البيانات[4].

عند إرسال البيانات وتحميلها بعد ذلك على ذاكرة الإستقبال يتم إعلام المرسل بعدم بعث أي بيانات أخرى إلى حين قراءة محتويات ذاكرة الإستقبال. تعتمد تقنية المصافحة على الخطوط التالية:

- CTS: Clear To Send.
- RTS: Ready To Send.
- DTR: Data Terminal Ready.
- DSR: Data Set Ready[4].

الشكل (III-13) يمثل مخطط بياني لعملية نقل البيانات لتقنية المصافحة.



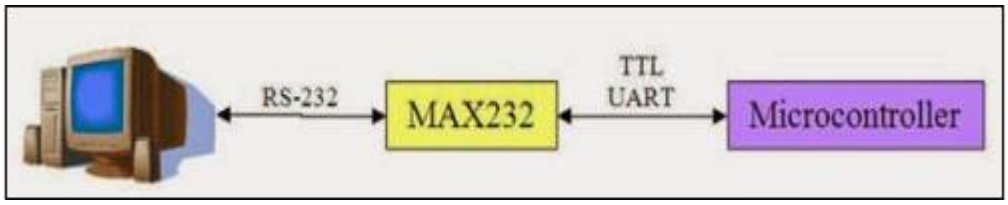
الشكلIII-13- مخطط بياني لتقنية المصافحة Handshaking[5].

III-4-8-بروتوكول الإتصال التسلسلي UART في المتحكمات الدقيقة:

تمتلك أغلب المتحكمات واجهة اِتصال تسلسلي USART، تعتمد هذه الواجهة على مستويات جهد حسب تقنية الـ TTL وهي 2V إلى 5V للقيمة المنطقية '1'، و0V إلى0.8V فولت للقيمة المنطقية '0'.

كما أسلفنا الذكر فإن المعيار RS232 يعمل على مجال جهد 3V- إلى 25V- (الجهد القياسي12V-) للقيمة المنطقية '1'، وعلى مجال 3 فولت إلى 25 فولت(الجهد القياسي 12 فولت) للقيمة المنطقية '0'. لذا وجب تعديل مستويات الجهود وضبطها للتوافق فيما بينها لإجراء اِتصال تسلسلي بين الحاسوب والمتحكم الدقيق[8].

هذا يمكن تحقيقه ببساطة بِاستخدام شريحة تدعى MAX232 المصنعة من قبل شركة MAXIM. هذه الشريحة تحتاج عددا من المكثفات الخارجية من أجل الشحن كي تستطيع توليد12V و12V- الذي يحتاجه المعيار RS232. الشكل(III-14) تمثيل لربط المتحكم مع الحاسوب بواسطة MAX232 [8].



الشكلIII-14- الدارة المدمجة MAX232 بين الحاسوب والمتحكم [9].

III-5-الناقل التسلسلي العام Universal Serial Bus USB:

III-5-1-تعريف:

هو ناقل بيانات وطاقة معتمد من النوع ركب وشغل Plug & Play يسمح بوصل أغلب الملحقات الطرفية والأجهزة المقترنة بالحاسوب لنقل المعلومات بينهما تسلسليا[10]، ويعد واحدًا من أكثر المنافذ استخدامًا وشيوعًا في أيامنا هذه. يعتبر واجهة تسلسلية عالية السرعة. هذه الواجهة

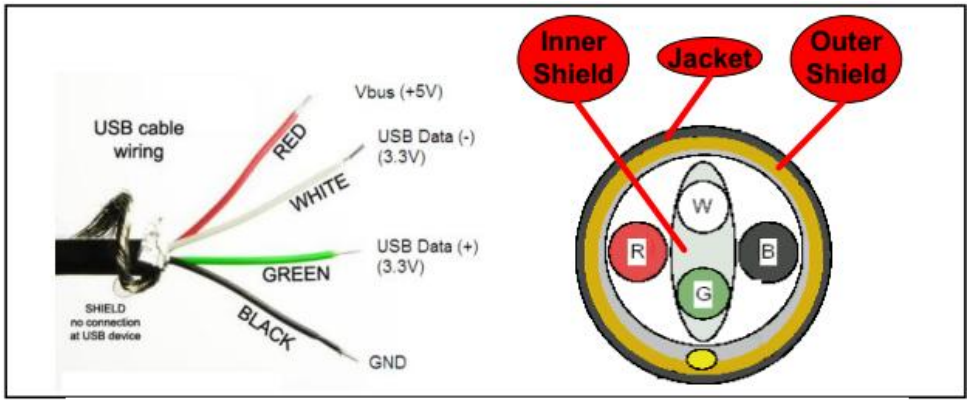
بإمكانها تزويد الطرف المتصل بها بالتغذية. التيار الأعظمي الذي يوفره منفذ USB هو 100 mA بجهد 5V [8].

قبل أن نتكلم عن بروتوكول USB يجب تعريف بعض المصطلحات الخاصة بتقنية الـUSB [8]:

- **المضيف Host:** هو الجهاز المسؤول على المسرى وفي حالتنا هو الحاسوب.
- **التجهيز Device:** وهو الطرف المتصل بالمضيف مثل المتحكم الدقيق.
- **المجمع Hub:** هو جهاز يتيح وصل أكثر من تجهيز كما يتحكم في الطاقة وتحويل سرعة الإتصال بين المضيف والتجهيز.
- **البوابة (المنفذ) Port:** فيزيائيا هي المُوَصِّل Connector، برمجيا هي موقع قابل للعنونة؛ أي أن البرمجيات تحمل على عاتقها مراقبة المنفذ والتحكم به من خلال عنوانه. الميزة الأساسية للمنافذ USB التي تميزها على منافذ أخرى مثل RS-232 أن منافذ المسرى USB جميعها موصولة بممر واحد إلى المضيف. أما في RS-232 فكل منفذ مستقل عن الآخر.
- **نقطة الاتصال Endpoint:** هي ذاكرة مؤقتة ضمن الجهاز المتصل يتم عنونتها بشكل خاص. البيانات المخزنة فيها عبارة عن البيانات المُستقبَلة والبيانات المراد إرسالها.
- **قناة الاتصال Pipe:** وهي عبارة عن وصلة منطقية تربط المضيف مع نقطة اِتصال في الجهاز. يقوم المضيف بإنشاء قنوات الإتصال عند وصل الجهاز مع المسرى العام ويقوم بإزالتها عند فصلها، وربما يُنشئ أو يحذف وصلات أثناء العمل حسب اللزوم.

III-5-2- أسلاك الناقل USB:

يحتوي خيط الناقل USB على مجموعة من الأسلاك محمية بأغلفة[11]. الشكل(III-15) يوضح صورة مقطعية للناقل USB.



الشكل III-15-الأسلاك داخل خيط الناقل USB [11].

تحت الغلاف العلوي Outer Shield يوجد سلكان؛ السلك V_{BUS} بالأحمر لجهد التغذية الموجب 5 فولت والسلك GND بالأسود للخط الأرضي. تحت الغلاف السفلي Inner Shield الذي يكون من الألمنيوم يوجد سلكان آخران ملتفان حول بعضهما؛ السلك الأبيض هو -D والسلك الأخضر هو +D [11].

يوفر الناقل USB ثلاث سرعات مختلفة لنقل البيانات؛ السرعة الكاملة Full Speed 12Mb/s ميغا بايت/ثانية والسرعة العالية High Speed 480 Mb/s والسرعة المنخفضة Low Speed 1,5 Mb/s [11].

بالنسبة للسرعتين العالية والكاملة فإنَّ أقصى طول لخيط الناقل USB هو 5 أمتار. أمّا بالنسبة للسّرعة المنخفضة فإنَّ أقصى طول للخيط هو 3 أمتار. عند هذه السرعة يمكن الاستغناء عن لف السلكين +D و-D [11].

السلك V_{BUS} يوفر جهد تغذية ثابت بقيمة محصورة بين4,4V و5,52V فولت. بينما تعمل أسلاك نقل البيانات +D و-D على جهد 3,3V فولت [11].

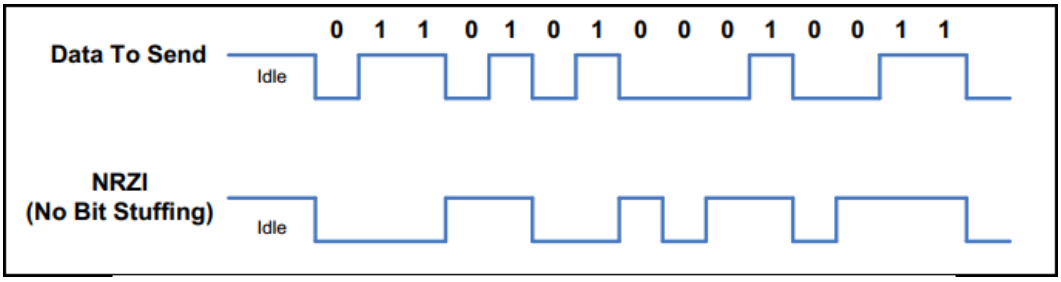
يستعمل الناقل USB النقل التفاضلي Differential Transmission وطريقة التشفير NRZI [11].

III-5-3-تشفير المعلومة:

يعتمد الناقل USB على طريقة NRZI في تشفير البيانات، والتي تعمل وفق المنهجية التالية:

الشكل(III-16) يمثل مخطط بياني لطريقة التشفير NRZI.

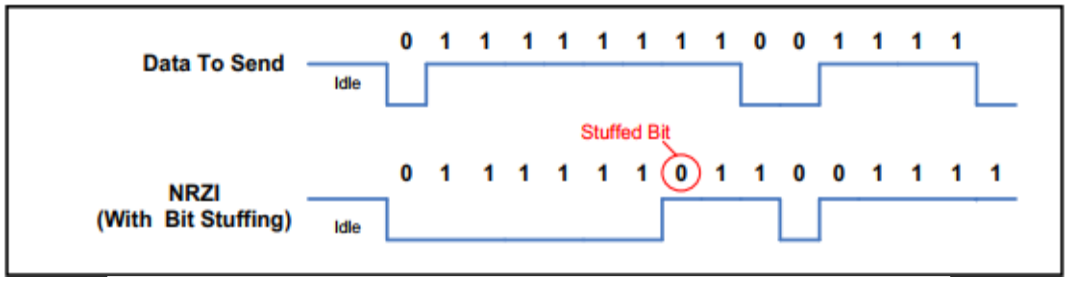
عند القيمة '0' في حزمة البيانات يَحدث اِنقلاب للإشارة عكس حالتها عند البت السابق، وعند القيمة'1' تحافظ الإشارة على حالتها كما كانت عند البت السابق[11].



الشكلIII-16- تشفير NRZI[11].

إحدى مساوئ هذه الطريقة هي أنَّ المستقبِل قد يفقد التزامن في حال تعاقب عدد كبير من البتات عند نفس المستوى المنطقي. لذلك يتم إضافة بت بقيمة '0' بعد كل ست قيم متتالية للواحد '1' [8]

يمثل الشكل (III-17) طريقة تصحيح التزامن.

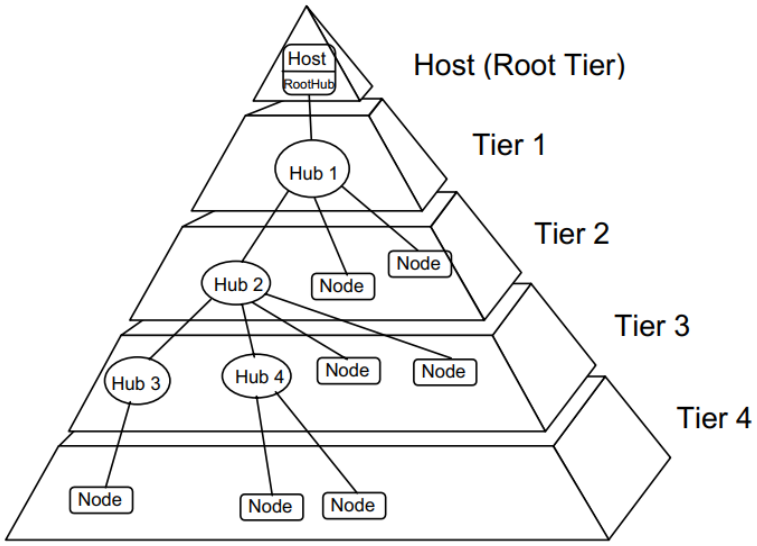


الشكلIII-17-إضافة بت '0' بعد 6 قيم متتالية للواحد [11].

III-5-4-التصميم الهندسي للخط USB:

يربط الناقل USB الأجهزة الطرفية المختلفة مع المضيف Host. يكون التصميم الهندسي للناقل على الشكل النجمي Start topology. يكون المجمِّع Hub في وسط النجمة. الإتصال بين طرفي كل خيط يكون وفق النمط نقطة إلى نقطة Point to point [12].

يوجد مضيف واحد في النظام USB، وهو المسؤول على توجيه وإدارة الإتصال مع الأجهزة المرتبطة بالخط. يمثل الشكل(III-18) التصميم الهندسي لشبكة USB[12].



الشكلIII-18-التصميم الهندسي لخط الناقل USB [12].

III-5-5-إصدارات البروتوكول USB:

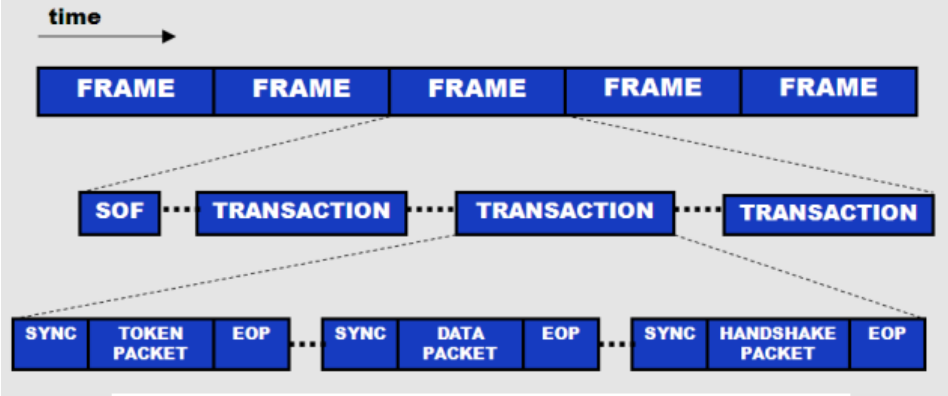
- **الإصدار USB1.1:** يدعم السرعتين المنخفضة والكاملة.
- **الإصدار USB2.0:** يدعم السرعات الثلاث المنخفضة والكاملة والعالية.
- **الإصدار USB3.0:** يدعم السرعة الفائقة Super Speed 4,8 Gb/s جيجا بايت/الثانية.

هذا الإصدار يتطلب وجود تجهيزات على اللوحة الأم للحاسوب فهو يحتوي على 9 أسلاك وهو غير مدعوم من قبل المتحكمات الدقيقة، وهو متوافق مع الإصدارات القديمة لكن لن يعمل بالسرعة الفائقة عند الإتصال بها[8].

III-5-6-إطار البيانات Data Frame في الناقل USB:

تنتقل البيانات في خط USB على شكل إطارات متتالية كل إطار يتكون من بداية Start Of Frame SOF متبوعة بواحدة أو أكثر من الناقلات وتسمى إصطلاحا صفقة Transaction [11].

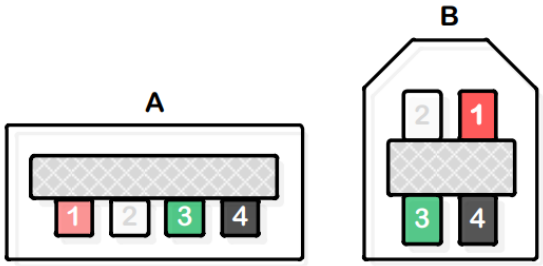
كل ناقلة تحتوي على 3 حزم متتابعة هي Token Packet وData Packet و Handshake Packet [11] وهذا موضح في الشكل(III-19).



الشكل III-19-بنية إطار البيانات[11]

III-5-7-أنواع الموصلات USB:

يوجد نوعان قياسيان من الموصلات USB، النوع A وB كما يوضحه الشكل(III-20).



الشكلIII-20- موصلان من النوع A وB [8].

وهناك أيضا نوع يسمى Mini B يستخدم مع الأجهزة التي تراعي الحجم كثيرا كالكاميرات. هذا الموصل يمتلك خمسة أطراف حيث إنَّ الطرف الخامس ID يستخدم في بعض الأجهزة للتعرف على النمط الإفتراضي للجهاز[8]. ويوجد أيضا نوع آخر يسمى Micro B وهو المستعمل في الهواتف النقالة، وهو مشابه تماما للنوع Mini B وله نفس الأطراف. يمثل الشكل(III-21) الموصلين Micro B وMini B.

الشكلIII-21- موصلان من النوع Mini B وMicro B[10].

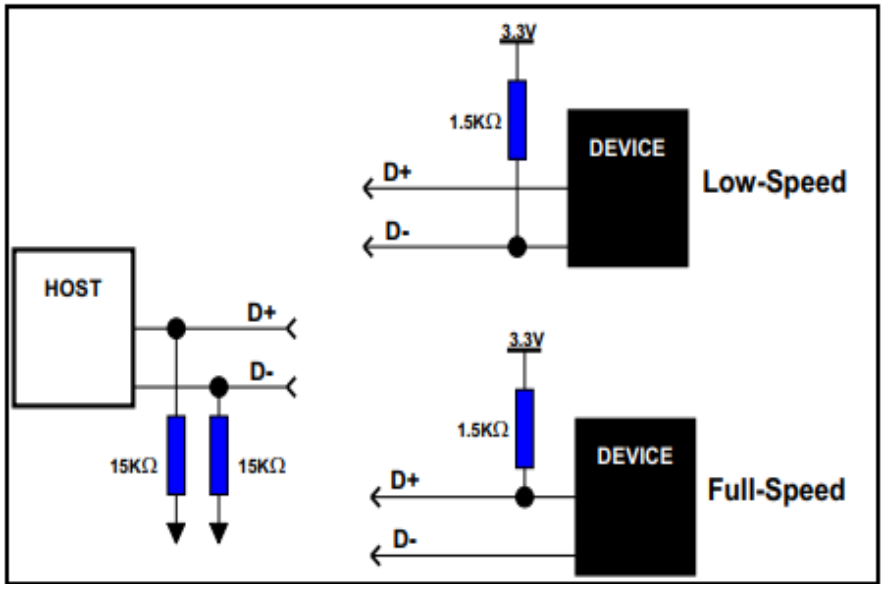
III-5-8-تحديد سرعة الناقل:

عند توصيل جهاز بالمضيف يقوم المجمع بتحديد سرعة الجهاز، وذلك عن طريق مقاومات الشدِّ على الخطين +D و-D [8].

إذا كانت مقاومة الشدِّ 1,5KΩ على +D فإن الجهاز يدعم السرعة الكاملة، وإذا كانت مقاومة الشدَّ على الخط -D فإن الجهاز يعمل على السرعة المنخفضة[11].

بالنسبة للأجهزة التي تدعم السرعة العالية فإنها تحدد بدايةً كأجهزة تدعم السرعة الكاملة ثم يقوم المجمع بإختبار الجهاز؛ هل يدعم السرعة العالية أم لا عن طريق إرسال إشارة مصافحة، وفي حال نجاح المصافحة يقوم الجهاز بإزالة مقاومة الشدِّ إلى الجهد 3,3V فولت ويبدأ الإتصال بالسرعة القصوى [8] [11].

يتم ربط خطي البيانات +D و-D للمضيف بمقاومتي شدٍّ 15KΩ كيلو أوم إلى الأرضي GND [11]. كل هذا موضح في الشكل(III-22).



الشكلIII-22- تحديد سرعة الناقل USB [11].

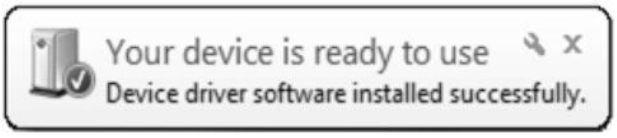
III-5-9-عملية إتصال المضيف مع الجهاز:

III-5-9-1-إعداد الإتصال:

يقوم المضيف بجمع المعلومات عن الجهاز الموصل به عن طريق سلسلة من الطلبات القياسية (يمكن إعتبارها أسئلة من أجل التعارف) فيقوم البرنامج المخزن بالجهاز بالإجابة عن هذه الأسئلة وإرسال المعلومات المطلوبة[8].

يوجد 11 طلبا قياسيا. بعض الأجهزة لا تدعم جميع هذه الطلبات فتقوم بإرسال إشارة تدل على أنها تدعم هذا الطلب[8].

هذه العملية بالنسبة لنظام ويندوز Windows تتم في خفاء وبشكل آلي بإستثناء إظهار رسالة تدل على إكتشاف جهاز جديد ونجاح العملية أم لا[8] كما في الشكل التالي.



الشكلIII-23- رسالة التعرف على جهاز جديد متصل بالمضيف[8].

III-5-9-2-مراحل الإعداد:

1. يقوم المستخدم بوصل الجهاز بمنفذ الخط USB.
2. يكتشف المجمع وجود جهاز موصول عن طريق مراقبة الجهود عند كل منفذ من منافذه.
3. يقوم المجمع بإعلام المضيف بوجود جهاز جديد متصل.
4. يقوم المجمع بتحديد سرعة الجهاز.
5. يقوم المجمع بإعادة تشغيل المنفذ المتصل بالجهاز وذلك بجعل الخطين +D و-D عند القيمة المنطقية '0'.
6. يحدد المضيف ما إذا كان الجهاز المتصل يدعم السرعة العالية أم لا.

7. يقوم المجمع بإنشاء مسار بين الجهاز والخط USB من خلال نقطة الاتصال 0 عبر قناة إفتراضية Default pipe.

8. يرسل المضيف طلبا للحصول على واصف الجهاز Device Descriptor وذلك لمعرفة الحجم الأقصى لحزمة البيانات.

9. يقوم المضيف بإرسال عنوان للجهاز؛ حيث يقوم هذا الأخير بالإستجابة وتخزين هذا العنوان وإرسال إشارة تعرف Acknowledge. يتم التعامل مع الجهاز عن طريق هذا العنوان عند أي إتصال آخر معه.

10. يرسل المضيف طلبا الحصول على الواصف لكن هذه المرة يقوم بقراءة كل ما يرسله الجهاز.

11. بعد أن يستقبل المضيف الواصف من الجهاز سيتمكن من تحديد أفضل برنامج قيادة Driver لهذا الجهاز.

12. يقوم برنامج القيادة بإرسال الإعدادات إلى الجهاز[8].

III-5-10-الواصف Descriptor:

يقوم المضيف كما أسلفنا بجمع معلومات عن الجهاز المتصل به حول خصائصه وإستطاعته. تتم هذه العملية من خلال الواصف وهو جدول موجود ضمن ذاكرة الجهاز يحتوي على بيانات واصفة له هذه الأخيرة مُعرّفة من قبل المصنع[11].

يحمل الواصف المعلومات التالية: رقم إعدادات الجهاز، والبرتوكولات التي يدعمها الجهاز، وتعريف البائع VID (الشركة)، وتعريف المنتج PID، والرقم التسلسلي للجهاز، وغيرها من المعلومات[11] التي يلخصها الجدول(III-1).

الجدولIII-1-جدول للبيانات المحمولة على الواصف [11].

Offset	Field	Size (Bytes)	Description
0	bLength	1	Length of this descriptor = 18 bytes
1	bDescriptorType	1	Descriptor type = DEVICE (01h)
2	bcdUSB	2	USB specification version (BCD)
4	bDeviceClass	1	Device class
5	bDeviceSubClass	1	Device subclass
6	bDeviceProtocol	1	Device Protocol
7	bMaxPacketSize0	1	Max Packet size for endpoint 0
8	idVendor	2	Vendor ID (or VID, assigned by USB-IF)
10	idProduct	2	Product ID (or PID, assigned by the manufacturer)
12	bcdDevice	2	Device release number (BCD)
14	iManufacturer	1	Index of manufacturer string
15	iProduct	1	Index of product string
16	iSerialNumber	1	Index of serial number string
17	bNumConfigurations	1	Number of configurations supported

III-5-11-أصناف الأجهزة Device Classes:

كل مجموعة أجهزة تتشابه في الوظائف توضع ضمن صنف واحد له برنامج قيادة واحد. يتضمن نظام التشغيل العديد من برامج القيادة للعديد من الأصناف. الجدول (III-2) يوضح بعض هذه الأصناف[8].

الجدولIII-2- جدول لبعض أصناف الأجهزة [8].

Device class	Description	Example device
0x00	Reserved	-
0x01	USB audio device	Sound card
0x02	USB communications device	Modem, fax
0x03	USB human interface device	Keyboard, mouse
0x07	USB printer device	Printer
0x08	USB mass storage device	Memory card, flash drive
0x09	USB hub device	Hubs
0x0B	USB smart card reader device	Card reader
0x0E	USB video device	Webcam, scanner
0xE0	USB wireless device	Bluetooth

صنف أجهزة مخاطبة الإنسان Human Interface Device HID هو صنف تندرج تحته كل الأجهزة التي تنقل المعلومات بسرعة متوسطة وتكون على إتصال مباشر مع المستخدم مثل لوحة المفاتيح والفأرة والمتحكم الدقيق[8].

يستخدم الصنف HID أحد الوصلتين: وصلة التحكم الافتراضية Default Control Pipe أو وصلة المقاطعة Interrupt Pipe[8].

III-6-خاتمة:

يعتبر بروتوكول الإتصال RS-232 غير متزامن ويعتمد في عمله على سرعة تدفق معينة وتغليف البيانات ببِت للبداية وبِت للنهاية لتحديد طول الحزمة وله عدة أنماط عمل، يستعمل أساسا لربط جهاز DTE مع DCE ولكن يمكن استعماله لربط جهازين DTE.

الاتصال التسلسلي USB هو أشهر تقنيات الإتصال يعتمد على تقنية النقل التفاضلي وعلى تقنية التشفير NRZI، يتطلب وجود تعارف بين الحاسوب والجهاز المتصل؛ يتم التعارف بتثبيت برنامج قيادة على الحاسوب اِعتمادا على واصف الجهاز.

الفصل الرابع : الجانب العملي

IV-1-المقدمة :

في هذا الفصل سنتطرق لمراحل إنجاز المشروع، حيث سنقدم شرحا وتوضيحا لأهم النقاط في الجانب العملي والمتمثلة في البرمجة، والبروتوكولات المستخدمة، ثم نتحدث عن الألواح التطويرية المنجزة خلال المشروع مع تعريف وشرح للوسائل المستخدمة من عتاد وبرمجيات.

IV-2-بروتوكولات البرمجة:

لكل عائلة من عائلات المتحكمات PIC بروتوكولات خاصة لبرمجتها، هذه الأخيرة تُعطى من قبل المُصَنِّع ضمن دفتر المواصفات Data Sheet. مَعْرِفة هذه البروتوكولات هو الأساس لصنع جهاز برمجة وتطوير تطبيق برمجي خاص به على الحاسوب.

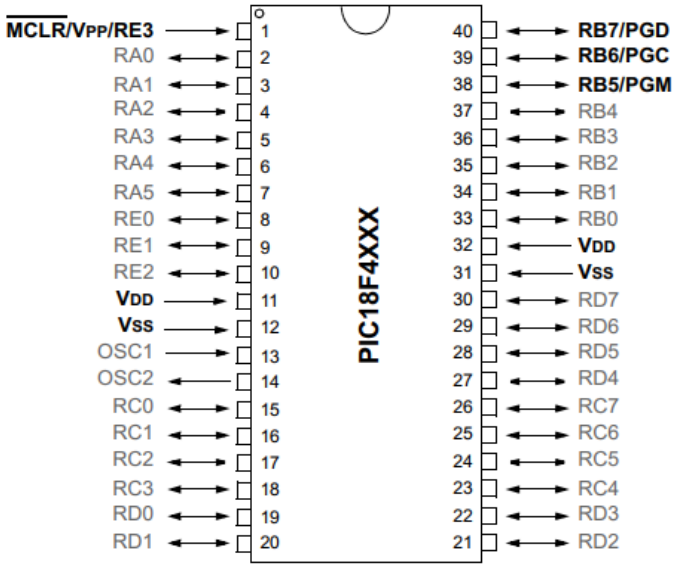
IV-2-1-طرق البرمجة داخل الدارة ICSP:

يمكن برمجة المتحكمات الدقيقة من العائلة PIC18F2xxx/4xxx بطريقتين، الأولى هي البرمجة داخل الدارة بجهد عالٍ HIGH Voltage ICSP، والثانية هي البرمجة بجهد منخفضLOW Voltage ICSP[1].

بالنسبة للبرمجة بجهد عالي فإننا نستعمل 3 أطراف من المتحكم الدقيق لإجراء العملية وهي الطرف MCLR/VPP/RE3 وRB6/PGC وRB7/PGD، كما أننا نحتاج إلى مصدرين مختلفين للجهد واحد من أجل التغذية Vdd والآخر للبرمجة Vpp[1].

أما بالنسبة للبرمجة بجهد منخفض فإننا نستعمل نفس الأطراف إضافة إلى الطرف RB5/PGM، وهنا لا نحتاج إلى مصدرين للجهد من أجل التغذية ويكفي مصدر واحد للبرمجة لكل من Vdd وVpp[1].

يبين الشكل(IV-1) الأطراف المستعملة في البرمجة.



الشكل IV-1- توضيح للأطراف المستعملة في برمجة متحكمات العائلة PIC18F4xxx [1].

يمثل الجدول(IV-1) وظيفة كل طرف أثناء البرمجة.

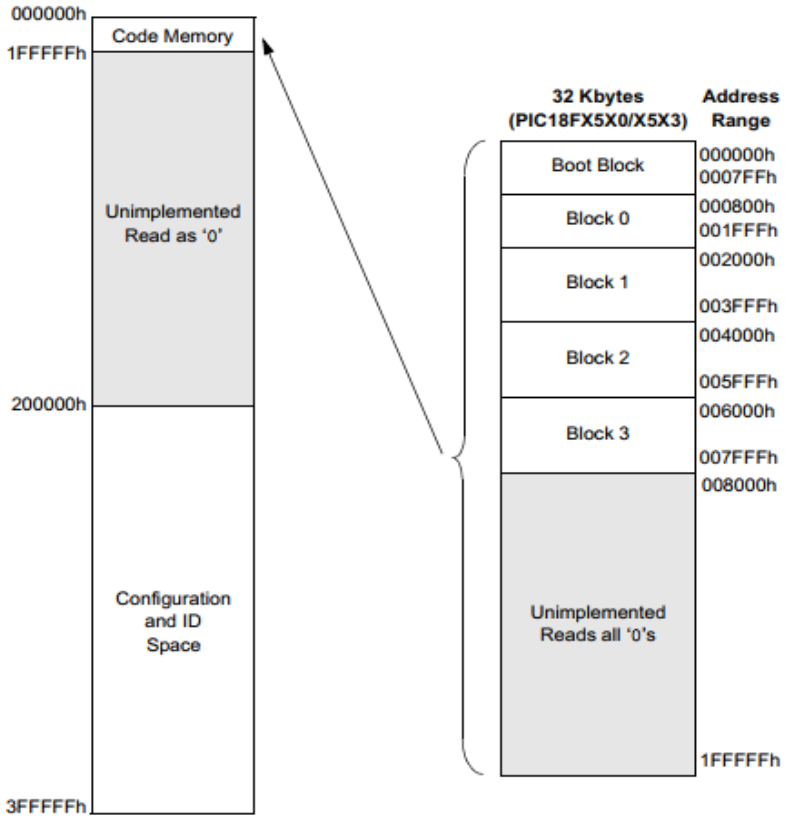
الجدول IV-1- وظائف الأطراف الخاصة ببرمجة المتحكم[1].

الطرف Pin	وضعيتها Type	وظيفتها Function
MCLR/Vpp/RE3	P	تفعيل البرمجة
VDD	P	جهد التغذية Power Supply
VSS	P	الأرضي Ground
RB7/PGD	I/O	نقل البيانات
RB6/PGC	I	مدخل لإشارة الساعة Clock
RB5/PGM	I	الدخول لوضع البرمجة بجهد منخفض L.V.ICSP

P طاقة Power، I مدخل،O مخرج.

IV-2-2-تقسيم مساحة ذاكرة البرنامج Flash Memory:

بالنسبة للعائلة PIC18F2xxx/18F4xxx فإن مساحة الذاكرة المخصصة للملف البرمجي Hex Code محصورة بين العنوانين 000000 h و007FFF h أي بحجم 32 Kbytes مقسمة إلى 4 أقسام حجم كل منها 8 Kbytes. يوضح الشكل(IV-2) تقسيم مساحة الملف البرمجي[1].

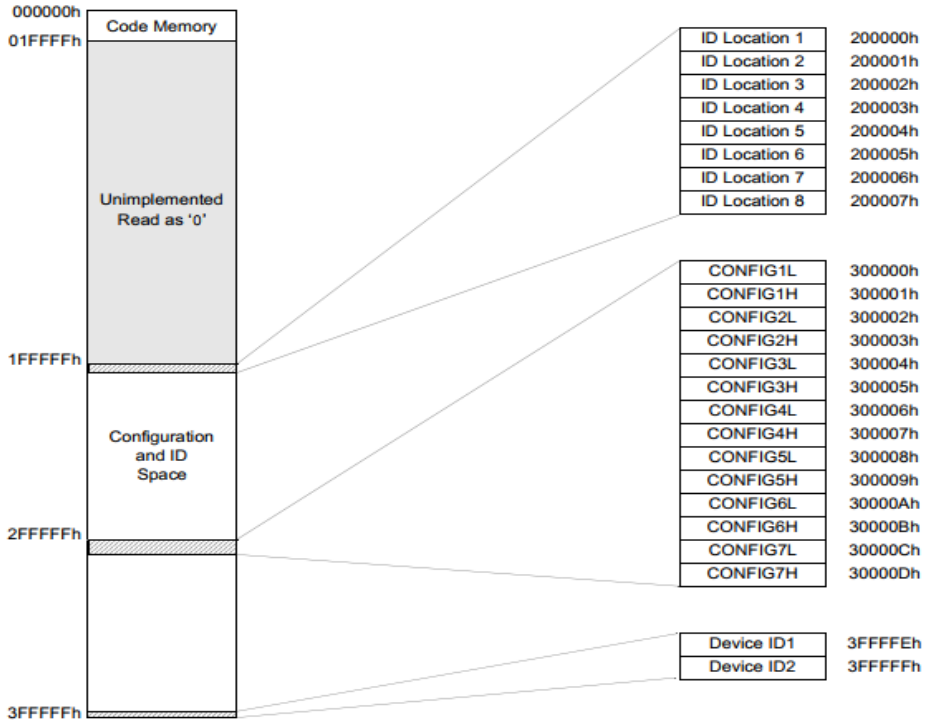


الشكل IV-2- المساحة المخصصة للملف البرمجي من ذاكرة البرنامج [1].

إضافة إلى جزء الملف البرمجي من الذاكرة Flash فإنه يوجد 3 أجزاء أخرى يمكن للمستخدم الكتابة فيها أو القراءة منها وهي[1]:

- جزء معلومات التعريف Identification Information ID يشمل 8 مسجلات ضمن الذاكرة في العناوين من 200000 h إلى 200007 h.
- الأماكن الموجودة بين العنوانين 300000 h و30000D h محجوزة لجزء بتات الإعدادات Configuration Bits.
- جزء رقم تعريف الجهاز ID bits محجوز له العنوانان 3FFFFE h و3FFFFF h.

الشكل (IV-3) يوضح تقسيم ذاكرة البرنامج.



الشكل IV-3- أجزاء ذاكرة البرنامج Flash Memory[1].

IV-2-3-مؤشر عنوان الذاكرة Memory Address Pointer:

يتم التعامل مع الذاكرة عن طريق العنونة (22 بيت) بواسطة جدول مكون من ثلاثة مسجلات؛

يمثل الجدول(IV-2) هذا الجدول؛ هذه المسجلات هي[1]:

- المسجل TBLPTRU في العنوان 0FF8 h من الذاكرة RAM.
- المسجل TBLPTRH في العنوان 0FF7 h من الذاكرة RAM.
- المسجل TBLPTRL في العنوان 0FF6 h من الذاكرة RAM.

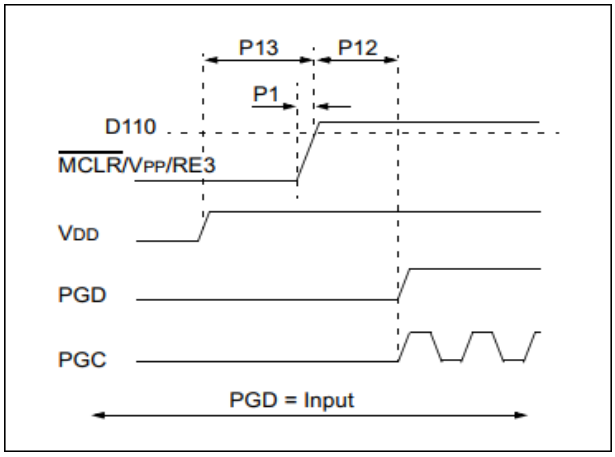
الجدول IV-2- مسجلات مؤشر عنوان الذاكرة [1].

TBLPTRU	TBLPTRH	TBLPTRL
Addr[21:16]	Addr[15:8]	Addr[7:0]

IV-2-4-بدء وإنهاء البرمجة بالجهد العالي :

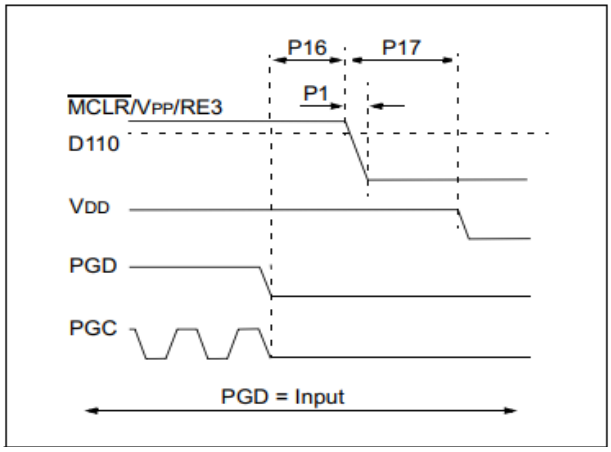
يتم الدخول إلى وضع البرمجة بجعل كل من الطرفين PGC وPGD في المستوى المنخفض'0'، ثم بعد ذلك يتم رفع الجهد الداخل إلى الطرف MCLR/Vpp/RE3 إلى أن يصل العتبة V_{IHH} [1] يوضح الشكل(IV-4) هذه العملية.

ملاحظة: جميع القيم والثوابت المشار إليها في تمثيلات بروتوكولات البرمجة مبينة في الملحق1.



الشكلIV-4- الدخول إلى وضع البرمجة عالية الجهد [1].

بينما يتم الخروج من وضع البرمجة بجعل كل من الطرفين PGC وPGD في المستوى المنخفض ثم بعد ذلك خفض جهد الطرف MCLR/Vpp/RE3 [1] كما هو موضح في الشكل(IV-5).

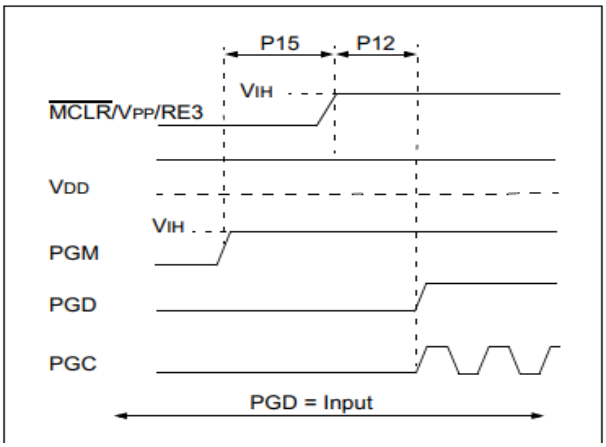


الشكلIV-5- الخروج من وضع البرمجة عالية الجهد [1].

IV-2-5-بدء وإنهاء وضع البرمجة بالجهد المنخفض:

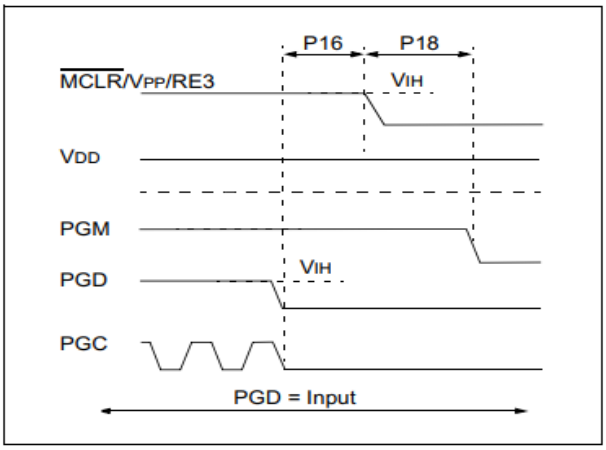
لتفعيل هذا الوضع يجب أن يكون البت LVP (Low Voltage Programming) من بتات الإعدادات يساوي واحد '1' [1].

يتم الدخول إلى هذا الوضع بجعل كل من الطرفين PGC وPGD في المستوى المنخفض'0' وجعل الطرف PGM في المستوى العالي '1' ثم بعد ذلك رفع الجهد في الطرف MCLR/Vpp/RE3 إلى المستوى العالي [1] كما هو موضح في الشكل(IV-6).



الشكلIV-6- الدخول إلى وضع البرمجة بالجهد المنخفض [1].

طريقة الخروج من وضع البرمجة هنا هي بجعل كل من PGC وPGD في المستوى المنخفض ثم بعد ذلك يتم خفض جهد الطرف MCLR/Vpp/RE3 ثم خفض جهد PGM [1]، وهذا موضح في الشكل (IV-7)



الشكلIV-7- الخروج من وضع البرمجة بالجهد المنخفض [1].

IV-2-6-نقل البيانات خلال عملية البرمجة:

الطرف PGC يستعمل كمدخل لنبضات الساعة، والطرف PGD يستعمل كمدخل أو مخرج للبيانات خلال عملية البرمجة [1].

يتم إرسال الأوامر البرمجية عبر PGD عند الحافة الصاعدة لنبضة الساعة The Rising Edge of PGC وتُثَبتُ عند الحافة النازلة للنبضة The Falling Edge، يرسل البت 0 أولا Lsb First [1].

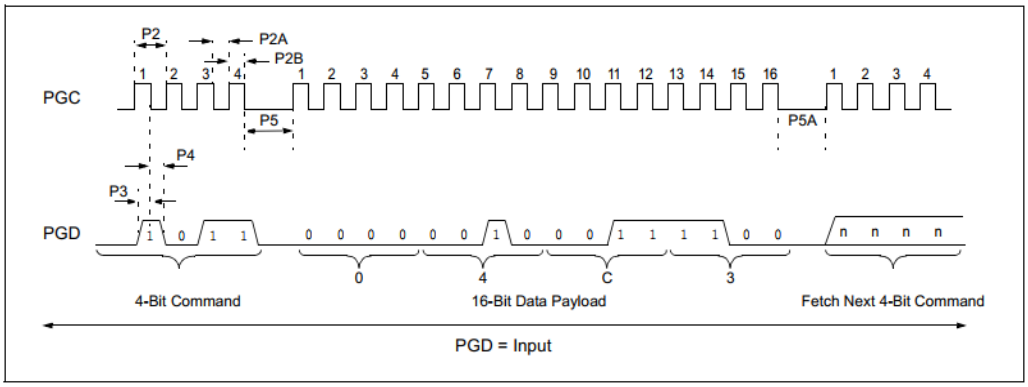
كل الأوامر تتكون من 20 Bits، أربعة بتات خاصة بأمر التحكم يتبعها 16 Bits للبيانات، يتم إدخال بتات التحكم عبر PGD خلال أربعة أدوار كاملة للPGC [1].

على حسب أمر التحكم تكون بتات البيانات الـ16 إما كلها بتات لبيانات داخلة أو 8 بتات داخلة ومثلها خارجة [1]. الجدول(IV-3) يبين أوامر التحكم الموجودة والجدول(IV-4) مثال لأحد الأوامر.

Description	4-Bit Command
Core Instruction (Shift in16-bit instruction)	0000
Shift Out TABLAT Register	0010
Table Read	1000
Table Read, Post-Increment	1001
Table Read, Post-Decrement	1010
Table Read, Pre-Increment	1011
Table Write	1100
Table Write, Post-Increment by 2	1101
Table Write, Start Programming, Post-Increment by 2	1110
Table Write, Start Programming	1111

الجدولIV-3- أوامر التحكم الخاصة بالبرمجة [1].

الشكل (IV-8) هو تمثيل بياني لإرسال الأمر التالي: 3C40 1101.



الشكلIV-8- تمثيل بياني لإرسال أمر معين عبر PGD بالتزامن مع نبضات[1].

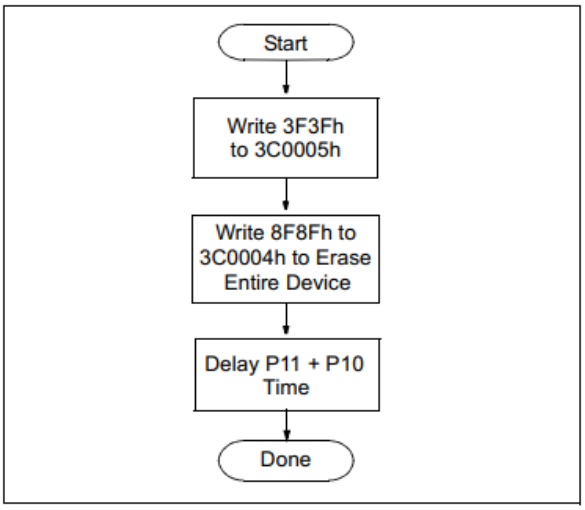
IV-2-7-عملية مسح الذاكرة ICSP Erase:

لإجراء عملية مسح لذاكرة البرنامج أو ذاكرة البيانات يجب القيام بإعدادات على المسجلين الخاصين بالتحكم في العملية، يتواجد كل من المسجلين في العنوانين 3C0004 h و3C0005 h[1].

يمكن أن يكون المسح جزئيا، ويمكن للمستخدم مسح الذاكرة بأكملها مرة واحدة، كما أن عملية المسح هذه يمكنها أيضا حذف أي قفل حماية موجود على أحد أجزاء الذاكرة[1]. والجدول(IV-5) يمثل خوارزمية المسح الكلي للذاكرة كما يوضح الشكل(IV-9) مخطط مراحل عملية المسح.

الجدولIV-4- خوارزمية عملية المسح الكلي للذاكرة [1].

4-Bit Command	Data Payload	Core Instruction
0000	0E 3C	MOVLW 3Ch
0000	6E F8	MOVWF TBLPTRU
0000	0E 00	MOVLW 00h
0000	6E F7	MOVWF TBLPTRH
0000	0E 05	MOVLW 05h
0000	6E F6	MOVWF TBLPTRL
1100	3F 3F	Write 3F3Fh to 3C0005h
0000	0E 3C	MOVLW 3Ch
0000	6E F8	MOVWF TBLPTRU
0000	0E 00	MOVLW 00h
0000	6E F7	MOVWF TBLPTRH
0000	0E 04	MOVLW 04h
0000	6E F6	MOVWF TBLPTRL
1100	8F 8F	Write 8F8Fh TO 3C0004h to erase entire device.
0000	00 00	NOP
0000	00 00	Hold PGD low until erase completes.



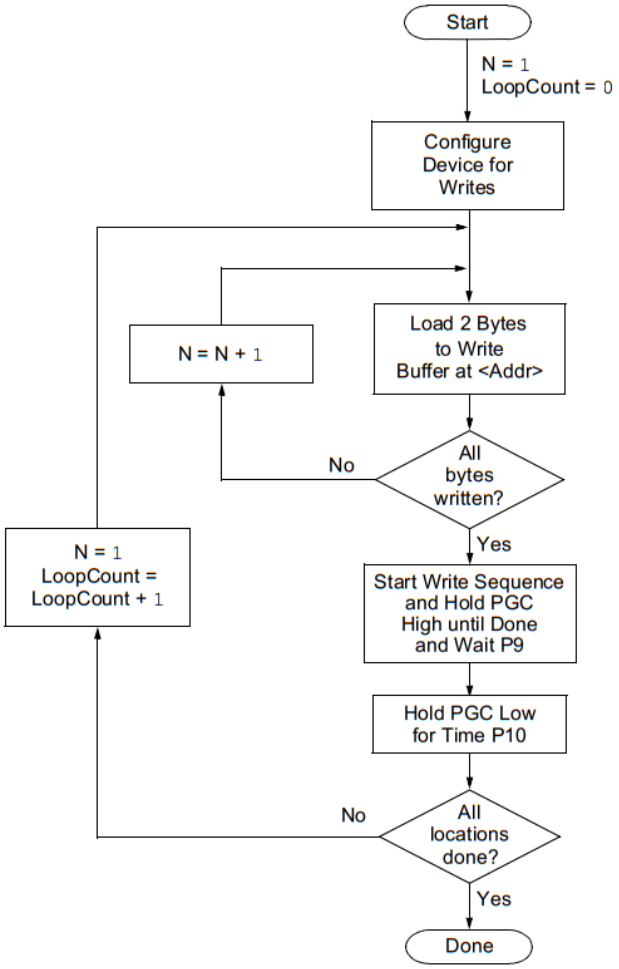
الشكلIV-9- مخطط لمراحل عملية المسح الكلي[1].

IV-2-8-كتابة البرنامج على الذاكرة Code Memory Programing:

تتم عملية كتابة البرنامج على الذاكرة أولا بتحميل البيانات على حامل للكتابة Write Buffer، وهو مصفوفة تحتوي على عدد معين من البايتات أي أنه يتم تحميل جزء من البرنامج فقط، في حالة المتحكم PIC18F4550 هذا الحامل حجمه 32 bytes وبالتالي يتم كتابة 32 bytes على الذاكرة مرة واحدة ثم تحميل جزء آخر من البرنامج وبنفس الحجم السابق وهكذا إلى أن يتم كتابة كل البرنامج على الذاكرة[1].

بعد تحميل الـ 32 bytes من البرنامج يتم كتابتها على الذاكرة بالأمر '1101' ما عدا 2bytes الأخيرين فإنهما يكتبان وحدهما، تبدأ عملية البرمجة بالأمر '1111'، ثم يوضع الخط PGD في الصفر، ويوضع الخط PGC في 1 لمدة P9 ثم في 0 لمدة P10[1].

الجدول(IV-6) يمثل خوارزمية عملية برمجة الذاكرة Flash، والشكل(IV-10) يمثل مخطط تنفيذي لهذه الخوارزمية.



الشكلIV-10- مخطط لمراحل عملية كتابة البرنامج على الذاكرة [1].

الجدولIV-5- خوارزمية عملية كتابة البرنامج على الذاكرة [1].

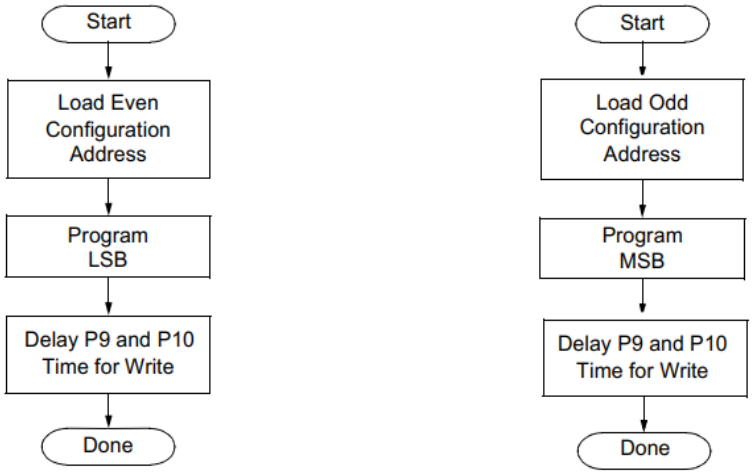
4-Bit Command	Data Payload	Core Instruction
Step 1: Direct access to code memory and enable writes.		
0000 0000	8E A6 9C A6	BSF EECON1, EEPGD BCF EECON1, CFGS
Step 2: Load write buffer.		
0000 0000 0000 0000 0000 0000	0E <Addr[21:16]> 6E F8 0E <Addr[15:8]> 6E F7 0E <Addr[7:0]> 6E F6	MOVLW <Addr[21:16]> MOVWF TBLPTRU MOVLW <Addr[15:8]> MOVWF TBLPTRH MOVLW <Addr[7:0]> MOVWF TBLPTRL
Step 3: Repeat for all but the last two bytes.		
1101	<MSB><LSB>	Write 2 bytes and post-increment address by 2.
Step 4: Load write buffer for last two bytes.		
1111 0000	<MSB><LSB> 00 00	Write 2 bytes and start programming. NOP - hold PGC high for time P9 and low for time P10.
To continue writing data, repeat steps 2 through 4, where the Address Pointer is incremented by 2 at each iteration of the loop.		

IV-2-9-برمجة الإعدادات Configuration Bits Programing:

على خلاف كتابة البرنامج، فإن برمجة الإعدادات تكون بالبايت حيث يُرسل بايت فقط (8 Bits) يحمل بيانات الإعدادات ضمن ال 16 Bits الكلية للبيانات المُرسَلة عبر PGD، وحَسْبَ عنوانِ مسجل الإعدادات المراد برمجته تكون وضعية بايت الإعدادات في حزمة البيانات المرسلة؛ بحيث إذا كان مسجل الإعدادات ذا عنوان زوجي Even Address فإن الإعدادات تكون في البايت الأول Lsb من الحزمة (16 بت)، وإذا كان مسجل الإعدادات ذا عنوان فردي Odd Address فإن الإعدادات تكون في البايت الثاني Msb من الحزمة[1]. خوارزمية الإعدادات في الجدول التالي والشكل(IV-11) مخطط تنفيذي لها.

الجدولIV-6- خوارزمية برمجة الإعدادات [1].

4-Bit Command	Data Payload	Core Instruction
Step 1: Enable writes and direct access to configuration memory.		
0000 0000	8E A6 8C A6	BSF EECON1, EEPGD BSF EECON1, CFGS
Step 2: Set Table Pointer for configuration byte to be written. Write even/odd addresses.(1)		
0000 0000 0000 0000 0000 0000 1111 0000 0000 0000 1111 0000	0E 30 6E F8 0E 00 6E F7 0E 00 6E F6 <MSB ignored><LSB> 00 00 0E 01 6E F6 <MSB><LSB ignored> 00 00	MOVLW 30h MOVWF TBLPTRU MOVLW 00h MOVWF TBLPRTH MOVLW 00h MOVWF TBLPTRL Load 2 bytes and start programming. NOP - hold PGC high for time P9 and low for time P10. MOVLW 01h MOVWF TBLPTRL Load 2 bytes and start programming. NOP - hold PGC high for time P9 and low for time P10.



الشكلIV-11- مخطط لمراحل عملية برمجة الإعدادات[1].

IV-2-10-قراءة ذاكرة البرنامج Reading Flash Memory:

قراءة ذاكرة البرنامج تكون بالبايت لكل أمر قراءة. تعنون الذاكرة أثناء العملية بواسطة جدول المؤشر. يتم استقبال البيانات من الطرف PGD بعد تحويله إلى مدخل بحيث تكون القراءة ابتداءًا بالبت Lsb وصولا إلى البت Msb وهذا خلال ثماني دورات كاملة للـ PGC يتم خلالها تحميل البتات عند الحافة النازلة للنبضة [1]. الجدول التالي يمثل خوارزمية عملية القراءة.

الجدولIV-7- خوارزمية عملية قراءة خانة واحدة من الذاكرة [1].

4-Bit Command	Data Payload	Core Instruction
Step 1: Set Table Pointer.		
0000 0000 0000 0000 0000 0000	0E <Addr[21:16]> 6E F8 0E <Addr[15:8]> 6E F7 0E <Addr[7:0]> 6E F6	MOVLW Addr[21:16] MOVWF TBLPTRU MOVLW <Addr[15:8]> MOVWF TBLPTRH MOVLW <Addr[7:0]> MOVWF TBLPTRL
Step 2: Read memory and then shift out on PGD, LSb to MSb.		
1001	00 00	TBLRD *+

IV-3-اللوحات التطويرية:

أنجزنا ثلاث ألواح تطويرية مختلفة فيما بينها في تقنية البرمجة لكنها متشابهة في الخصائص والإمكانات ولها نفس المتحكم الدقيق وكلها تدعم البرمجة بواسطة MikroC.

IV-4-اللوحة التطويرية الأولى:

في هذه اللوحة اِعتمدنا على تقنية برنامج التحميل Bootloader وهي نفسها التقنية المستخدمة في لوحات الأردوينو.

IV-4-1-برنامج التحميل:

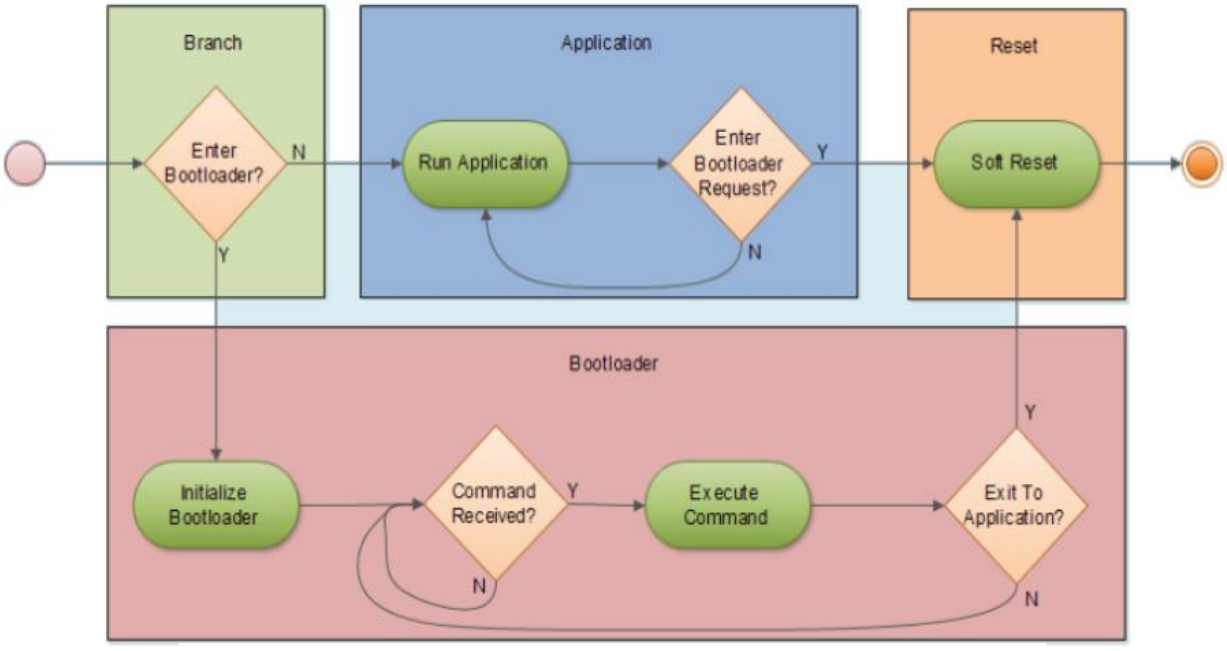
هو برنامج تم تثبيته مسبقا على المتحكم الدقيق يعمل على اِستقبال بيانات الملف البرمجي وكتابتها على ذاكرة البرنامج مباشرة دون دخول المتحكم إلى وضع البرمجة ICSP. يتم الاِتصال مع الحاسوب بتقنية الاتصال التسلسلي RS-232 أو USB؛ حيث يجب توفر تطبيق برمجي على الحاسوب يعمل على فتح وإرسال بيانات الملف البرمجي إلى المتحكم[2].

تختلف طريقة عمل برامج التحميل في المتحكمات لكنها تعتمد كلها على 3 مراحل أساسية[2]:

- مرحلة اِختيار وضع عمل المتحكم.
- مرحلة تنفيذ التطبيق البرمجي.
- مرحلة تنفيذ برنامج التحميل.

عند تشغيل المتحكم فإنه يدخل في حالة اِختبار للاتصال مع الحاسوب؛ هذا الإختبار لمعرفة ما إذا كان هناك إرسال لملف برمجي أو لا. عند وجود ملف برمجي مرسل من قبل الحاسوب فإن المتحكم يدخل في مرحلة تنفيذ برنامج التحميل بعدها يقوم بعملية تصفير وإعادة تشغيل للمتحكم، وفي حالة عدم وجود ملف برمجي مرسل من قبل الحاسوب فإن المتحكم يدخل في مرحلة تنفيذ التطبيق البرمجي الموجود فيه وأثناء ذلك وفي حالة وجود طلب لتنفيذ برنامج التحميل تحدث عملية تصفير وإعادة تشغيل للمتحكم[2].

يمثل الشكل التالي المراحل السابقة.



الشكلIV-13- مخطط لمراحل عمل تقنية برنامج التحميل [2].

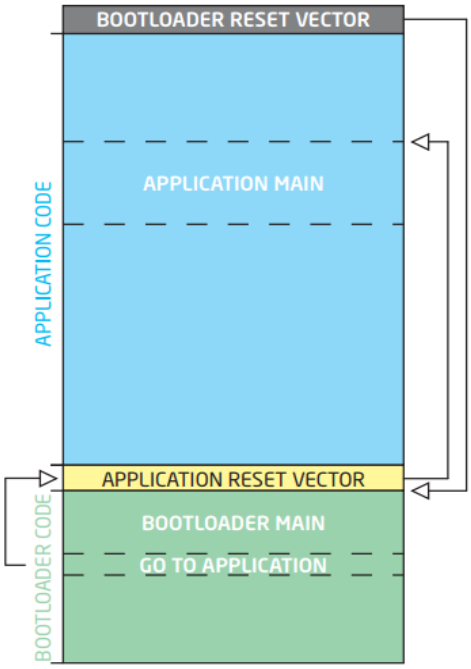
IV-4-2-برنامج USB HID Bootloader:

هو برنامج تحميل يسمح للمتحكم بالتواصل مع الحاسوب عبر واجهة الأجهزة USB HID. تم تصميم هذا البرنامج من طرف شركة MikroElektronika للعمل مع التطبيق البرمجي Mikro Bootloader الموجود ضمن أدوات بيئة تطوير MikroC.

صمم هذا البرنامج من أجل مجموعة من اللوحات التطويرية للشركة مثل لوحة Mini32 [3].

IV-4-3-تقسيم ذاكرة البرنامج :

يمثل الشكل(IV-14) تقسيما لذاكرة البرنامج بين برنامج التحميل والتطبيق البرمجي Application[4].



الشكلIV-14-تقسيم ذاكرة البرنامج بين برنامج التحميل والتطبيق البرمجي[4].

ينحسر التطبيق البرمجي المُحمّل ضمن مساحة محدودة من الذاكرة؛ وذلك لوجود برنامج التحميل الثابت ضمن الذاكرة. يتم مسح جزء التطبيق البرمجي فقط عند تحميل تطبيق جديد.

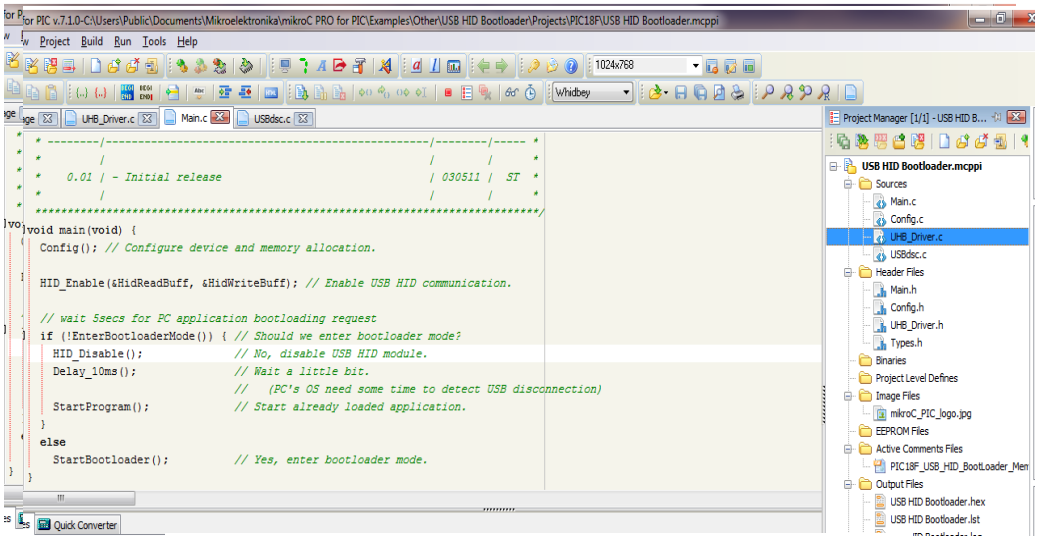
IV-4-4-حرق برنامج التحميل USB HID Bootloader على ذاكرة المتحكم PIC18F4550:

برنامج التحميل متواجد ضمن ملفات برنامج MikroC وفق المسرى التالي:

C:\Users\Public\Documents\Mikroelektronika\mikroC PRO for PIC\ Examples\ Other\USB HID Bootloader\Projects\PIC18F\USB HID Bootloader.mcppi

نلاحظ وجود ثلاثة ملفات مفتوحة على الميكروسي؛ الأول هو UHB Driver.c والثاني Main.c والثالث هو USBdsc.c.

الملف Main.c هو الملف الرئيسي لبرنامج التحميل لكن لا يمكن تحميله لوحده على المتحكم لأنه كما سبق وتكلمنا عن الأجهزة ذات برنامج القيادة HID فإنه يجب توفر واصف على هذه الأخيرة يسمح للحاسوب بالتعرف على الجهاز وتثبيت برنامج قيادة خاص به؛ الواصف هو الملف USBdsc.c والملف UHB Driver.c يحتوي على متغيرات ودوال التعامل مع البيانات الداخلة والخارجة من المتحكم عبر المنفذ USB أثناء عمل برنامج التحميل؛ لذلك يجب التأكد من إضافة هذين الملفين إلى الملف الرئيسي Main.c وذلك عن طريق نافذة إدارة المشروع Project Manager على يمين واجهة الميكروسي كما في الشكل(IV-15).



الشكلIV-15- نافذة إدارة المشروع ضمن بيئة تطوير

كما يجب إضافة الملف Config.c والملفات الرأسية Header Files التالية Main.h، وConfig.h، وUHB Driver.h، وTypes.h وذلك حسب التعليمات الواردة ضمن التعليق المقدم في الملف Main.c.

IV-4-5-واجهة التطبيق mikroBootloader:

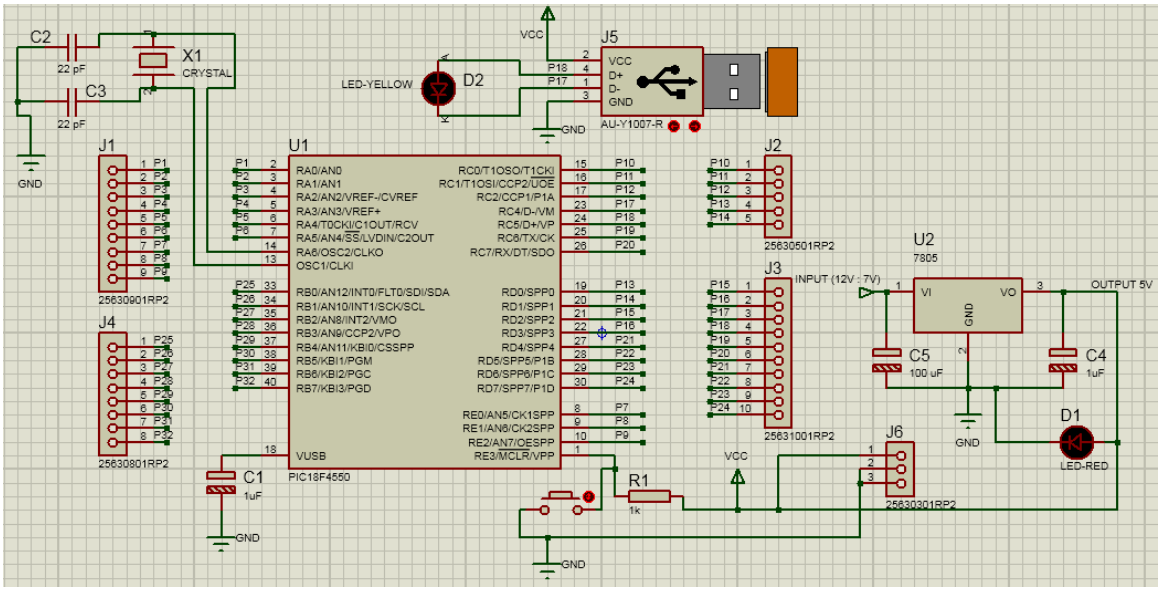
يوجد على بيئة التطوير MikroC أدوات للتعامل مع اللوحات التطويرية بتقنية برنامج التحميل؛ حيث توجد الأداة USB HID mikroBootloader من أجل المتحكمات ذات الواجهة USB من العائلة PIC18F وPIC32، والأداة mikroBootloder من أجل المتحكمات ذات الواجهة UART من العائلة 16F وحتى 18F وغيرها...؛ كما قدمنا فإننا اِستعملنا الأداة USB HID mikroBootloader حيث يمثل الشكل(IV-16) واجهتها.

الشكلIV-16-واجهة الأداة USB mikroBootloader.

IV-4-6-دارة اللوحة التطويرية:

قمنا بتصميم نموذج للوحة تطويرية بإستعمال برنامج Proteus 8؛ الدارة تتكون من متحكم دقيق PIC18F4550، وموصل USB، وموصل لجهد التغذية الخارجي مع منظم جهد lm7805 ومكثفين للاستقرار بقيمة 100 uF و1uF، ومجموعة موصلات لمداخل ومخارج منافذ المتحكم GPIO، مكثفة بقيمة 1uF، وزر ضغط Push Button مع مقاومة 1kΩ، وهزاز كريستال بقيمة

20Mhz مع مكثفين بقيمة22pF، وصمامين ضوئيين LEDs. الشكل (4-17) يمثل تصميما للوحة التطويرية.



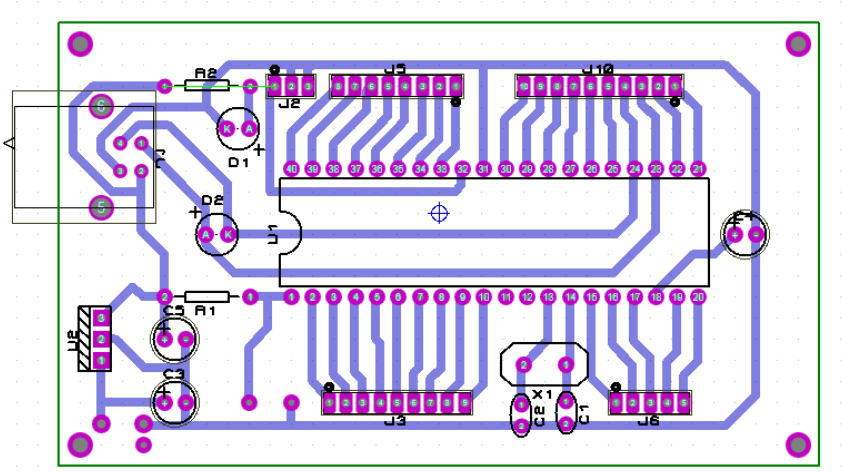
الشكلIV-17-نموذج لدارة اللوحة التطويرية الأولى على Proteus.

يوجد مصدران لجهد التغذية في الدارة؛ المصدر الأول هو جهد التغذية Vcc من الموصِّل USB وهو بقيمة ثابتة 5V والمصدر الثاني هو دارة التغذية المتكونة من منظم الجهد lm7805 ؛ حيث إن أقصى قيمة لمدخل منظم الجهد هو 25V لكن نحرص على أن يكون الجهد محصورا بين 12V و7V عموما. توفر دارة التغذية جهد ثابتا بقيمة 5V وتيارا بقيمة قصوى تبلغ 1,5A حسب إمكانيات منظم الجهد lm7805.

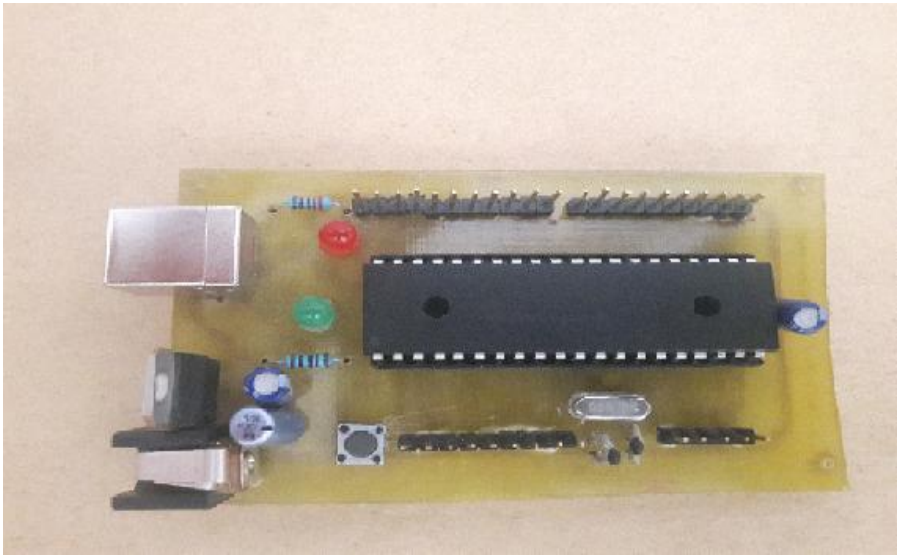
يربط هزاز الكريستال –Quartz– بمكثفين قيمة كل منهما 22pF، واِخترنا القيمة 20Mhz ليعمل المتحكم بالسرعة القصوى. دارة التصفير للرجل MCLR تتكون من مقاومة 1KΩ موصولة على التسلسل بجهد التغذية VCC وبزر ضغط على التفرع متصل هو الآخر بالمأخذ الأرضي GND.

المكثفة C1 هي مكثفة خاصة بمنظم الجهد 3V3 للوحدة USB، حيث ينصح حسب دفتر بيانات المتحكم Data Sheet بمكثفتين بقيمة 220nF مربوطتين على التفرع إلى المأخذ الأرضي

GND؛ لكن بالتجريب وبالإطلاع على منتديات موقع شركة ميكروشيب تبين أنه يوجد مجال لقيمة المكثفة المستعملة مع منظم الجهد لوحدة USB وهو من قيمة 220nF إلى 10uF ونحن استعملنا مكثفة 1uF. الصمامان الضوئيان أحدهما للتأكد من تشغيل الدارة والآخر للتأكد من اِنتقال البيانات عبر المنفذ USB. تحتوي الدارة أيضا على منفذ لجهد ثابت 5V للاِستعمال الخارجي إضافة إلى منفذين للمأخذ الأرضي GND. صممنا مخططا للدارة المطبوعة لهذه اللوحة التطويرية باستخدام الأداة PCB Layout من برنامج Proteus كما يوضحه الشكل(IV-18).



الشكلIV-18-مخطط للدارة المطبوعة الخاصة باللوحة التطويرية على برنامج Proteus.



الشكلIV-19-صورة للوحة التطويرية الأولى.

IV-4-7-إستعمال اللوحة التطويرية :

لتحميل أي برنامج مكتوب بلغة الميكروسي على اللوحة التطويرية المنجزة نقوم بالتالي[4]:

1/ التأكد من أن إعدادات الملف البرمجي مضبوطة وفق المتحكم PIC18F4550.

الشكل IV-20-تحديد المتحكم الدقيق من طرف برنامج mikroBootloader.

2/ نفتح واجهة برنامج USB HID mikroBootloader.

3/ نوصل اللوحة التطويرية بالحاسوب عندها تتم عملية التعرف ثم يصبح لون شعار USB على واجهة برنامج mikroBootloader أحمرا دلالة على تحديد الجهاز؛ يبقى هذا مدة 5 ثوانٍ إن لم يتم الإنتقال إلى المرحلة التالية فإن الإتصال سينقطع وهذا موضح في الشكل (IV-20).

4/ خلال 5 ثوان من تحديد الجهاز نضغط على إتصال Connect كما يبين الشكل(IV-21)، عندها يدخل المتحكم إلى برنامج التحميل لتنفيذه. عند حدوث إنقطاع للإتصال نضغط على زر التصفير Reset للمتحكم ونعيد الإتصال من جديد.

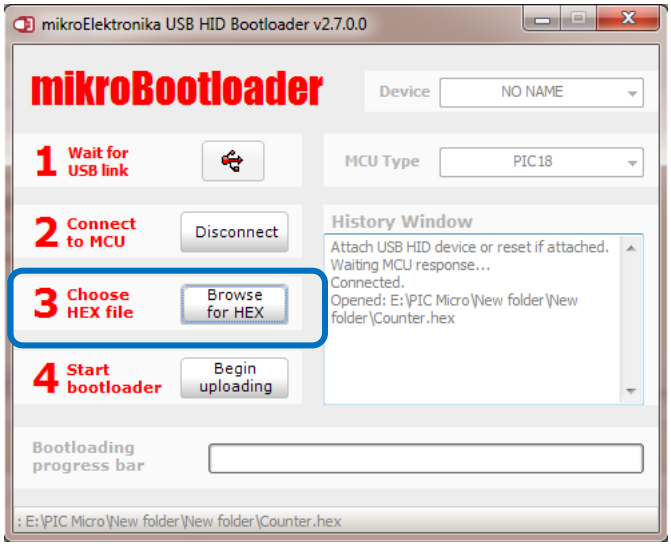
الشكل IV-21-تحديد المتحكم الدقيق من طرف برنامج mikroBootloadr.

5/ نقوم بإختيار الملف البرمجي المراد تحميله على المتحكم وهذا بالضغط على Browse for hex كما في الشكل (22-IV).

6/ بعد إختيار الملف المراد تحميله على المتحكم نضغط على بداية التحميل Begin uploading كما في الشكل(23-IV)

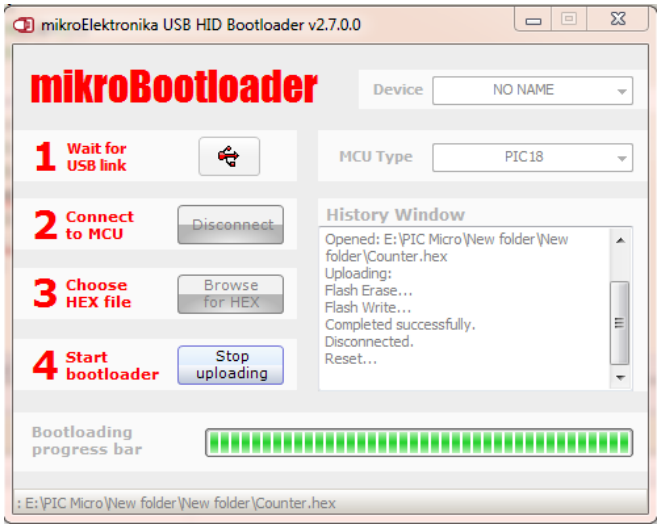
هنا يبدأ المتحكم في تنفيذ برنامج التحميل.

7/ بعد الإنتهاء من تحميل الملف البرمجي تحدث عملية تصفير للمتحكم، كما يبينه الشكل(24-IV).



الشكل IV-22-البحث عن الملف البرمجي باستعراض المجلدات.

الشكل IV-23- بدأ التحميل.



الشكل IV-24-نهاية عملية التحميل وتصفير المتحكم وإعادة تشغيله.

IV-5-اللوحة التطويرية الثانية:

في هذه اللوحة اِعتمدنا في عملية البرمجة على دمج مبرمجة من نوع JDM في الدارة وعزل أرجل المتحكم المرتبطة بها بواسطة مفتاح Switch.

IV-5-1-المبرمجة JDM:

تعتبر من أشهر المبرمجات التسلسلية، تعتمد على تحويل الشفرات البرمجية إلى قلب المتحكمات عبر المنفذ المتتالي (التسلسلي) المعروف باسم RS232.[5]

إنَّ البرمجة بواسطة مبرمجة JDM تعتمد بشكل أساسي على توصيل 5 أطراف من المنفذ التسلسلي Serial Port للكمبيوتر إلى البيك وذلك لإتمام عملية تبادل المعلومات بينهما.[6]

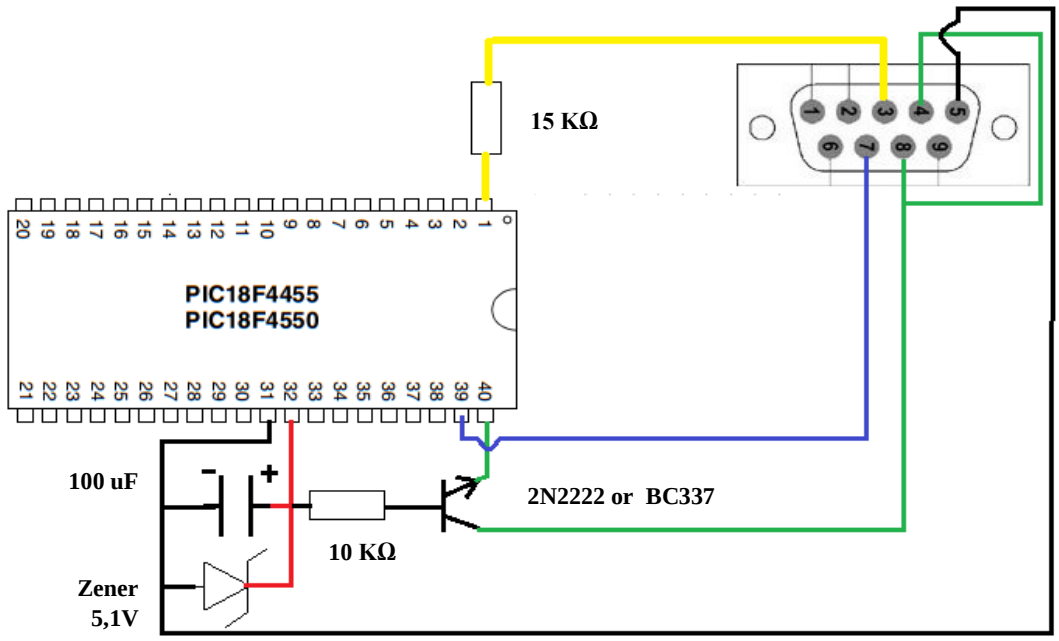
المنفذ التسلسلي هنا لا يستعمل في وضع الاِتصال التسلسلي الاِعتيادي له؛ أي USART (حيث RxD و TxD يستخدمان بشكل عادي في نقل البيانات) [7] وإنما يتم التحكم في أطراف المنفذ لإجراء بروتوكولات البرمجة التسلسلية عليه، لهذا لا يمكن بأي شكل أن تعمل هذه المبرمجة باِستعمال محول USB/Serial [5]، ذلك لأن هذا الأخير يحول بروتوكول الاِتصال التسلسلي USART إلى بروتوكول USB.

تستخدم المبرمجة 5 أطراف للاِتصال مع المتحكم وهي MCLR/Vpp، PGD، PGC، Vdd و Vss لذا فهي تعمل بطريقة البرمجة بالجهد العالي.

تتنوع تصاميم المبرمجة JDM وتتعدد، ونحن جربنا العديد من المبرمجات واِخترنا أبسط مبرمجة وجدناها من حيث عدد العناصر الإلكترونية وتعقيد الدارة.

IV-5-2- دارة المبرمجة JDM :

يوضح الشكل(IV-25) مكونات الدارة : مقاومة 15KΩ، مقاومة 10KΩ، ترانزستور 2N2222 أو BC337، مكثفة مستقطبة بسعة 100uF وصمام زينر Zener Diode ذو جهد 5,1V.



الشكل IV-25-دارة المبرمجة JDM.

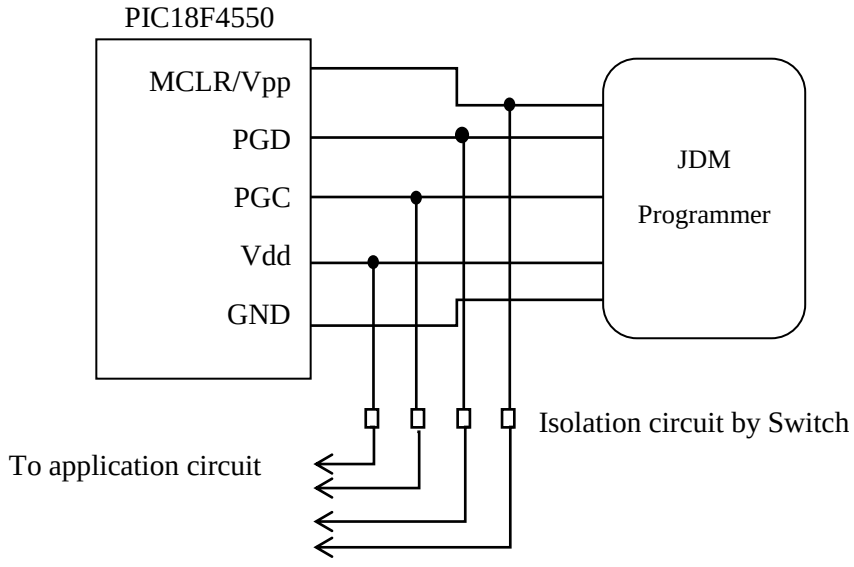
يتم ربط المتحكم المراد برمجته بموصِّل DB9 Female للاِتصال التسلسلي، بحيث يوصل كل طرف كالتالي [8]:

- الطرف MCLR/Vpp يوصل مع مقاومة 15KΩ الموصولة هي الأخرى بالطرف رقم 3 للموصل.
- الطرف PGC يربط مباشرة مع الطرف رقم 7 للموصل DB9.
- الطرف PGD موصول بباعث Emitter الترانزستور، والجامع Collector موصول مباشرة بالطرفين 8 و4 للموصل DB9.

- قاعدة Base الترانزستور موصولة بمقاومة 10KΩ والتي بدورها موصولة بالطرف Vdd للمتحكم.
- توصل المكثفة بحيث يكون قطبها الموجب مع الطرف Vdd للمتحكم وقطبها السالب مع الطرف Vss للمتحكم.
- يوصل الصمام زينر على عكس المكثفة أي يكون قطبه السالب مع Vdd وقطبه الموجب مع Vss.
- النقطة المشتركة بين الصمام والمكثفة والمقاومة والموصولة بالطرف Vdd للمتحكم توصل بدورها مباشرة بالطرف 5 للموصل DB9 .

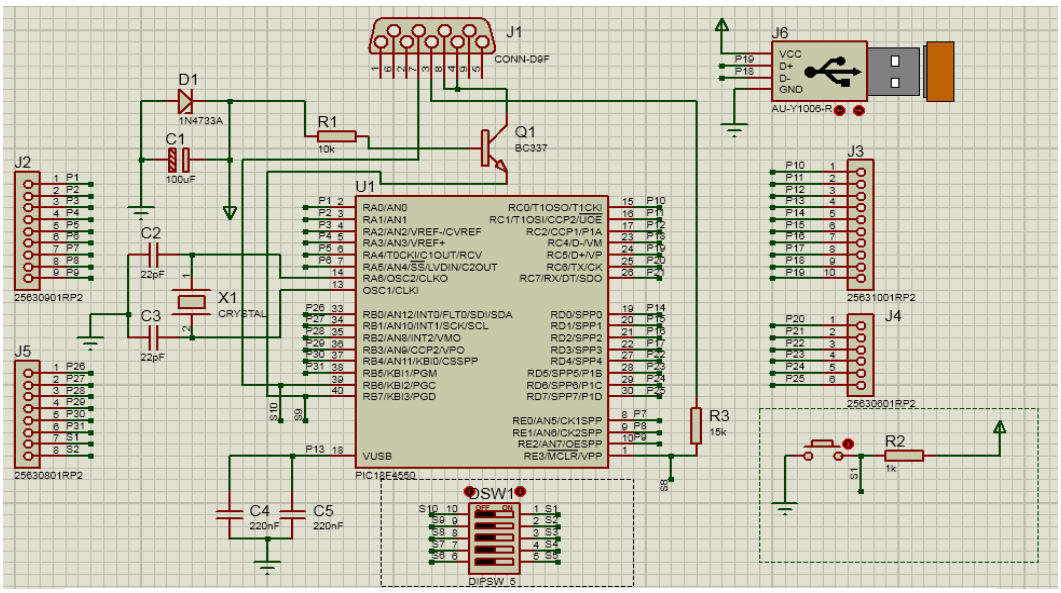
IV-5-3-دمج المبرمجة داخل اللوحة التطويرية:

قمنا بربط المبرمجة مع المتحكم بشكل عادي. الأطراف الموصولة بها PGD، وPGC، وMCLR، وVDD توَصَّل بدورها إلى مفتاح Switch؛ هذا الأخير هو الذي يعزل هذه الأطراف عن كل ما هو موصول بها عدا المبرمجة. كما يوضحه الشكل(IV-26)

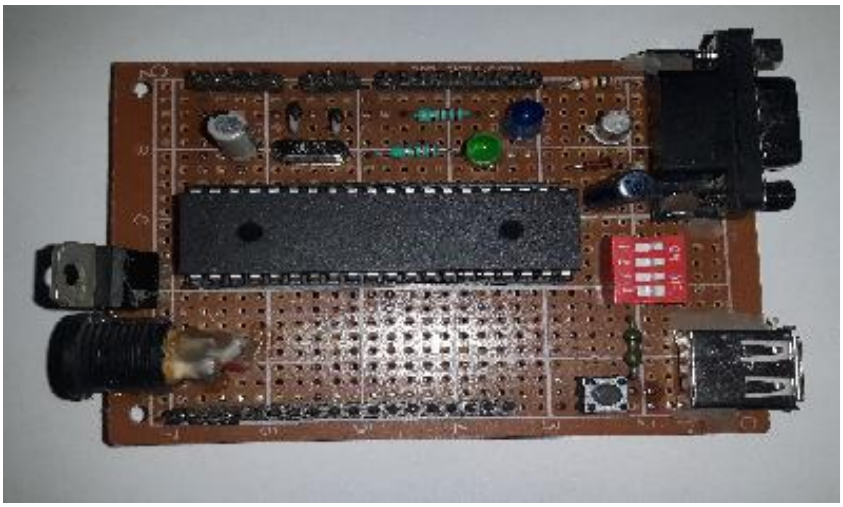


الشكل4-26-عزل أطراف المبرمجة JDM بواسطة مفتاح.

يمثل الشكل(IV-27) نموذجا للوحة التطويرية على برنامج Proteus 8.



الشكلIV-27-نموذج للوحة التطويرية الثانية بواسطة برنامج Proteus.



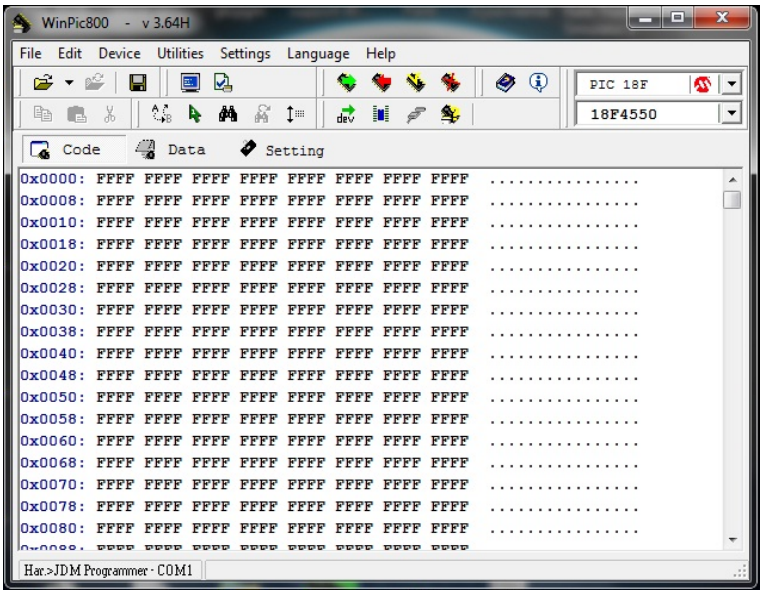
الشكلIV-28- صورة للوحة التطويرية الثانية.

IV-5-4-استعمال اللوحة التطويرية:

يتم تحميل الملف البرمجي على المتحكم بواسطة أحد التطبيقات البرمجية المتوافقة مع المبرمجة JDM. ونحن اِخترنا من بينها التطبيق WinPic800 للعمل عليه، نظرا لبساطة واجهته وسهولة اِستخدامه، ومن التطبيقات البرمجية الأخرى: ICProg و PicPgm...

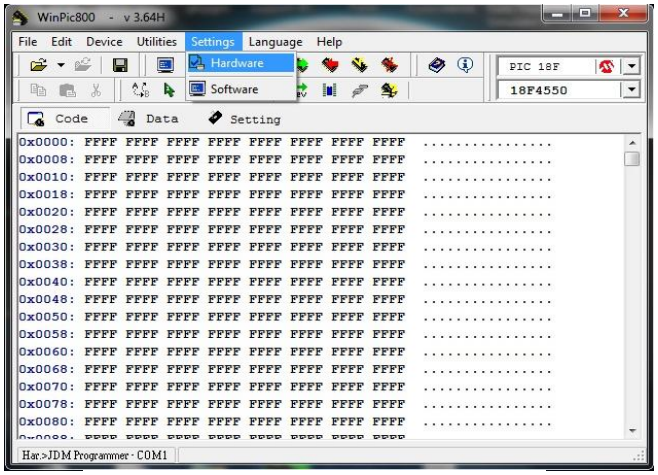
وهذا مثال عن كيفية العمل على برنامج WinPic800 :

1/ عند فتح البرنامج تظهر لنا الواجهة التالية: الشكل(IV-29)



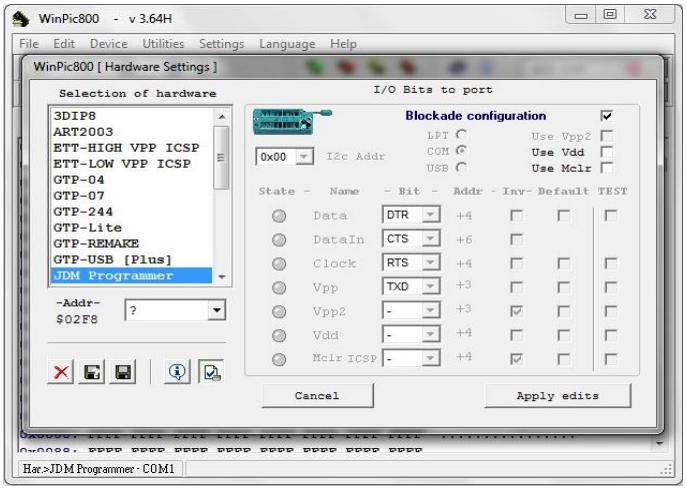
الشكلIV-29- واجهة برنامج WinPic800.

2/ أولا نقوم بضبط الإعدادات لكي يتعرف البرنامج على المبرمجة JDM.



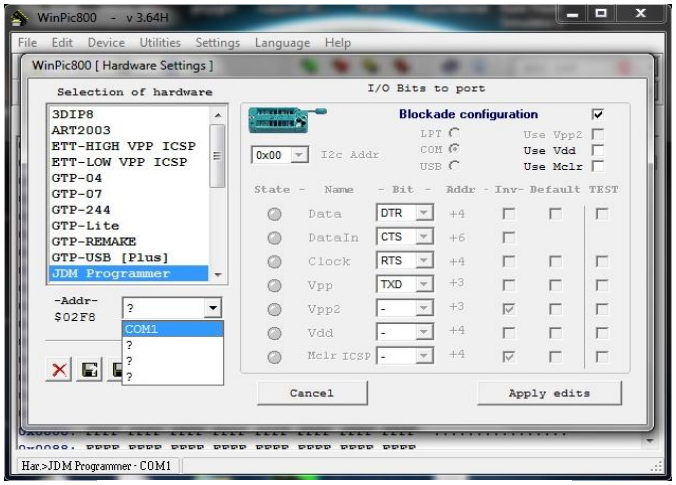
الشكلIV-30- فتح نافذة إعدادات

3/ نختار المبرمجة JDM من قائمة اختيار الأجهزة. الشكل(IV-31).



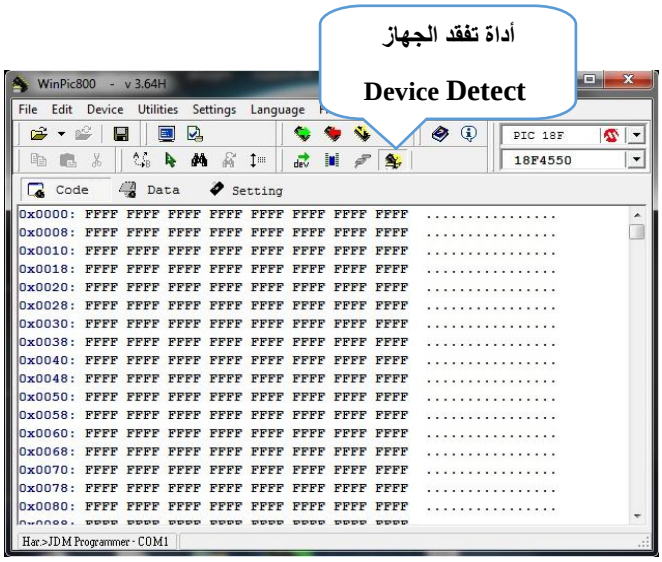
الشكلIV-31- نختار المبرمجة JDM من قائمة الأجهزة.

نختار المنفذ التسلسلي المتصل بالمبرمجة؛ هنا اخترنا COM1. الشكل(IV-32).



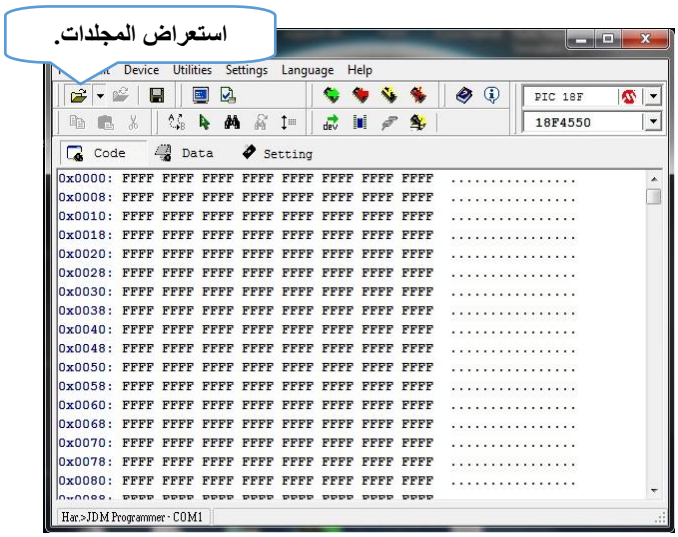
الشكلIV-32-نختار المنفذ COM الموصول بالمبرمجة Port.

5/ الآن، نقوم بتفقد الجهاز لنعلم هل المتحكم موجود على اللوحة التطويرية أو لا. بواسطة الأداة المبينة في الشكل(IV-33).



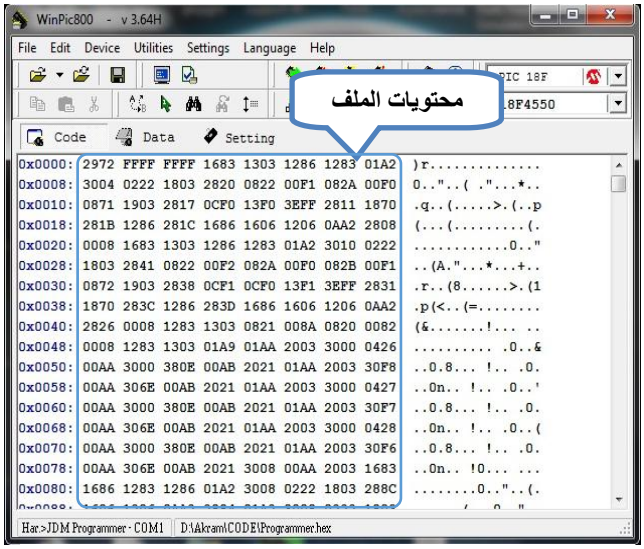
الشكلIV-33- أداة تفقد الجهاز من شريط الأدوات.

6/ الآن، نقوم بفتح الملف البرمجي الذي نريد إرساله إلى ذاكرة المتحكم عن طريق الأداة كما في الشكل(IV-34).



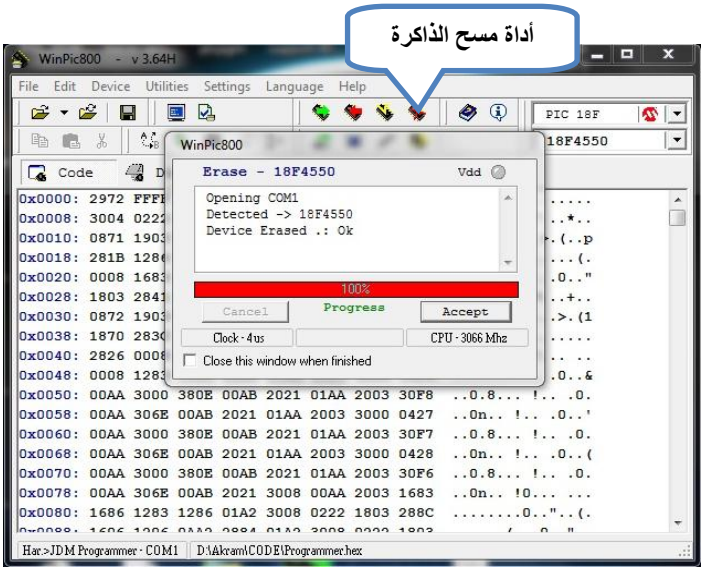
الشكلIV-34- أداة استعراض المجلدات.

7/ يمثل الشكل (IV-35) فتح الملف البرمجي بواسطة WinPic800.



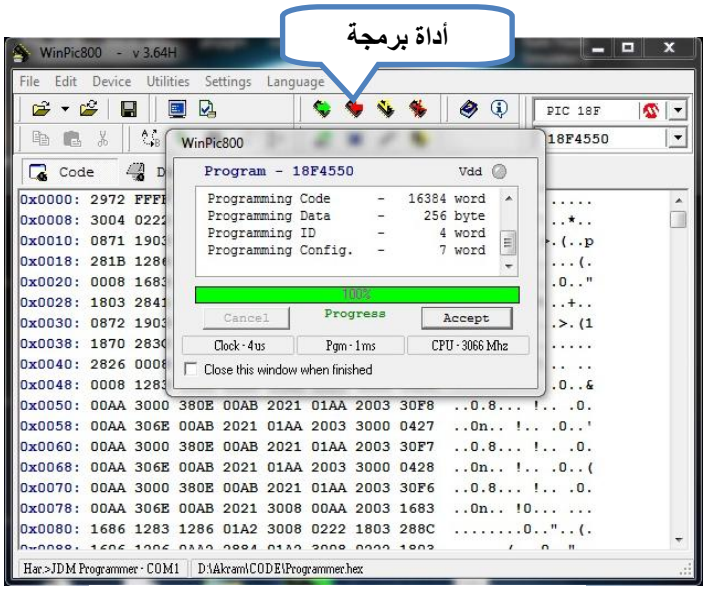
الشكلIV-35- عرض محتويات الملف البرمجي.

8/ نقوم بمسح ذاكرة المتحكم قبل نقل الملف البرمجي إليها. بواسطة الأداة الموضحة في الشكل(IV-36).



الشكلIV-36- مسح ذاكرة المتحكم.

9/ الآن، نقوم بإرسال الملف البرمجي إلى المتحكم.



الشكلIV-37- برمجة المتحكم.

IV-6-اللوحة التطويرية الثالثة:

في هذه اللوحة اِعتمدنا على متحكم دقيق يقوم مقام الدارة المبرمجة، حيث إننا قمنا بإجراء جميع بروتوكولات البرمجة عن طريق هذا المتحكم.

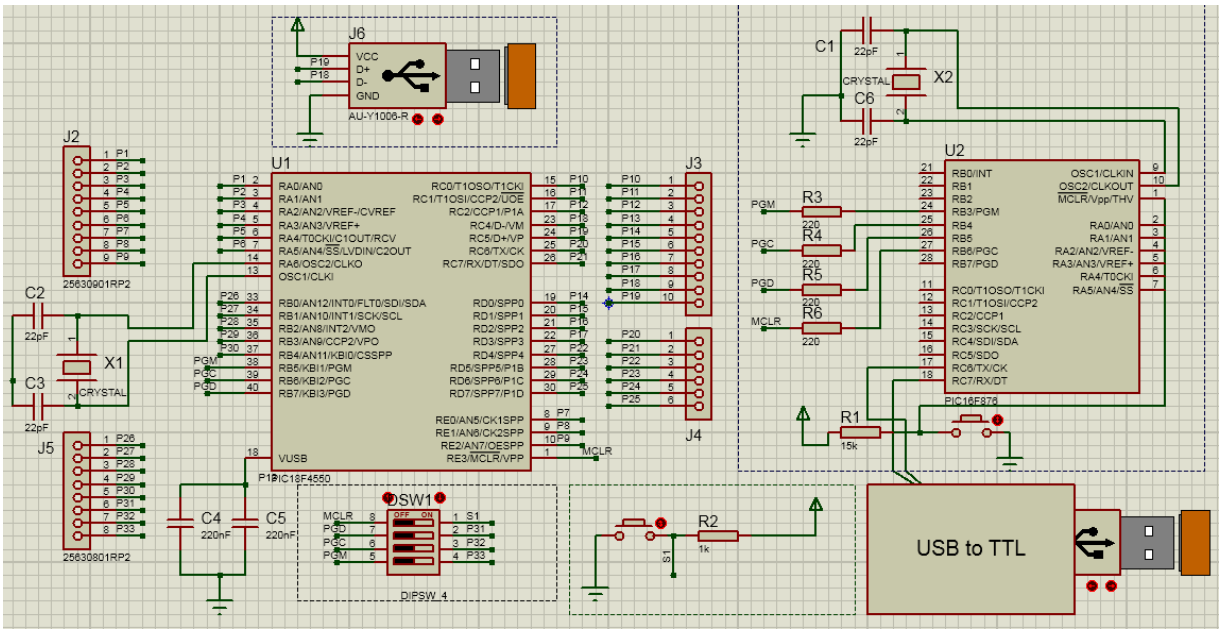
في هذه الدارة اِستبدلنا المبرمجة JDM التي لا تعمل إلا مع الموصل DB9 بمتحكم دقيق يتصل بالحاسوب عبر منفذ USB ويقوم بإجراء بروتوكولات تقنية البرمجة بالجهد المنخفض على المتحكم PIC18F4550 وهذا أيضا يسمح لنا باِستغلال كامل لمساحة ذاكرة البرنامج من أجل التطبيق البرمجي خلاف تقنية برنامج التحميل.

IV-6-1-دارة اللوحة التطويرية:

تتكون الدارة أساسا من متحكم دقيق PIC16F876 ومحول USB/TTL – يقوم بتحويل البروتوكول UART إلى USB والعكس- ومفتاح لعزل أطراف البرمجة عن باقي الدارة، إضافة إلى المتحكم الدقيق PIC18F4550 وباقي المكونات من دارة تصفير ودارة هزاز الكريستال، ومنفذ للاتصال USB.

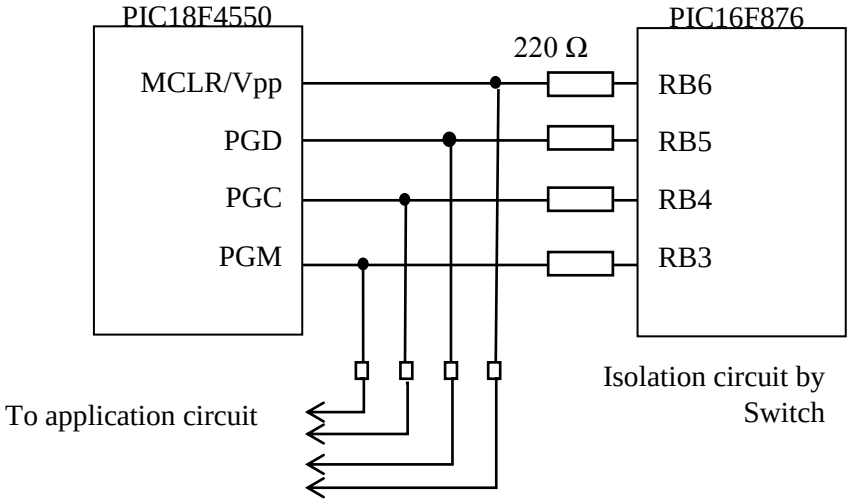
استعملنا في عملية البرمجة متحكم PIC16F876 ومحول USB/TTL بدلا من متحكم يدعم الإتصال USB؛ وذلك لكون المتحكمات من عائلة 18F غالية الثمن إضافة إلى عملنا على البروتوكول RS-232 في التطبيق البرمجي الذي طورناه بلغة C للاِتصال باللوحة التطويرية.

يمثل الشكل(IV-38) نموذجا لدارة اللوحة التطويرية المنجزة على برنامج Proteus 8.

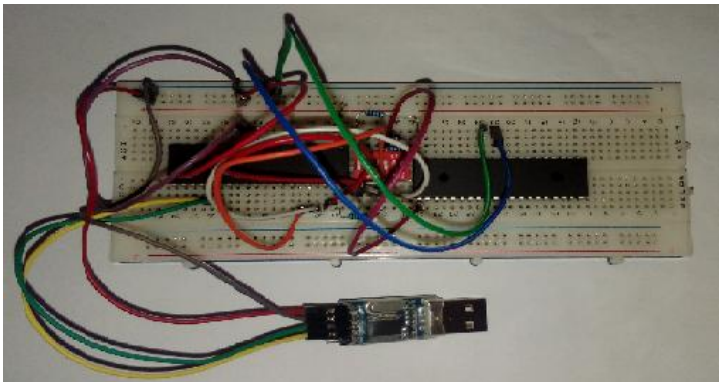


الشكلIV-38- نموذج للوحة التطويرية الثالثة على Proteus.

ويمثل الشكل(IV–39) توضيحا للأطراف المعزولة أثناء عملية البرمجة.



الشكلIV-39- عزل أطراف البرمجة بواسطة مفتاح.

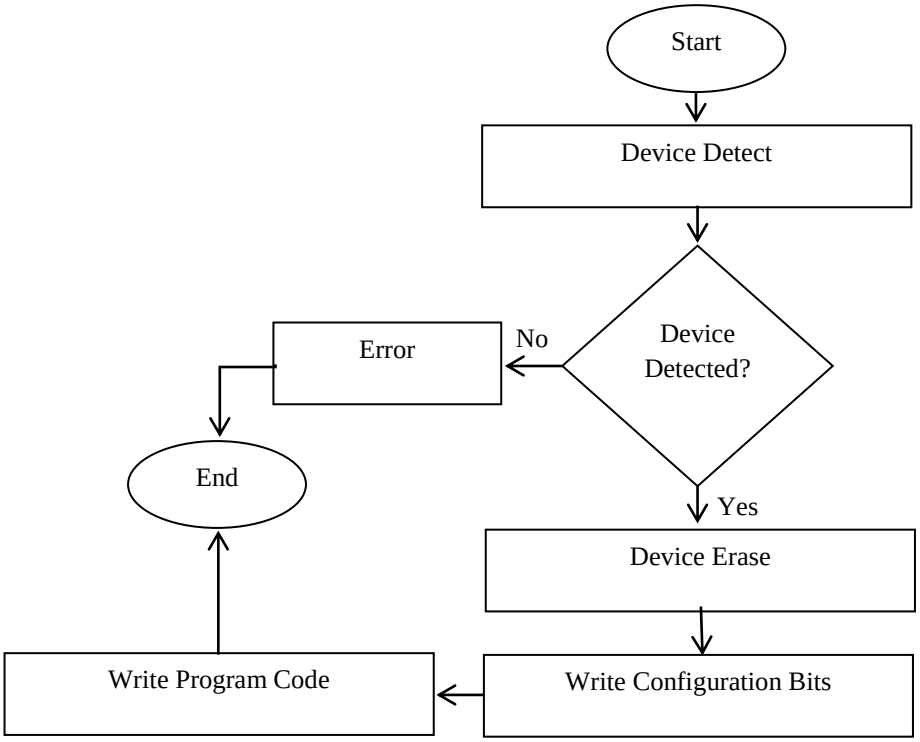


الشكلIV-40- صورة للوحة التطويرية الثالثة.

IV-6-2-آلية برمجة المتحكم على اللوحة التطويرية:

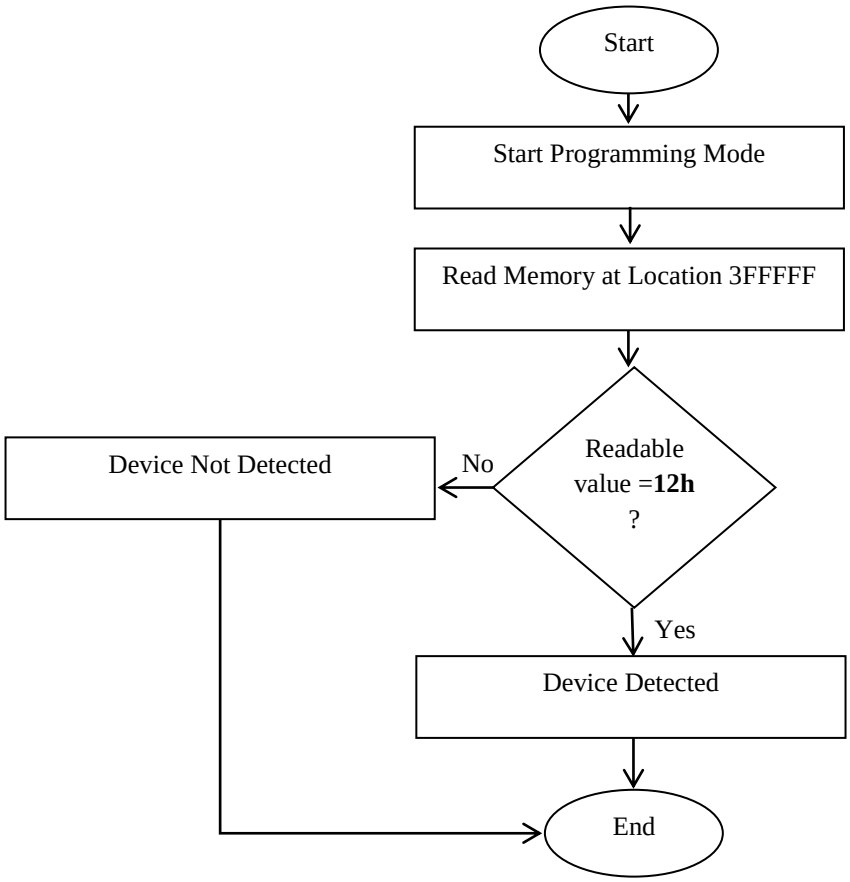
قمنا ببرمجة المتحكم PIC16F876 ليقوم بتنفيذ وإجراء بروتوكول البرمجة L.V.ICSP من خلال الأرجل الأربعة المُشار إليها في الشكل(IV-38).

ويمثل الشكل(IV-41) مخططا لمراحل عملية البرمجة.



الشكلIV-41- مخطط مراحل عملية البرمجة.

يمثل الشكل(IV-42) مخططا لمراحل عملية تحديد الجهاز (المتحكم) وذلك بقراءة رقم تعريفه المتمثل في 12h بالنظام الستا عشري.



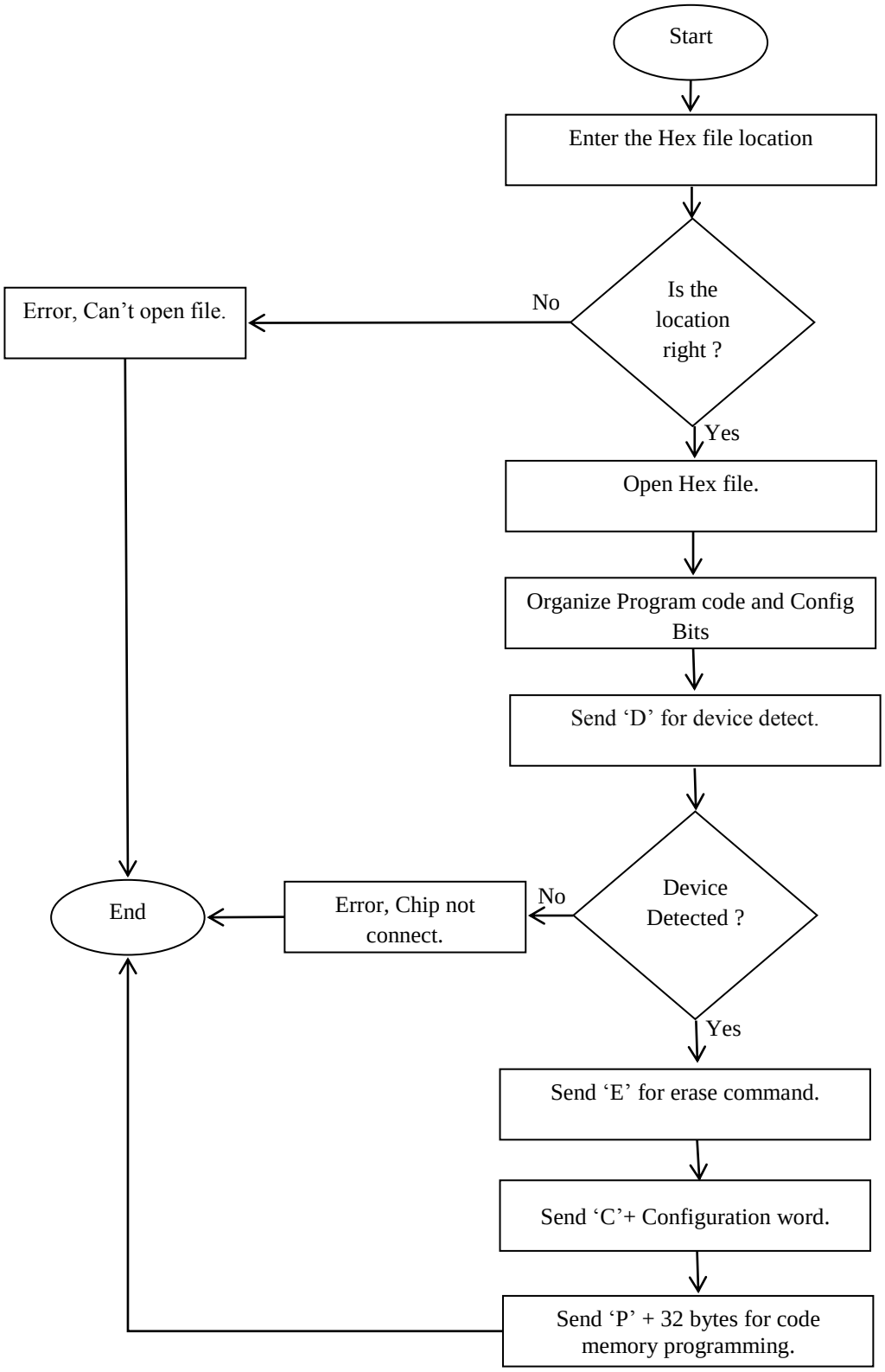
الشكلIV-42- مخطط عملية تحديد المتحكم.

IV-6-3-التطبيق البرمجي The Application:

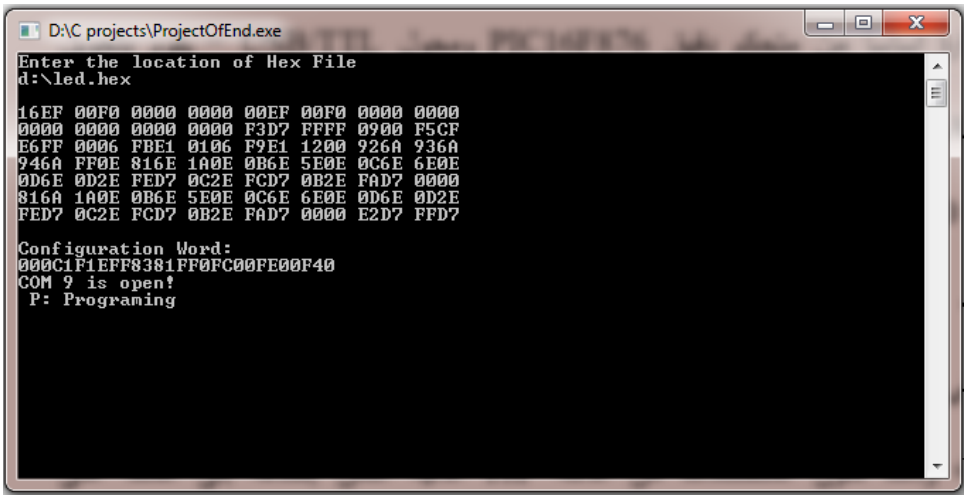
قمنا بتطوير تطبيق برمجي اِعتمادا على لغة C؛ هذا التطبيق يعمل على فتح الملف البرمجي وإرسال بياناته إلى اللوحة التطويرية عن طريق بروتوكول الإتصال RS-232 وعبر المنفذ USB للمحول USB/TTL.

الملف البرمجي مكتوب بالتشفير السادس عشر Hex وهو يحتوي على برنامج التطبيق Application Program وعلى بتات الإعدادات وعلى مجموعة من الشفرات الخاصة بنظام التشفير المتبع من طرف برنامج MikroC؛ لذا وجب علينا فصل شفرات برنامج التطبيق وشفرات الإعدادات عن باقي الشفرات.

يمثل الشكل(IV-43) مخططا لمراحل عمل التطبيق البرمجي.



الشكلIV-43- مخطط لعمل التطبيق البرمجي.

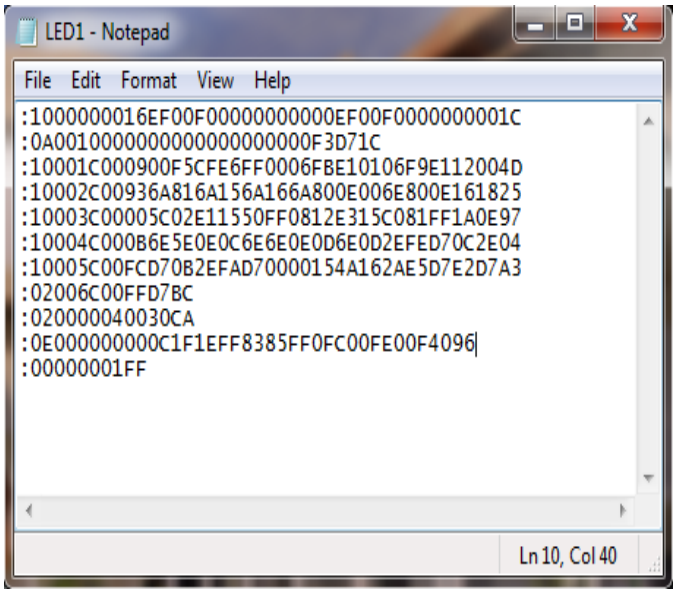


الشكلIV-44- واجهة التطبيق البرمجي.

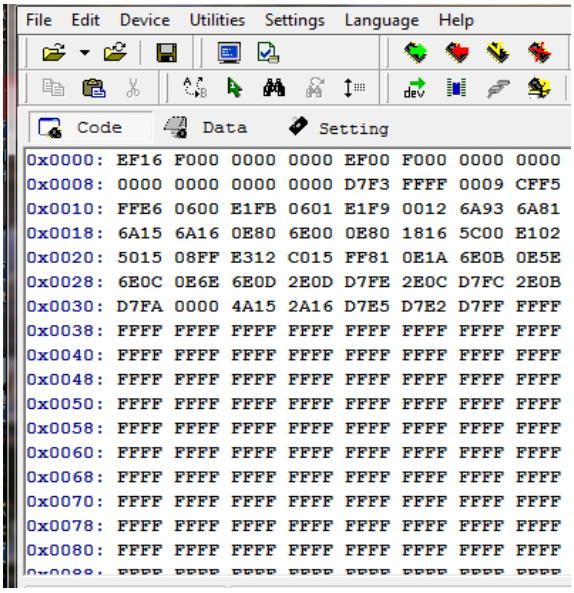
IV-6-4-تحليل شفرات الملف البرمجي:

بعد بحث ومقارنة للشفرات الموجودة على الملف البرمجي عند فتحه بتطبيق Notpad وعند فتحه بتطبيق Winpic800 إستطعنا تمييز شفرات برنامج التطبيق وشفرات الإعدادات؛ وذلك أن تطبيق Winpic800 يقوم بعرض شفرات برنامج التطبيق فقط مع الإعدادات. كما قد إستعنَّا بما وجدنا على بعض المواقع في ما يخص هذا الأمر[10].

الشكلان(IV-45) و (IV-46) يوضحان مقارنة لنفس الملف عند فتحه بالتطبيقين السابقين.



الشكلIV-46- قراءة Notpad للملف البرمجي.



الشكلIV-45- قراءة Winpic800 للملف البرمجي.

الشفرتان الأوليتان في كل سطر بعد النقطتين عدا السطرين الأخيرين هي شفرات خاصة بنظام تشفير معين؛ الأربع شفرات التي بعدها هي شفرات لعنوان أول بايت موجود في السطر؛ الشفرتان التاليتان مضافتان، والشفرات التي بعدها ماعدا الأخيرة وما قبل الأخيرة هي شفرات لبرنامج التطبيق.

مثال: تحليل السطر التالي من الملف البرمجي: :0A0010000000000000000000F3D71C

:0A	0010	00	00000000000000000F3D7	1C
شفرات خاصة.	عنوان أول بايت في السطر	شفرات خاصة.	شفرات الملف البرمجي	شفرات خاصة.

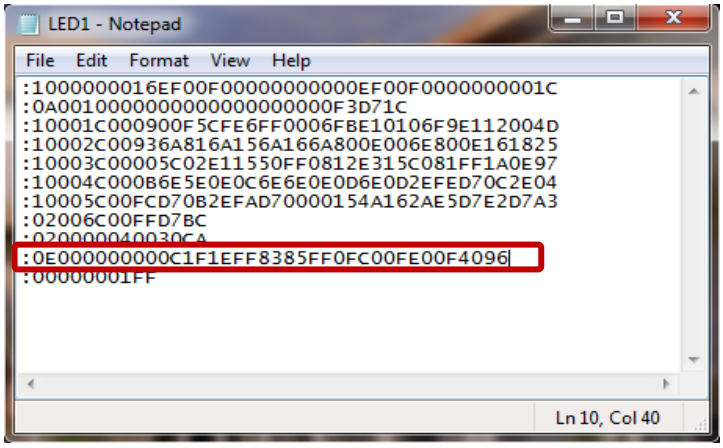
الشكلIV-47-تحليل أحد أسطر الملف البرمجي.

شفرات الإعدادات تكون في السطر ما قبل الأخير وهذا خاص بالمتحكمات 18F؛ حيث تتواجد في السطر بدءا من الشفرة رقم 9 على اِمتداد 14 بايت أي أن الشفرتين الأخيرتين غير معنيتين.

:0E000000	000C1F1EFF8385FF0FC00FE00F40	96
شفرات خاصة.	شفرات الإعدادات	شفرات خاصة.

الشكلIV-48- تحليل أحد سطر الإعدادات.

مثال: يوضح الشكل(IV-49) سطر شفرات الإعدادات.



الشكلIV-49- سطر بيانات الإعدادات.

IV-7-خاتمة:

بينا في هذا الفصل 3 طرق للبرمجة؛ اِثنان منها للبرمجة التسلسلية؛ الأولى هي البرمجة بالجهد العالي والثانية هي البرمجة بالجهد المنخفض؛ إضافة إلى الطريقة الثالثة وهي تقنية برنامج التحميل، ومن هذا قمنا بإنجاز ثلاث لوحات تطويرية كل واحدة منها لإحدى طرق البرمجة الثلاث.

خاتمة عامة

خاتمـــة عامـــة:

قمنا في هذا المشروع بإنجاز ثلاث لوحات تطويرية للمتحكم الدقيق PIC18F4550، كل منها يعتمد على طريقة برمجة معينة بناءًا على أفكار موجودة في اللوحات المختلفة؛ فاللوحة التطويرية الأولى مشابهة لحد ما في طريقة برمجتها للوحة أردوينو؛ حيث اِستغللنا وجود أداة وتقنية MikroBootloader ضمن بيئة تطوير MikroC لتكون عملية كتابة الملف البرمجي ونقله إلى متحكم اللوحة التطويرية مباشرة عبر MikroC كما هو الحال بالنسبة لبيئة تطوير أردوينو Arduino IDE. تدعم هذه اللوحة أيضا اِستخدام جميع المتحكمات من عائلة 18F لكن بشرط اِحتوائها على برنامج التحميل.

بالنسبة للوحة الثانية، هي لوحة بسيطة تعتمد على البرمجة التسلسلية عبر منفذ DB9 بواسطة مبرمجة JDM موجودة ضمن اللوحة التطويرية. يتم التحكم في نمط عمل اللوحة إما ببرمجة أو تنفيذ التطبيقات بواسطة مفتاح Switch. تدعم هذه اللوحة استخدام جميع المتحكمات من العائلة 16F والعائلة 18F ذلك أنَّ المبرمجة المستخدمة تدعم برمجة العديد من عائلات المتحكمات PIC.

طريقة البرمجة في اللوحة التطويرية الثالثة مشابهة لمثيلتها على لوحة Launch pad في وجود متحكم دقيق يعمل على اِستقبال بيانات الملف البرمجي من الحاسوب وإجراء بروتوكول البرمجة التسلسلية؛ فهو يوفر خاصية البرمجة عبر منفذ USB دون الحاجة لبرنامج تحميل، تدعم اللوحة أيضاً اِستخدام المتحكم PIC18F2550.

اللوحات الثلاث متوافقة مع الملف البرمجي المكتوب بلغة MikroC.

نتمنى أن يكون مشروعنا هذا إسهاما بسيطا في المكتبة العلمية العربية وأن يكون عونا لطلبة العلم في مشاريعهم وتطبيقاتهم.

قائمة المراجع

الفصل الأول:

[1]: Programming language,

https://en.wikipedia.org/wiki/Programming_language

[2]: Introduction to Programming Languages,

https://www.geeksforgeeks.org/introduction-to-programming-languages/

[3]: Computer Programming Languages,
https://www.computerscience.org/resources/computer-programming-languages/

[4]: Assembly Introduction,
https://www.tutorialspoint.com/assembly_programming/assembly_introduction

[5]: Assembly language,
https://en.wikipedia.org/wiki/Assembly_language

[6]: B.W.KERNIGHAN & D.M.RITCHIE, The C Programming Language. Second Edition, AT&T Bell Laboratories Murray Hill, 1988, New Jersey

[7]: Java (programming language),
https://en.wikipedia.org/wiki/Java_(programming_language)

[8]: Java beginners guide, http://www.oracle.com/events/global/en/java-outreach/resources/java-a-beginners-guide-1720064.pdf

[9]: علي أبو سيفين، تعلم لغة جافا للمبتدئين ، الفريق الفلسطيني لأنظمة المعلومات،2018

[10]: cysboy et John-John, Apprenez à programmer en Java, www.openclassrooms.com/

[11]: Simulation, https://en.wikipedia.org/wiki/Simulation

[12]: M. SHENOY, Introduction to Simulink in MATLAB, https://electrosome.com/introduction-simulink-matlab/, June 6, 2013.

[13]: T. ATKINS, Introduction to Simulink, MathWorks, 2010.

[14]: Circuit Simulation Software, www.labcenter.com/.

[15]: Proteusمقدمه عن برنامج المحاكاة بروتس,

http://eng-mega-ali-kamel.blogspot.com/2017/09/proteus-professional-proteus-proteus.html

[16]: Compiler, https://en.wikipedia.org/wiki/Compiler

[17]: M.H. FERRARI, Automata Theory, Languages and Computation.

[18]:A. AQEEL, Introduction to Arduino IDE, https://www.theengineeringprojects.com/2018/10/introduction-to-arduino-ide.html ,October 3, 2018.

[19]: Arduino IDE, https://en.wikipedia.org/wiki/Arduino_IDE

[20]: mikroElektronika, MicroC PRO for PIC User Manual.

[21]: MPLAB, https://en.wikipedia.org/wiki/MPLAB.

[22]: Description & Caractéristiques des kits de développement, https://www.electronique-mixte.fr/kits-de-developpement-processeurs-et-microcontroleurs/

[23]: Texas Instruments, MSP-EXP430G2 LaunchPad Development Kit, User's Guide.

[24]: 2014 ,عبد الله علي عبد الله, راسبيري باي ببساطة

[25]: mikroElektronika, Ready for PIC(DIP40) Manual.

الفصل الثاني:

[1]: Microcontrollers ,لجنه الدعم الفني لفريق معرض الأزهر للتطبيقات الهندسية, AZEX_2015.

[2]:مصطفى حاج عبد الرحمن, تطبيقات الاتصالات التسلسلية بإستخدام المايكروكنترولر دراسة بنية المتحكمات من عائلةPIC18 , 2013, جامعة حلب.

[3]: K.M. BHURCHANDI & A.K. RAY, Advenced Microprocessors and Peripherals, Tata Mc Graw Hill Education, 2013.

[4]: عبد الله على عبد الله, تعلم AVR ببساطة, 2015

[5]: S. BERNAOUI, Etude et réalisation d'un kit de développement d'application à base de pic de la famille 16Fxxx, Mémoire de Master, Universités de Biskra,2012.

[6]: حمدي سلطان عبد الخالق, ميكروبيديا موسوعة الميكروكنترولر, شركة أنورالجمال,2012

[7]: Microchip, PIC18F2455/2550/4450/4550 Data Sheet.

[8]: C. SANTORO, Using the digital I/O interface of Microchip PIC18F Microcontrollers, ARSLAB. Dipartimenato di Matematica e Informatica – universita di Catania, ITALY.

[9]:http://sadamkhaled.weebly.com ,Pic المهندس صدام خالد, دروس المتحكم الدقيق

الفصل الثالث:

[1]: Communication, http://students.iitk.ac.in/eclub/

[2]: RS-232, https://en.wikipedia.org/wiki/RS-232

[3]: RS-232 INTERFACE, http://www.tscm.com/

[4]: B. BUCHANAN, Handbook of Data Communications and Networks, Springer Science+Business Media Dordrecht, 1999.

[5]: S. BERNAOUI, Etude et réalisation d'un kit de développement d'application à base de pic de la famille 16Fxxx, Mémoire de master, Universités de Biskra, 2012.

[6]: Simplex, Half Duplex, Full Duplex, https://teachcomputerscience.com/simplex-half-duplex-full-duplex/

[7]: serial communication standards, http://www.mit.bme.hu/eng/system/files/oktatas/targyak/8449/serial_prot_eng.pdf

[8]: مصطفى حاج عبد الرحمن, تطبيقات الاتصالات التسلسلية بإستخدام المايكروكنترولر دراسة بنية المتحكمات من عائلةPIC18 , 2013, جامعة حلب.

[9]:RS232, http://iraqembededsystems.blogspot.com/2015/01/rs232.html.

[10]:USB, https://en.wikipedia.org/wiki/USB.

[11]: R. MURPHY, USB 101: An Introduction to Universal Serial Bus 2.0, https://www.cypress.com/

[12]: Universal Serial Bus Specification Revision 1.1, Compaq Computer Corporation, Intel Corporation, Microsoft Corporation, NEC Corporation, September 23 1998.

الفصل الرابع:

[1]: Microchip PIC18F2xxx/4xxx FAMILY, Flash Microcontroller Programming Spécification.

[2]: J. BENINGO. Bootloader Design for Microcontrollers in Embedded Systems, Beningo Engineering.

[3]: mikroElektronika, USB HID mikroBootloader Manual.

[4]: mikroElektronika, PIC18F USB HID Bootloader Memory Layout.

[6]: H. HAMED, Simple JDM PIC Programmer, https://www.instructables.com/id/Simple-JDM-PIC-Programmer/

[7]: https://arabteam2000-forum.com/index.php?/topic/211939-مع-البيك-كيفية-صناعة-مبرمجة-سيريال-مرحلة-الانتقال-الى-الجانب-العملي/

[8]: R. KASTIEL, Simple serial port programmer (JDM programmer), https://rado.heliohost.org/content-6.html, 2015.

[9] :H. SOLTAN. Lec 38 Programmer For PIC16F877A Part1,

https://www.youtube.com/watch?v=7KOXtjZ3K4g&list=PLjDHaI4HQzTy6lzhYFWI73lilBGPx5c27&index=3, Jan 29. 2012.

[10]: K. KULAKOV, Arduino as a Pic18F programmer.

https://sites.google.com/site/thehighspark/arduino-pic18f.

الملاحق

الملاحق

الملحق1 : قيم الأزمنة والجهود في عملية البرمجة بالجهد العالي والمنخفض:

Standard Operating Conditions
Operating Temperature: 25°C is recommended

Param No.	Sym	Characteristic	Min	Max	Units	Conditions
D110	VIHH	High-Voltage Programming Voltage on MCLR/VPP/RE3	VDD + 4.0	12.5	V	**(Note 2)**
D110A	VIHL	Low-Voltage Programming Voltage on MCLR/VPP/RE3	2.00	5.50	V	**(Note 2)**
D111	VDD	Supply Voltage During Programming	2.00	5.50	V	Externally timed, Row Erases and all writes
			3.0	5.50	V	Self-timed, Bulk Erases only **(Note 3)**
D112	IPP	Programming Current on MCLR/VPP/RE3	—	300	µA	**(Note 2)**
D113	IDDP	Supply Current During Programming	—	10	mA	
D031	VIL	Input Low Voltage	VSS	0.2 VDD	V	
D041	VIH	Input High Voltage	0.8 VDD	VDD	V	
D080	VOL	Output Low Voltage	—	0.6	V	IOL = 8.5 mA @ 4.5V
D090	VOH	Output High Voltage	VDD – 0.7	—	V	IOH = -3.0 mA @ 4.5V
D012	CIO	Capacitive Loading on I/O pin (PGD)	—	50	pF	To meet AC specifications
P1	TR	MCLR/VPP/RE3 Rise Time to Enter Program/Verify mode	—	1.0	µs	**(Notes 1, 2)**
P2	TPGC	Serial Clock (PGC) Period	100	—	ns	VDD = 5.0V
			1	—	µs	VDD = 2.0V
P2A	TPGCL	Serial Clock (PGC) Low Time	40	—	ns	VDD = 5.0V
			400	—	ns	VDD = 2.0V
P2B	TPGCH	Serial Clock (PGC) High Time	40	—	ns	VDD = 5.0V
			400	—	ns	VDD = 2.0V
P3	TSET1	Input Data Setup Time to Serial Clock ↓	15	—	ns	
P4	THLD1	Input Data Hold Time from PGC ↓	15	—	ns	
P5	TDLY1	Delay Between 4-Bit Command and Command Operand	40	—	ns	
P5A	TDLY1A	Delay Between 4-Bit Command Operand and Next 4-Bit Command	40	—	ns	
P6	TDLY2	Delay Between Last PGC ↓ of Command Byte to First PGC ↑ of Read of Data Word	20	—	ns	
P9	TDLY5	PGC High Time (minimum programming time)	1	—	ms	Externally timed
P10	TDLY6	PGC Low Time After Programming (high-voltage discharge time)	100	—	µs	
P11	TDLY7	Delay to Allow Self-Timed Data Write or Bulk Erase to Occur	5	—	ms	
P11A	TDRWT	Data Write Polling Time	4	—	ms	
P12	THLD2	Input Data Hold Time from MCLR/VPP/RE3 ↑	2	—	µs	
P13	TSET2	VDD ↑ Setup Time to MCLR/VPP/RE3 ↑	100	—	ns	**(Note 2)**
P14	TVALID	Data Out Valid from PGC ↑	10	—	ns	
P15	TSET3	PGM ↑ Setup Time to MCLR/VPP/RE3 ↑	2	—	µs	**(Note 2)**
P16	TDLY8	Delay Between Last PGC ↓ and MCLR/VPP/RE3 ↓	0	—	s	
P17	THLD3	MCLR/VPP/RE3 ↓ to VDD ↓	—	100	ns	
P18	THLD4	MCLR/VPP/RE3 ↓ to PGM ↓	0	—	s	