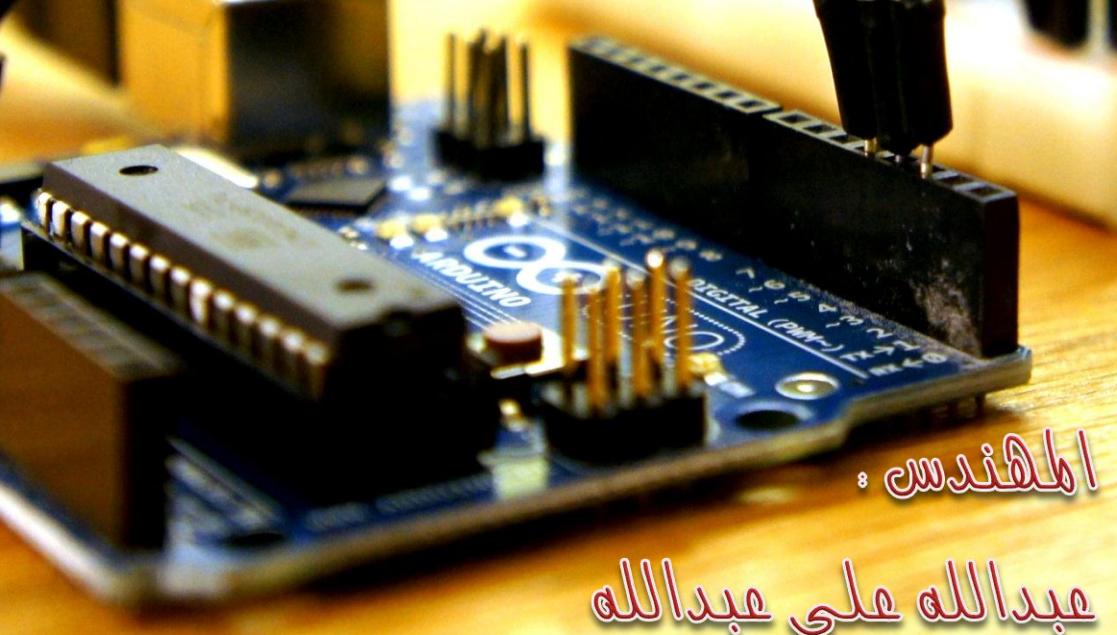


# اِردوینو بیساطة

دلیلک العملي لتعلم اساسيات  
الالكترونيات التفاعلية

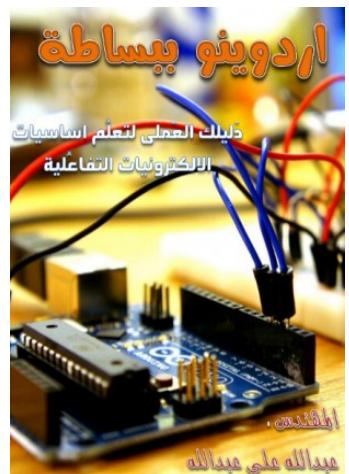
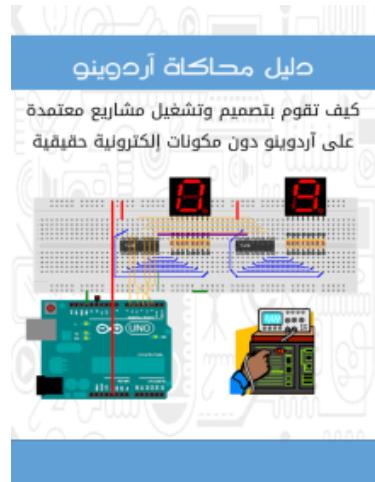
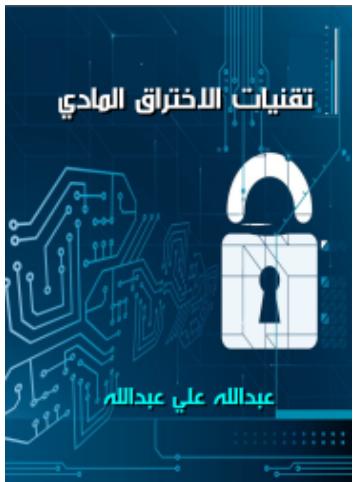


المهندس :

عبدالله على عبدالله

# سلسلة "تعلم ببساطة"

تهدف سلسلة كتب "تعلم ببساطة" إلى توفير علوم الإلكترونيات الحديثة باللغة العربية بصورة مجانية ومفتوحة المصدر مع الحفاظ على المبدأ الأساسي "البساطة" في شرح المعلومات، جميع الكتب موجهة للعامة من هواة الإلكترونيات وطلبة الكليات الهندسة.



# رخصة الكتاب

كتاب "آردوينو ببساطة" منشور مجاناً للجميع تحت رخصة المشاع الإبداعي الإصدارة الرابعةCreative Common v4 CC-NC-SA بشروط النسبة - المشاركة بالمثل - عدم الاستغلال التجاري.



**رخصة المشاع الإبداعي-CC-NC (غير تجارية)** لك كامل الحق في نسخ وتوزيع وتعديل أو الإضافة أو حتى طباعة الكتاب ورقياً كما تشاء وأشجعك على ذلك أيضاً شرط عدم إستغلال الكتاب تجارياً بأي صورة مباشرة أو غير مباشرة، كما يجوز طباعة الكتاب وتوزيعه بشكل عام شرط أن يباع بسعر التكلفة دون أي ربح.

**المشاركة بالمثل-SA** إذا تم اشتغال أي عمل من هذا الكتاب بصورة إلكترونية أو مادية مثل عمل كتاب آخر أو محاضرة تعليمية (أو حتى كورس متكامل) أو فيديو فيجب أن يتم بصورة مجانية وبنفس الرخصة (المشاع الإبداعي: النسبة، المشاركة بالمثل، الغير تجارية). يمكنك التعرف أكثر على رخصة المشاع الإبداعي من الموقع الرسمي [creativecommons.org](http://creativecommons.org)

جميع كتب سلسلة "تعلم ببساطة" منشورة بنفس الرخصة  
يمكنك تحميل السلسلة من الموقع [http://simplyarduino.com/?page\\_id=889](http://simplyarduino.com/?page_id=889)

للتواصل مع المؤلف

[abdallah.ali.abdallah.elmasry@gmail.com](mailto:abdallah.ali.abdallah.elmasry@gmail.com)

حساب Facebook

[www.facebook.com/abdallah.ali.elmasry](http://www.facebook.com/abdallah.ali.elmasry)

حساب linkedIN

[eg.linkedin.com/in/abdallahali](http://eg.linkedin.com/in/abdallahali)

عبدالله علي عبدالله

.م 1434 هـ الموافق 2012

# اهداء

الى والدى العزيز.....

الى والدتي العزيزة.....

الى كل من ساهم فـ نشر المعرفه محريه

[www.networkset.net](http://www.networkset.net)

أيمحـ التعـيمـ

[www.aabouzaid.com](http://www.aabouzaid.com)

أحمد محمد أبو زيد

[www.shabayek.com](http://www.shabayek.com)

روفـ شـبـاـيـكـ

[linuxac.org](http://linuxac.org)

مـجـتمـعـ لـينـكـسـ الـعـربـ

# فهرس الكتاب

( 1 ) .....	<b>رخصه الكتاب</b>
( 5 ) .....	<b>معلومات حول الكتاب</b>
( 7 ) .....	<b>الفصل الأول:</b> مقدمه عن اردوينو و المُتحكِّمات الدقيقة.
(21) .....	<b>الفصل الثاني:</b> التجهيزات
(43) .....	<b>الفصل الثالث:</b> ابدأ مع اردوينو
(70) .....	<b>الفصل الرابع:</b> استخدام الحساسات
(92) .....	<b>الفصل الخامس:</b> التواصل مع الحاسوب الآلى
(113) .....	<b>الفصل السادس:</b> استخدام المحرّكات
(125) .....	<b>الفصل السابع:</b> وسائل الأدخال والاخراج المتقدمه
(166) .....	<b>الفصل الثامن:</b> أغطيه اردوينو
(173) .....	<b>الفصل التاسع:</b> تاريخ تطور اردوينو
(179) .....	<b>تعلم و شارك الاخرين</b> Fablab Egypt
(183) .....	<b>ملحق المراجع:</b> مراجع (كتب + مواقع تعليميه)

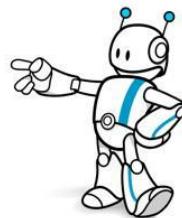
قال الله ﷺ في القرآن الكريم:

إِنَّمَا يُنذَّرُ الْمُجْرِمُونَ  
لِلَّهِ الرَّحْمَنُ الرَّحِيمُ

أَقْرَأْ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ ۝ خَلَقَ الْإِنْسَنَ مِنْ عَلْقٍ ۝  
 أَقْرَأْ وَرَبُّكَ الْأَكْرَمُ ۝ الَّذِي عَلِمَ بِالْقَلْمَنِ ۝ عَلِمَ الْإِنْسَنَ  
 مَا لَمْ يَعْلَمْ ۝

## الفئات المستهدفة من الكتاب

إلى كل من يريد تعلم تقنيات التحكم الآلي بسهولة ويسر سواء كان لك معرفه تقنيه او خبره بالمحكمات الدقيقه او حتى ما زلت في بدايه تعلم هذا المجال والى كل من لديه الشغف لمعرفه الجيل الجديد من المحكمات الدقيقه مفتوحه المصدر

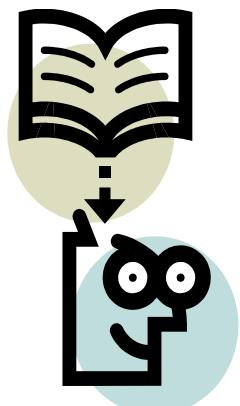


## هل احتاج الى خبرة في مجال الالكترونيات والبرمجة حتى اقرأ هذا الكتاب ؟



- بعض المعرفه بأساسيات الالكترونيات والبرمجه  
محبذه جدا لاكنها ليست شرط ضروري للتعلم فلقد  
قمت بكتابه المعلومات بأسلوب سهل و ميسر قدر  
الامكان و يمكنك اذا واجهت صعوبه ما ان تراسلني  
على البريد الالكتروني ☺

## ما الذي سوف اتعلمه في هذا الكتاب ؟؟



- كيف تصمم الالكترونيات التفاعلية بنفسك
- العمل على بيئة تطوير **اردوينو** Arduino IDE
- التحكم الالكتروني لكل ما هو حولك من اجهزه الكترونيه



**سيجعلك الكتاب** تكتسب المهارات الأولى لفهم الأكواد البرمجية المستخدمة في العديد من المشاريع التي تعتمد على [اردوينو](#) ، ولقد أضفت العديد من المصادر الخارجية التي تحتوى على العديد من المشاريع الرائعة والتي يمكنك ان تنفذها بنفسك بعد قراءة الكتاب

## كيف تم تصميم الكتاب ؟

**تم تصميم** الكتاب بأسلوب يعتمد على الشرح بالأمثله Examples و حيث يتم تقديم المعلومات على هيئه أمثله صغريه ويقدم كل مثال جزء جديد فى تعلم لغه برمجه اردوينو ، كما ستجد في نهاية كل فصل صفحه مخصصه لكي تكتب ملاحظاتك الشخصية.



اذا كنت من مستخدمي ويندوز يمكنك أن تكتب داخل ملفات الكتب الإلكترونية من نوع PDF باستخدام برنامج Foxit PDF واذا كنت تقرأ الكتاب باستخدام لوحة الكترونى Ipad مثل Tablet يمكنك استخدام برنامج Good Reader مع ملاحظه ان الكتاب متواافق مع جميع اجهزه التابلت المختلفه التي تمتلك شاسه بقياس 7 بوصه او اكبر .

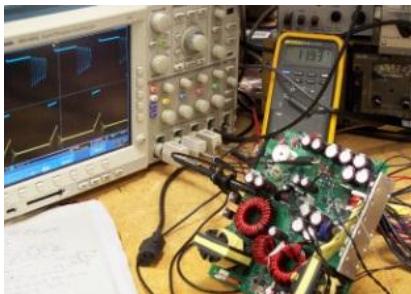
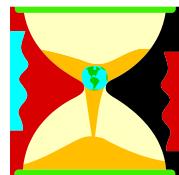
# الفصل الأول

مقدمه الى عالم اردوينو و المتحكمات الدقيقه

Introduction to Arduino & microcontrollers

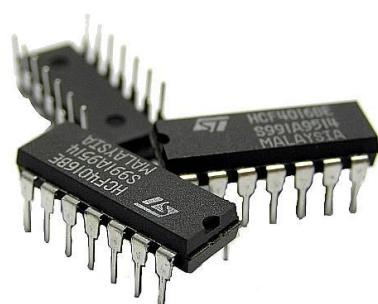


**منذ زمن** ليس بعيد كان العمل على صناعه دائره الكترونيه للقيام بوظيفه معينه يعني بناء تصميم الكتروني معقد من مكونات مثل المقاومات، المكثفات، الملفات، الترانزستور...الخ.



**كانت** الدوائر الالكترونية ثابتة التصميم و اعاده تغير او تعديل جزء بسيط فيها كان يعني الكثير من عمليات معقده مثل اللحام و قطع الاسلاك و اعاده النظر في المخططات الالكترونية والكثير من الامور المزعجه والتى ادت إلى اقتصار وظيفه تطوير المنتجات الالكترونية على مجموعه من المهندسين المتخصصين فقط.

**بفضل** التطور التكنولوجى فى مجال اشباه الموصلات و اختراع الدوائر المدمجه *Integrated Circuits* (IC) اصبح من الممكن وضع دائرة الكترونيه كامله على شريحة صغيره حجمها قد لا يتتجاوز رأس الدبوس حتى انه فى الوقت الحالى هناك دوائر الكترونيه حجمها يقدر بالنانو متر Nano Meter و التي لا يمكن روئيتها الا باستخدام مكبرات ضوئيه خاصة.



**كما ادى** تطور Integrated Circuits الى ظهور جيل خاص من الدوائر الالكترونية يسمى المتحكمات الدقيقة Micro Controllers وهى اشبه بكمبيوتر مصغر قابل للبرمجة لاداء مجموعه من الوظائف مثل قرائه درجه حرارة، التحكم فى محرك كهربى، او حتى اداره خطوط الانتاج فى المصانع الكبرى، وكل ذلك يتم ببساطه عن طريق اوامر برمجيه و بذلك تحولت تقنيه صناعه الدوائر والأنظمة الالكترونية من التصميم الالكترونيى البحث المعتمد على المكونات الصلبه فقط الى اوامر برمجيه يمكن لاي فرد ان يكتبها و يصممها بنفسه بسهوله و يسر.



**تنمية** المتحكمات الدقيقة ايضا بأمكانيه التغيير و التعديل في اي وقت ، بكل بساطه لو اردت ان تغير شيئا ما في مشروعك يمكنك ذلك بالتعديل في السطور البرمجيه و اعاده وضع الاوامر الجديده على المتحكمه الدقيقة و تجربتها اكثر من مره و هكذا الى ان تصل بمشروعك للهدف المنشود ...



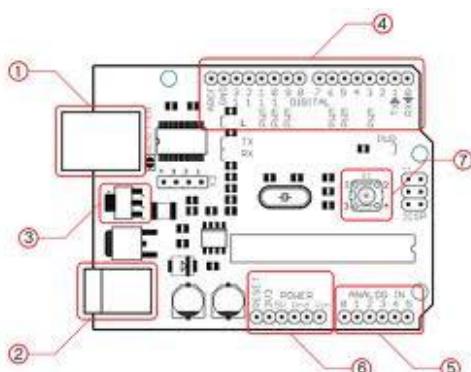
## اردوينو Arduino

هي بورده الكترونيه مفتوحة المصدر Open Hardware لتطوير الكثير من الافكار والمشاريع المتعلقة بالتحكم الآلى بصورة سهلة وبسيطة عن طريق استخدام لغه برمجه مفتوحة المصدر Arduino C يتم برمجه المتحكمه الموجوده على البورده باستخدام

**Arduino IDE: Integrated Development Environment** والذى يتوفّر مجاناً للتحميل من موقع اردوينو الرسمى <http://arduino.cc>



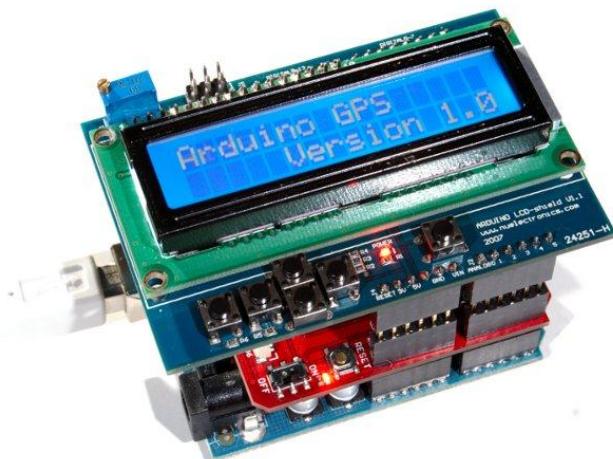
## ما معنى ان اردوينو مفتوح المصدر ؟؟



يعنى انه يمكنك الاطلاع والتعديل على التصميمات الهندسية والشفرات المصدرية لكل من بوردات **اردوينو** المختلفة Arduino Boards بما يتناسب معك ويتمكنك ايضاً تطوير لغة برمجه Arduino C بحرية تامه و الاطلاع على الشفرات المصدرية الخاصة بها

كما ان كل هذه المميزات والبرمجيات مجانيه تماماً على غرار بعض البيئات التطويريه مثل **Mikro C** والتى تتطلب منك شراء رخصه مكلفة تصل فى بعض الاحيان الى الآف الدولارات لاستخدامها.

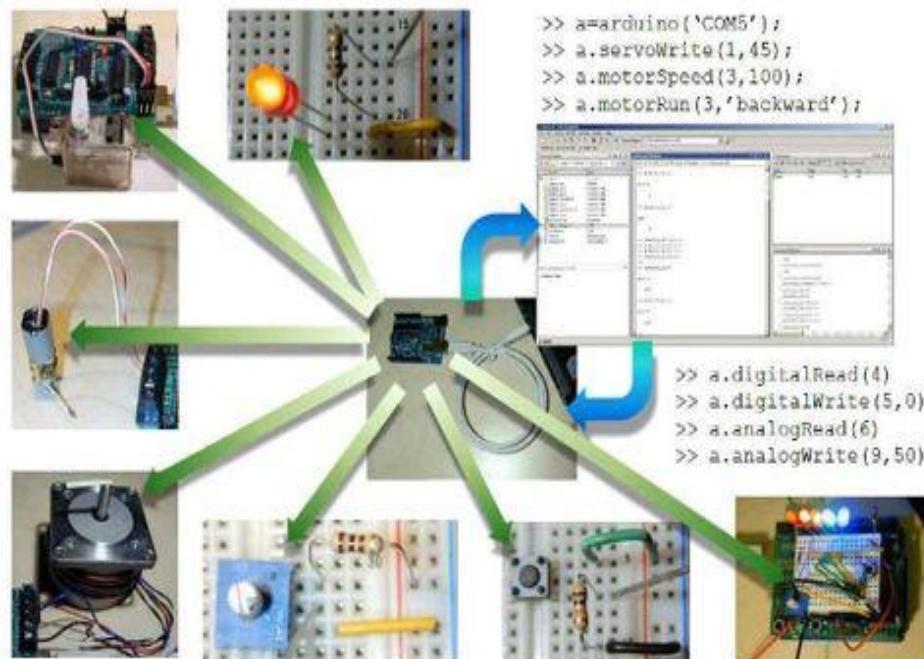
**ما يميز بوردات اردوينو** عن باقى البويرات التطويريه للمتحكمات الدقيقه الاخرى Micro Controllers Development boards هو مدى السهوله التعامل معها و بساطه اللغة البرمجيه والتى عمل فريق من ايطاليا على تطويرها منذ عام 2005 حتى الان ، ولقد تم اشتراق لغه اردوينو البرمجيه من لغه processing و لغه "سي C" والتى تعد اساس لغات البرمجه الحديثه و صاحبه ثوره تقنيه البرمجيات.



قد يظن البعض ان اردوينو مصمم للهواء فقط لakan هذا ليس صحيح لقد تم تطويره ليتناسب جميع المستويات ابتداء من الهواء و انتهاء بالمشاريع المتطورة والدليل انه هناك مميزات جباره يجعل

**اردوينو** على قمه المتحكمات الدقيقه وهى امكانيه دمجه فى مشاريع يتم برمجتها بلغات هندسيه متطوره مثل Java و لغه MATLAB حيث ستجد مكتبات برمجيه فى جاهزه للغه MATLAB والجافا خاصه بالتعامل مع **اردوينو**، وستجد في صفحة المراجع بعض الموارد التي تعلمك كيفيه استخدام الاردوينو مع المات لاب، كما يمكنك ايضا استخدام لغه Java و لغه VB.NET في التواصل مع بوردات **اردوينو** المختلفه

**على سبيل المثال:** يمكنك عمل دائرة استشعار للطقس والحرارة بالاردوينو وارسال البيانات الى MATLAB على الحاسوب الآلي لاجراء التحليلات الحسابية المتطورة المتوفّر بلغه MATLAB.



**من أكثر** المشاريع التي اثارت اعجابي كانت مستكشف زلازل باستخدام الاردوينو ولله برمجه MATLAB وكانت من تصميم شاب في السادسه عشر من عمره في الولايات المتحدة حيث قام بعمل وحدة استشعار وتحليل للزلزال بتكلفة لا تتجاوز 80 دولار باستخدام بعض المستشعرات و بورده **اردوينو** فقط مع العلم ان الدوائر الالكترونية المماثله قد تتتكلف اكثراً من 2000 دولار

## انواع بوردات اردوينو Arduino Boards



Arduino UNO, Arduino Mega, Arduino Nano, Arduino Mini, Arduino Lilypad, Arduino Demulive, Boarduino

تختلف البوردات عن بعضها البعض من ناحيه عدد المخارج والمداخل والتى تحدد عدد الاجهزه التي يمكن التحكم بها وعدد الحساسات Sensors التي يمكن دمجها مع البورده وكذلك نوع المتحكمه الدقيقه وسرعه المعالج الموجود بداخليها وامكانيه تبديلها ام لا وسوف نتناول فى هذا الكتاب بورده

**Arduino Uno**



## نظرة عامة على دائرة

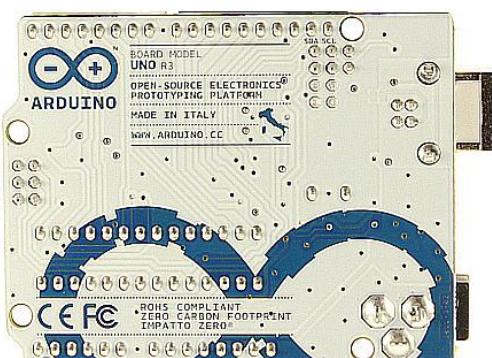
### Arduino Uno

**دائرة الكترونية** صغيرة تستخدمن فى برمجه  
متحكمه من شركه اتيل ATmega328  
توفر هذه الدائرة منافذ لتوصيل المكونات

الاكترونيه الى المتحكمه مباشره عن طريق 14 (مدخل | مخرج) من النوع الرقمي  
من هذه الـ 14 يوجد 6 يمكن استخدامها كمخارج PWM أو ما يعرف Digital In/out  
بالتعديل الرقمي المعتمد على عرض النبضة (Pulse-Width modulation) وسوف  
نتحدث عن هذه الخاصيه فى فصل كامل يسمى الدخول والخرج التمايل Analog

### Inputs & Outputs

أيضا تحوي الدائرة على مهتز كريستال  
Crystal Oscillator بتردد MHz16  
بالاضافه الى مدخل USB من أجل  
التواصل مع الحاسب، وهناك مدخل  
ICSP للطاقة منفصل ، بالاضافه الى  
header والذى يعتبر طريقه اضافيه  
لبرمجة المتحكمه وهي لا تزال موصله

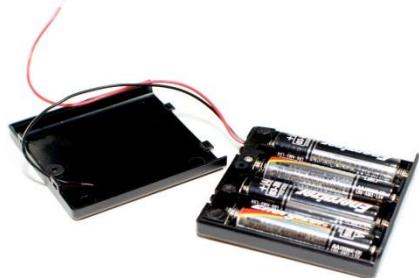
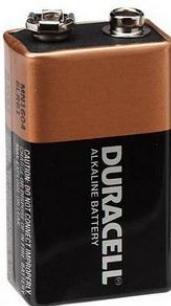


بالبورده (خلاف USB) و يمكنك ان تعتبر بورده اردوينو هذه بورده تطوير و برمجه  
صغره و مهيئه للاستخدام المباشر Development Board فهى تقريبا تحوي كل ما  
تحتاج لكي تعمل عليها سواء عن طريق منفذ USB أو عن طريق مصدر خارجي للطاقة  
مثل البطاريه.

## امداد الدائرة بالطاقة Power up



**يمكنك** امداد الدائرة بالطاقة إما من خلال منفذ USB فقط أو عن طريق استخدام مصدر خارجي للكهرباء كمحول AC\DC ليمد الدائرة بالجهد اللازم للعمل او حتى عن طريق بطارية 9 فولت او 4 بطاريات 1.5 فولت حيث يتم توصيل طرفي البطارية الى مدخل الارضي Gnd وال Vin في الدائرة.



**تستطيع** الدائرة العمل على جهد يتراوح بين الـ 6 - 20 فولت، لكن يجب الانتباه حيث انه اذا قمنا بتتأمين جهد أقل من 7 فولت فإن المخرج المتحكم Pin 5V قد لا يستطيع تأمين جهد خرج يبلغ الـ 5 فولت المطلوب وقد يؤدي الى عدم استقرار الدائرة، اما اذا قمنا بتزويد الدائرة بجهد أعلى من 12 فولت فإنه قد يؤثر على عنصر تنظيم الجهد voltage regulator و يؤدي الى ارتفاع درجة حرارته مما يؤدي الى تلف البورده ، لذا فإن مجال الجهد الذي يفضل استخدامه هو من 7 الى 12 فولت.



## مخارج و مدخل الطاقة الكهربائية للمتحكم

### Power Inputs/Outputs

**Vin** : جهد الدخل عندما نستخدم

مصدر طاقة خارجي، يمكننا تأمين الجهد من خلال هذا المدخل ، إذاً كنا نقوم بتتأمين الطاقة للدائرة من خلال مدخل المحول يمكننا الوصول له من خلال هذا المدخل أيضاً.

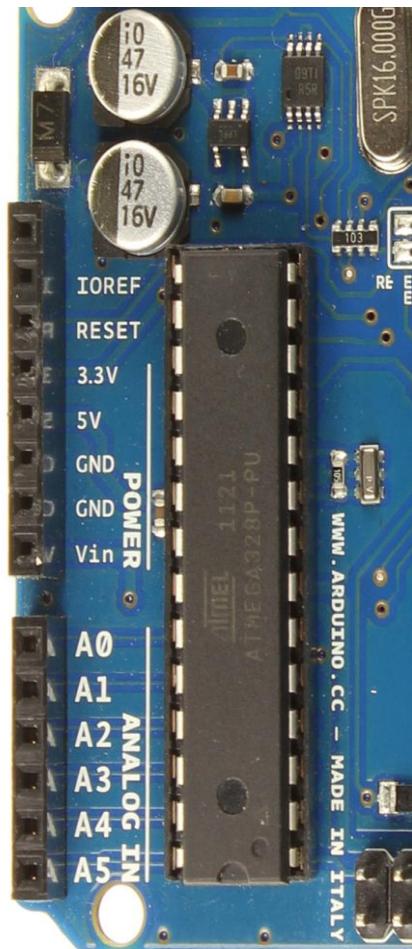
**V5** : جهد منتظم يستخدم لتتأمين الطاقة

للعناصر المستخدمة على الدائرة و سوف نستخدمه لتوفير الطاقة للقطع الالكترونيه التي سنضيفها، قد يأتي هذا الجهد من خلال Vin عبر منظم جهد داخلي أو تأمينه من خلال منفذ USB أو أي مصدر جهد منتظم بقيمة 5 فولت.

**V3.3** : مصدر للجهد بقيمة 3.3 فولت

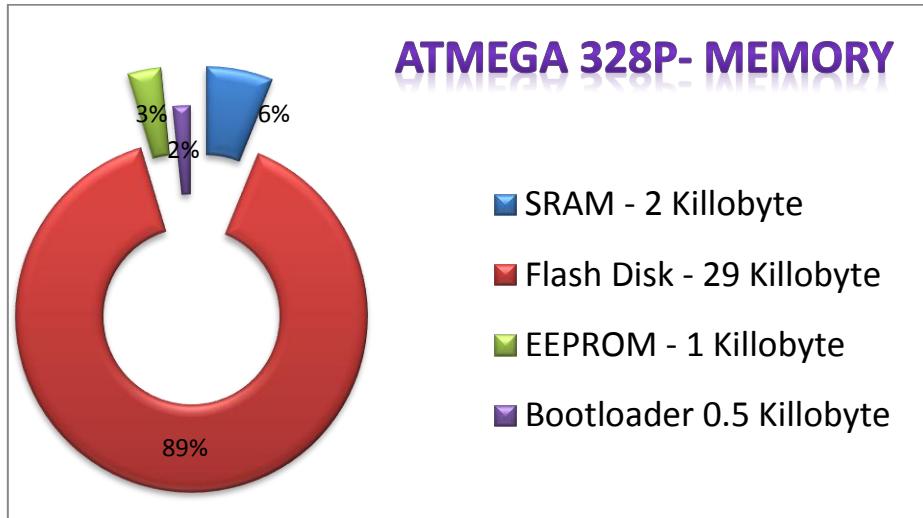
مؤمن من قبل منظم الجهد الداخلي للدائرة وأقصى قيمة لسحب التيار من خلال هذا الخط هو 50 ميلي أمبير.

**GND** : الخط الارضي.



## المعالج الدقيق و الذاكرة Microprocessor

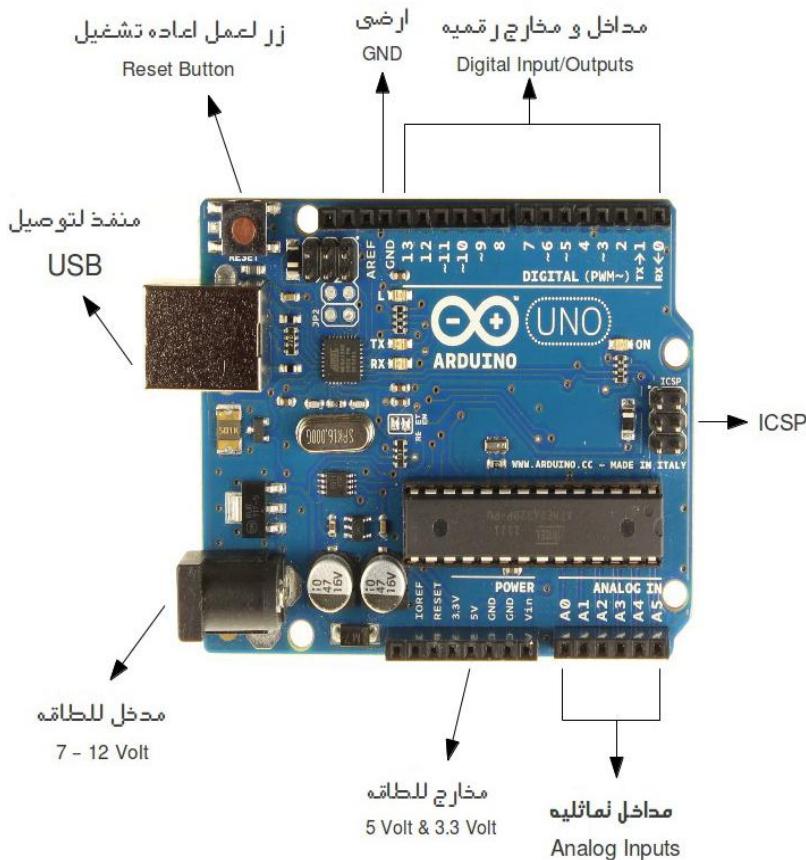
المتحكمات الدقيقة اشبه بوحده حاسب آلى صغيره الحجم وتحوي المتحكمه الدقيقه Kilo Byte 32 على معالج بسرعه 16 ميجا هرتز و ذاكرة كليه تساوى ATmega328



- Arduino : السوفت وير المسؤول عن كيفية فهم الدائره للغه Boot loader
- SRAM: تعتبر الذاكرة المستخدمه فى تسجيل المتغيرات بصوره مؤقتة.
- Flash Disk : مساحه تخزينيه تستخدم فى تخزين البرنامج الذى سنكتبه لتشغيل المتحكمه ، فى الوهله الاولى قد يبدو هذا الرقم صغير جدا لكنه فى الحقيقه كافى لكتابه الكثير من الاوامر كما سرى فى الامثله القادمه.
- EEPROM : الذاكرة المسئوله عن تسجيل بعض المتغيرات بصوره دائمه داخل المتحكمه وتظل محفوظه بقيتها حتى بعد فصل الكهرباء و يمكننا ان نعتبرها مثل السوقه Hard Disk فى الكمبيوتر الشخصي.

## (I/O) Input & Output Pins مداخل و مخارج التحكم

**يمكن** تخصيص الخطوط الرقمية الاربعه عشر (Digital Pins 14) كمداخل أو مخارج (Digital Pins 14) وذلك باستخدام الاوامر البرمجيه كما سنرى في الفصل الثاني و تعمل هذه الخطوط على جهد اقصاه 5 فولت وكل خط يمكن أن يؤمن سحب للتيار بحدود ال 40 ميلي أمبير، وهناك 6 خطوط دخل تماثليه Analog و معنونة من A0 الى A5، بشكل افتراضي تستطيع هذه المداخل قياس جهد من صفر حتى 5 فولت.



## ملاحظات شخصيه :

هذه الصفحة مخصصة لكتابه ملاحظاتك الشخصية عن الفصل الاول :

اكتب ملاحظاتك  
 هنا

# الفصل الثاني: التجهيزات

أولاً: المكونات المادية Hardware & Tools



## كيف تبدأ استخدام اردوينو و ماذا ستحتاج

سوف نستخدم بورده من نوع UNO Arduino وستحتاج للمكونات التالية كبدايه:

(ملحوظه: في بعض الامثله ستحتاج اضافات لهذه المكونات)



### Arduino Uno V.3

هي البورده المسؤوله عن استقبال الأوامر البرمجيه من الحاسب الآلي وكتابتها فى ذاكره المتحكمه الدقيقه **ATmega328** الموضوعه بداخليها



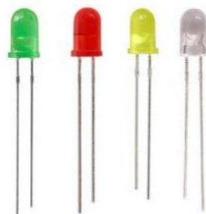
### USB printer cable (1 Meter)

كابل طابعه لتوصيل البورده بالحاسب الآلي، يمكنك شراء اي نوع ويفضل ان يكون قصير قدر الامكان حتى لا يسبب ازعاج اثناء العمل ،طول " واحد متر " سيكون اكثرب من كافي

### لوحة التجارب (Breadboard)



بورده خاصه تستخدلم لتركيب وفك المكونات الالكترونيه عليها بسهوله دون الحاجه الى لحام ، يمكنك شراء اي حجم Half Sized Breadboard ويفضل

**دايدود ضوئي (الوان مختلفة)**

بعض الليدات وهى اشبه بالمسابيح الصغيره و تعمل على تحويل التيار الكهربى الى ضوء، يفضل ان تشتري 10 ليدات بألوان و احجام مختلفه و توفره بألوان مثل:**الاحمر، الأخضر، البرتقالي، الأصفر، الأبيض و الأزرق**

**مقاومات Resistors**

ستحتاج الى مقاومات للتحكم فى التيار الكهربى على مداخل و مخارج المتحكمه الدقيقه و ستحتاج المقاومات التالية:

**Resistor 560 Ohm (x5)**

**Resistor 10 Killo-Ohm (x5)**

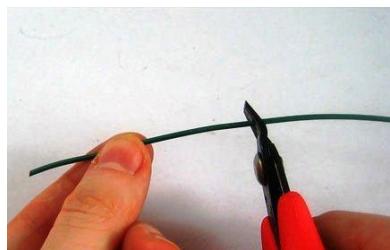
**AVO Meter (multi-meter device)**

**(اختياري)** جهاز قياس كهربائي متعدد الاستخدامات و يستطيع قياس المقاومات و فرق الجهد و شده التيار الكهربى

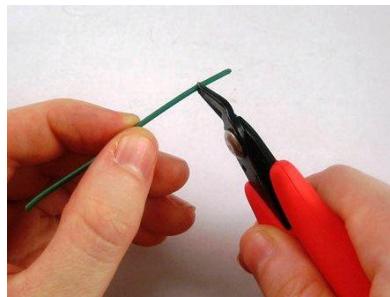
**بعض الاسلاك للتوصيلات بين المكونات Wires**

ستستخدم في توصيل المكونات بعضها على لوحة التجارب و ستتعلم كيف نقوم بتجهيز السلك بنفسك كما سنرى في الصفحة التالية.

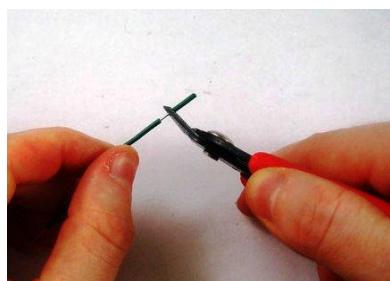
## كيف تُجهز أسلاك التوصيل



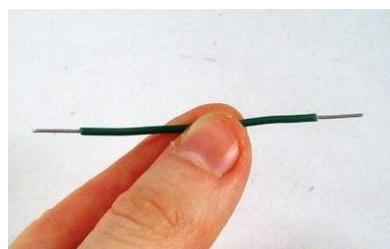
يمكنك الحصول على وصلات عالية الجودة عن طريق استخدام قطعه من كابل Cat5 المستخدم في وصلات شبكات الحاسب الآلي و تستطيع شراءه من محلات مستلزمات الكمبيوتر والشبكات .



قم باستخدام قشاره الاسلاك و قص واحد سنتي متر من الجهتين للسلوك المراد تجهيزه كما في الصورة الاولى، ثم كرر نفس العملية مره اخرى ولكن باطوال مختلفه حتى تحصل على مجموعة اسلاك جاهزة للاستخدام



اذا كنت لا تفضل ان تقوم بهذه العملية يمكنك شراء مجموعة اسلاك مقشره جاهزة للاستخدام مع ال breadboard و ستتجدها في محلات المكونات الالكترونية



**اذا كنت** في مصر و ترغب في شراء تلك المكونات هناك 3

موقع مصرية على الانترنت تستطيع طلب تلك المكونات منها  
و سيتم شحنها لباب منزلك في غضون يومين وسيتم اضافه  
تكلفه الشحن على سعر المكونات.



<http://www.fut-electronics.com>

<http://egyrobots.com>

<http://www.ram-e-shop.com>

اما اذا كنت تفضل زيارة بعض المحلات نفسك



يمكنك الذهاب الى محل رام الكترونيكس RAM Electronics  
الموجود في "باب اللوق" ومحل فيوتشر الكترونيكس Future Electronics  
الموجود امام كلية هندسة عين شمس في ميدان عبده باشا في العباسية .

و اذا كنت في احدى الدول العربية او اي دولة اخرى لا يتوفّر

بها محلاً تبيع هذه المكونات (خاصه **اردوينو**) يمكنك زيارة  
الموقع التالي و ستتجد عليه الكثير من المنتجات الخاصه

**بادروينو** مع الشحن المجاني

<http://www.buyincoins.com>



كما يمكنك زيارة قائمه الموزعين العالميين **اردوينو** لتبث عن موزع محلي قرير منك

عبر الرابط التالي:

<http://arduino.cc/hu/Main/Buy>



## Arduino Uno محتويات عليه

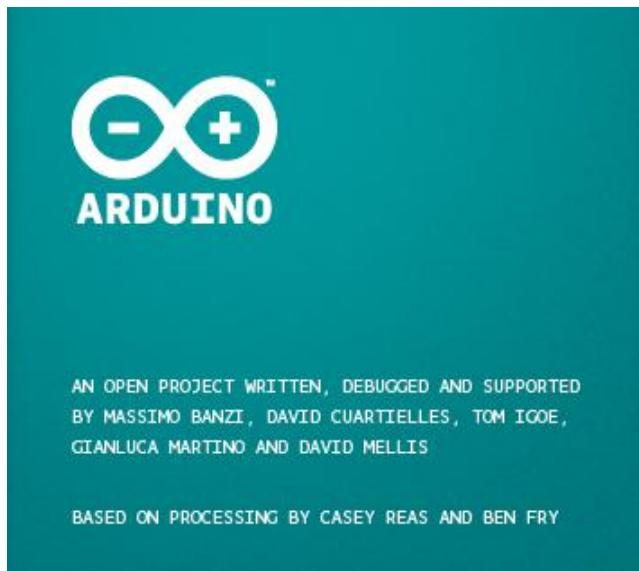
ملحوظه قد تختلف مكونات العلبه من بلد الى اخرى على حسب المورد و رقم  
اصداره البورده كما قد تحتوى على ملصقات اضافيه لاردوينو

## ثانياً: تجهيز البرمجيات



# Software's

## بيئه تطوير اردوينو (IDE)



**تعتبر** بيئه التطوير **Arduino IDE** الاداه المستخدمه فى كتابه الاكواود البرمجيه بلغه **Arduino C** و تحويلها بعد ذلك الى صيغه تنفيذيه يمكن وضعها على المتحكمه الدقيقه الموجوده على البورده .

**تتميز** بيئه تطوير اردوينو بالبساطه و السهوله فى التعامل فهى تقاد تخلو من اي تعقيدات فى المظهر العام و تحتوى فقط على ما يحتاجه المبرمج ليبدأ تطوير برامج بلغه **اردوينو C** كما انها تستخدم فى نفس الوقت لرفع البرنامج مباشره الى المتحكمه الدقيقه و بذلك لن تحتاج الى برنامج اخر مخصص لرفع الصيغه التنفيذيه للبورده .



**سيتم الشرح** على يئه تطوير اردوينو النسخه الخاصه **بويندوز** بأعتباره اكثراً نظمته التشغيل شيوعاً و شهراً بين المستخدمين بالرغم من انى انصح بشده باستخدام لينكس لانه مجاني و سريع و آمن بالإضافة الى انه **مفتوح المصدر**، و فى جميع الاحوال يمكنك العمل على اي نظام تشغيل تحب.

من  **يريد** التعرف اكثراً على نظام تشغيل لينكس هناك كتاب **عربى مجانى [ابونتو ببساطه]** من تأليف: **أحمد محمد أبو زيد** و يشرح استخدام نظام تشغيل **لينكس ابونتو Ubuntu** باسلوب سهل و ممتع و يمكنك تحميله مجاناً من على الموقع الرسمى للكتاب : <http://www.simplyubuntu.com/>



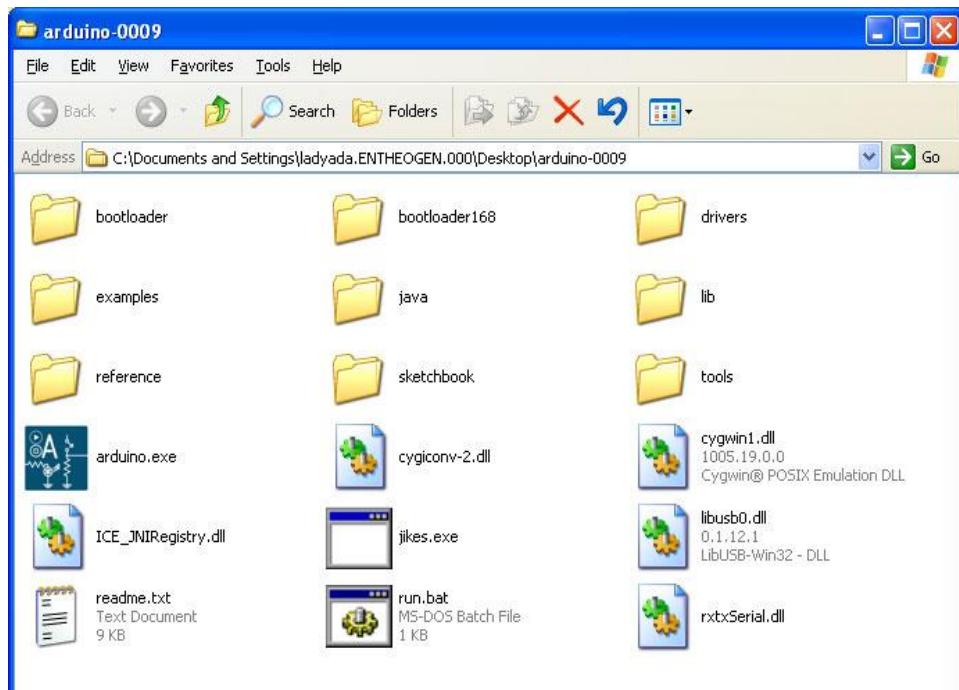
## تنزيل بيئة التطوير Arduino IDE

**توفر** بيئة التطوير البرمجي لجميع أنظمه التشغيل المختلفة Windows, Linux, MAC ويمكنك تحميلها من الموقع الرسمي لمطوري اردوينو على الرابط التالي:

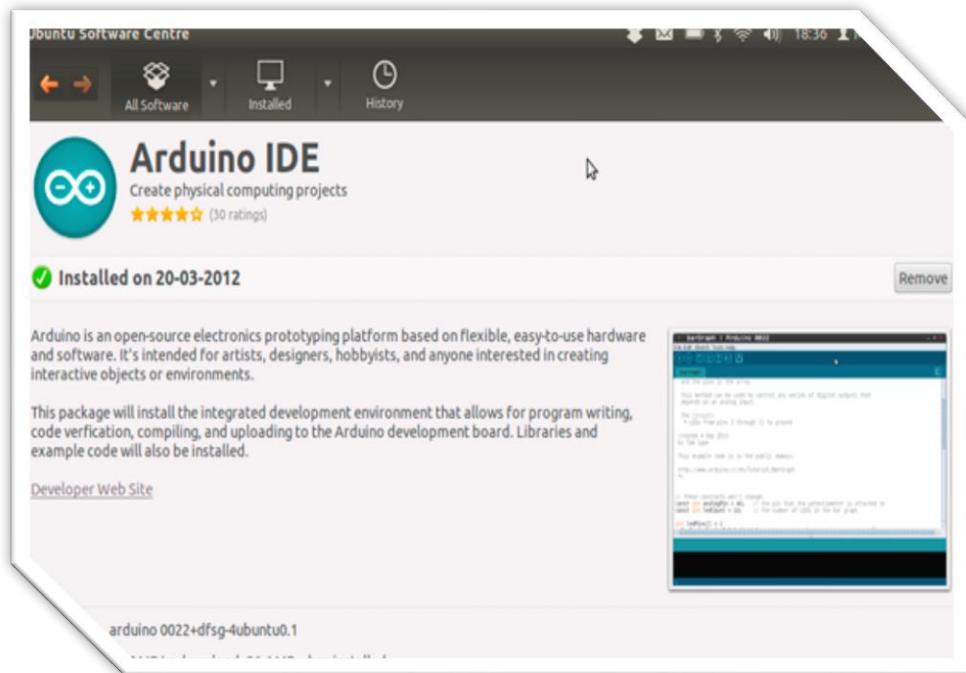
<http://arduino.cc/hu/Main/Software>



**بالنسبة** لمستخدمي ويندوز او ماك قم بتنزيل الاصداره الخاصه بهما من الرابط اعلاه وستجد ملف مضغوط، قم بفك الضغط باستخدام احدى البرامج مثل 7zip ثم قم بتشغيل arduino.exe ملف



**لمستخدمو** نظام لينكس او بنتو Ubuntu يمكنك استخدام مركز البرمجيات فقط ابحث عن Arduino IDE ثم اضغط زر Install في Software center بالنسبة لمستخدمي أنظمه لينكس الأخرى مثل فيدورا fedora و نسخه او بن سوزي ستجدون بيئه التطوير موجوده في مركز البرمجيات الخاص بال DISTRIBUTION Open SUSE ان لم تستطع ايجاده يمكنك تحميل الملف الخاص بلينكس من على الموقع الرسمي.

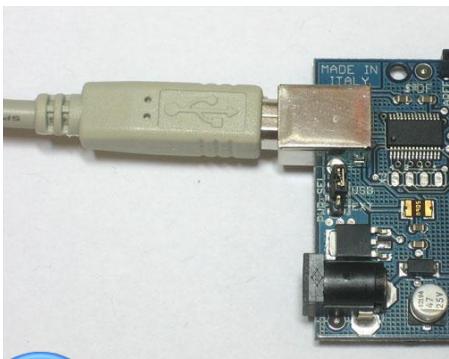


ملحوظه: تعمل البيئه التطويره لاردوينو بصورة اسرع على لينكس اكثر من ويندوز لذلك اناصح اذا كان جهازك المستخدم في عمليه البرمجه ذو امكانيات متوسطه او ضعيفه من ناحيه Hardware يفضل ان تستخدم احدى نسخ لينكس

**بعد الانتهاء من تنصيب Arduino IDE قم بالخطوات التالية لتوصيل البوarde بالكمبيوتر:**



1



2

- قم بتوصيل بورده اردوينو بالكابل
- ادخل الطرف الآخر في فتحة USB في الحاسب الآلي وانتظر قليلا حتى تظهر لك رساله تفيد بأن جهاز الحاسب تعرف على قطعه عتاد جديد Found Arduino Uno Board تلك الرساله يأتي الوقت لبدأ العمل على البيئه التطويريه



3



4

## التعرف على الواجهه الرسوميه لبيئه التطوير



ت تكون بيئه التطوير البرمجيه من واجهه بسيطه و تنقسم الى اربعه اجزاء رئيسيه :

**الاول:** شريط القوائم

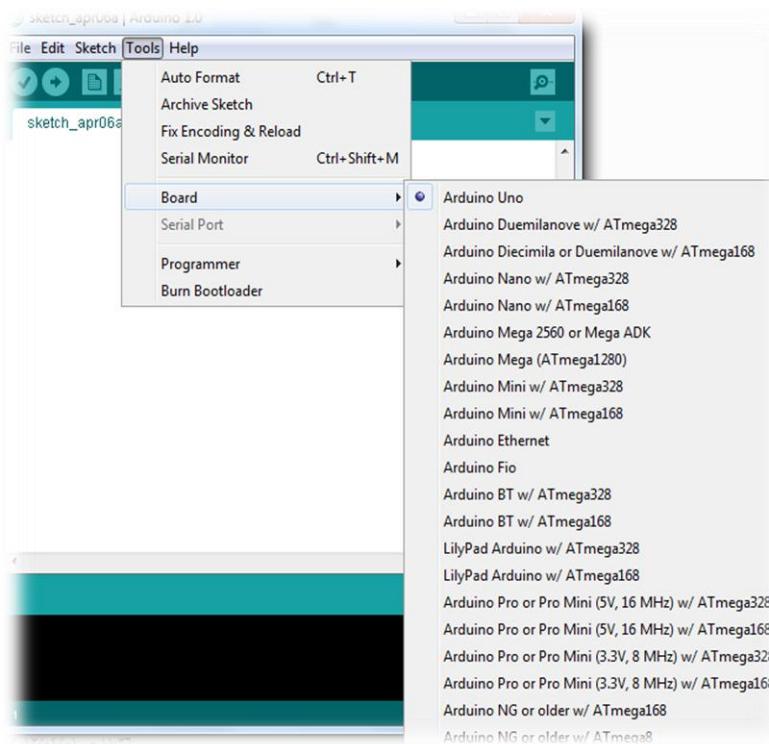
**الثانى:** شريط الاوامر السريعه

**الثالث:** منطقة كتابه الاكواد البرمجيه

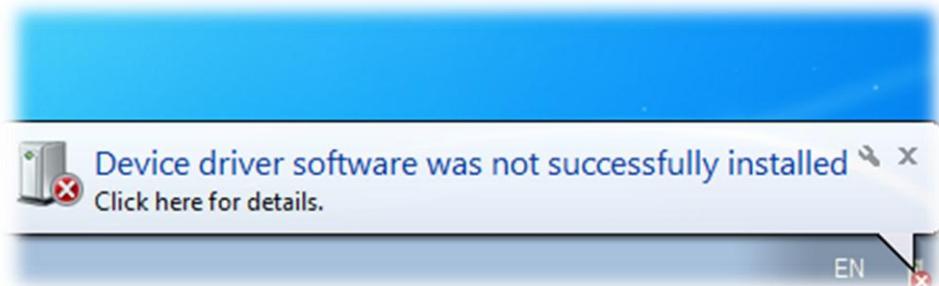
**الرابع:** الجزء الخاص بعرض التنبهات والاخطاء البرمجيه في بيئه التطوير

## تجهيز بيئه التطوير

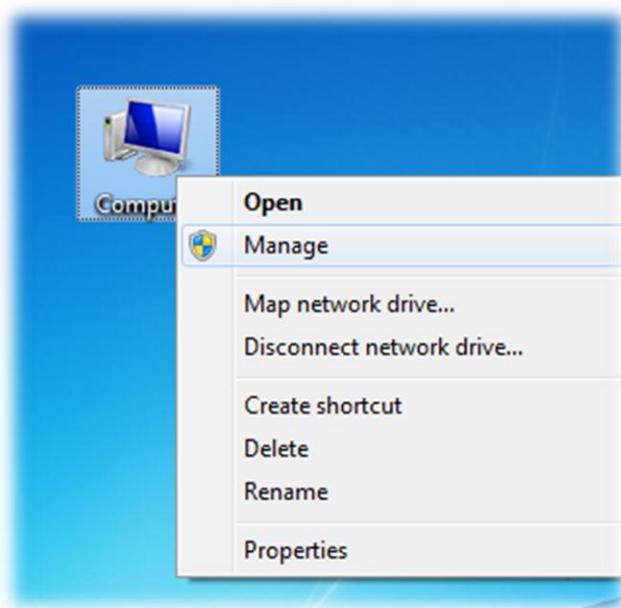
**اول** خطوه لتجهيز البيئه التطوير هى اختيار البورده التى سنتعامل معها و يتم ذلك عن طريق الذهاب الى قائمه Tools واختيار Board ثم نختار نوع البورده المتوفره لدينا و فى هذه الحاله سنختار Arduino Uno كما فى الصوره التاليه

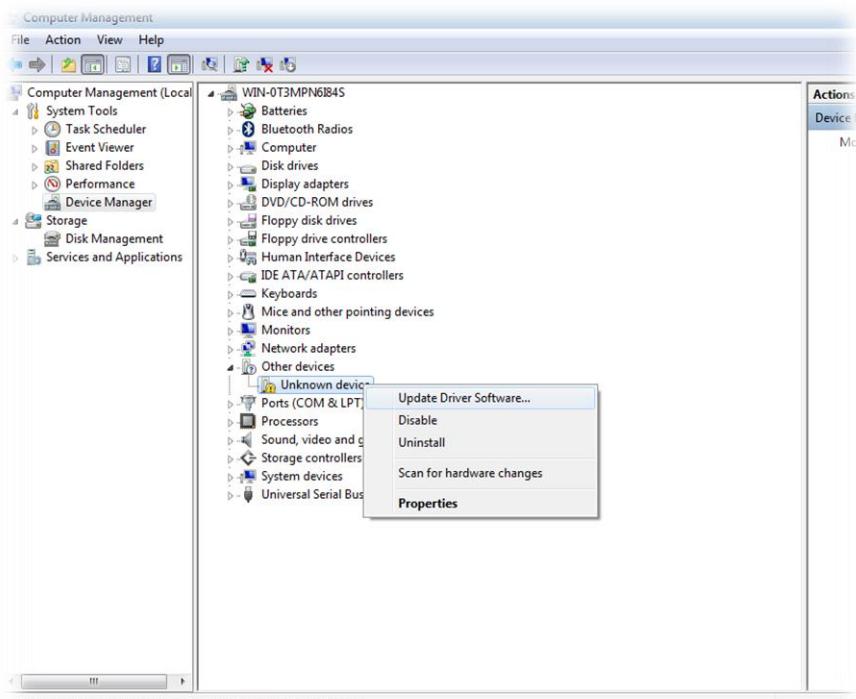


**ثاني** خطوه تتم في حالة ظهور هذه الرساله بعد توصيل اردوينو و التي تفيد بأن الجهاز لم يجد تعريفات لكي يستخدم اردوينو (تظهر غالبا مع مستخدمي ويندوز Vista او Windows 7) ويجب ان نقوم بتنصيب التعريفات كالتالى:



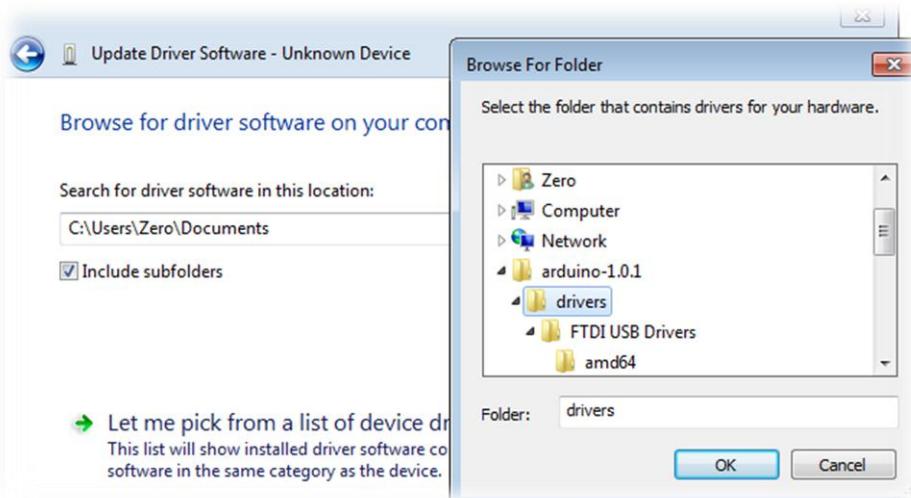
اولا: افتح مدير الاجهزه Device Manager كما في الصور التاليه



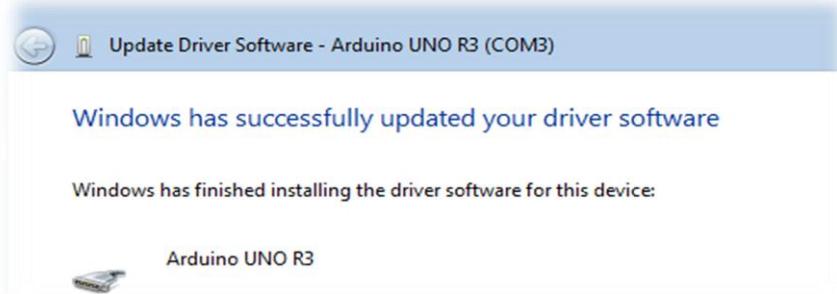


يُفتح المُنافذ لـ "الجهاز لاختيار التعريفات" ثم اختيار الفولدر الذي يوجد به برنامج اردوينو



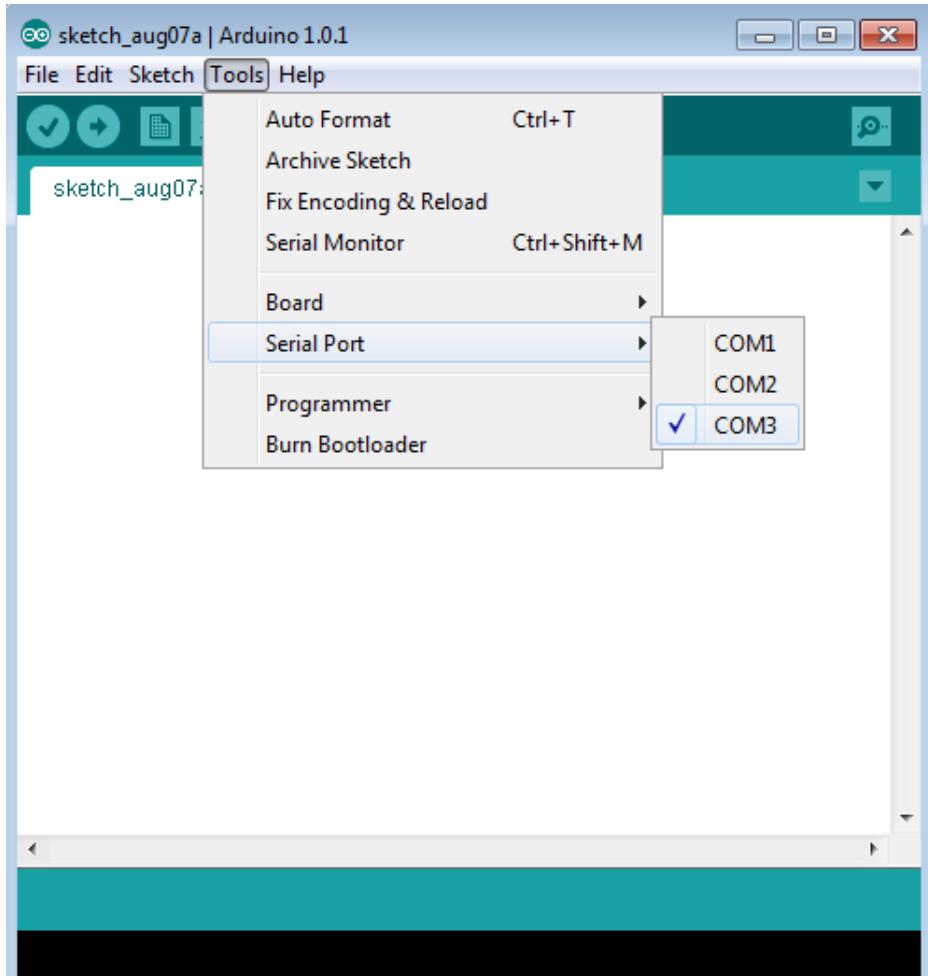


بعدها سوف تظهر رسالة تسؤالك عن "إذا ما كنت ترغب في تنصيب هذه التعريفات ام لا ؟"  
"اضغط Ok وانتظر قليلا حتى تنتهي عملية تعريف اردوينو بنجاح كما في الصورة التالية"



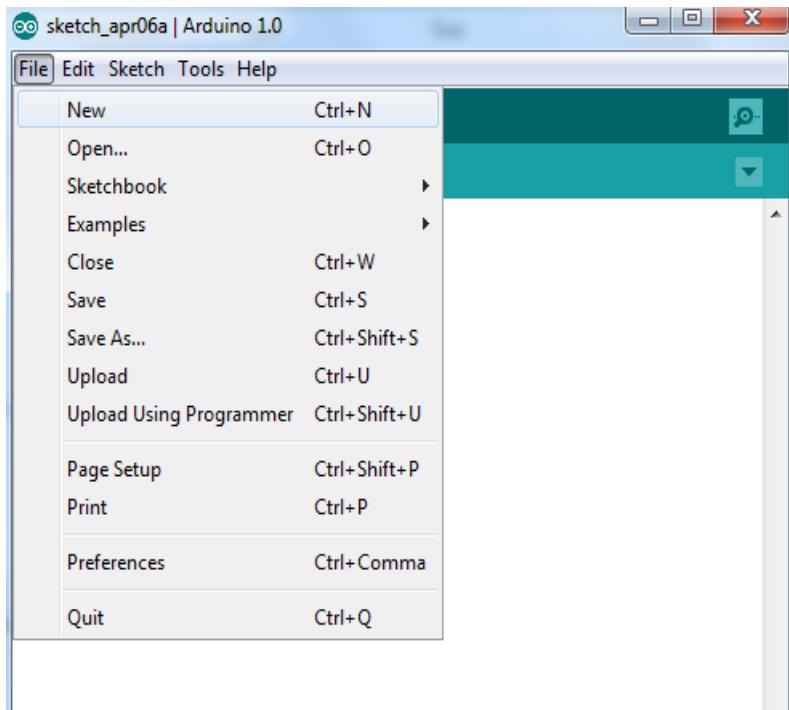
بعد الانتهاء من تنصيب التعريف لاحظ رقم المنفذ المستخدم فى اردوينو من مدير الاجهزه كما في الصورة التالية والتى تظهر رقم المنفذ COM3 ، تذكر ذلك الرقم فسوف تحتاجه لاحقاً ( مع ملاحظة ان ذلك الرقم قد يختلف عندك ) .

الخطوه الاخريه هى اختيار منفذ التوصيل بأردوينو من قائمه



بعد ذلك تستطيع البدأ في كتابة أول برنامج لك على الاردوينو وذلك عن طريق اختيار

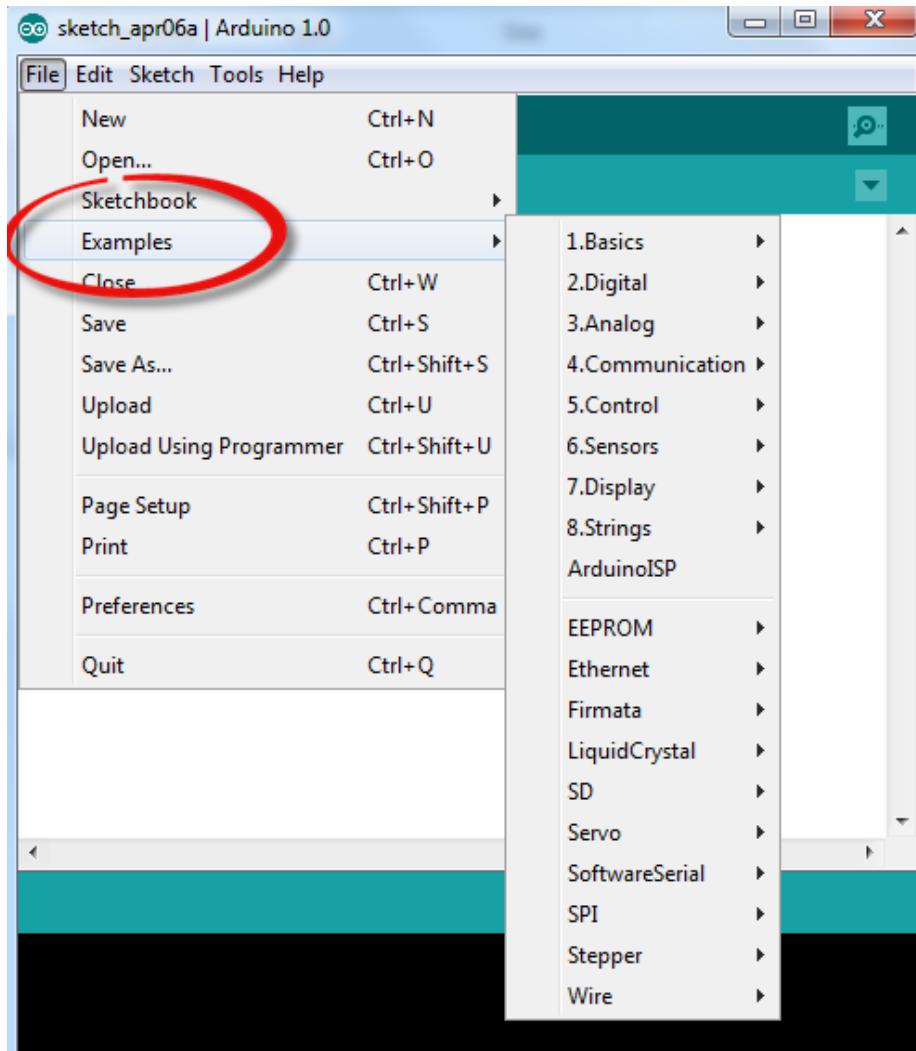
New من قائمه File الموجوده بشرط القوائم كما في الصوره التاليه



او يمكنك اختيار New مباشره من شريط الاوامر السريعه



ايضاً توفر لك البيئه التطويريه مجموعه من الامثله البرمجيه الجاهزه والتى تستطيع الوصول اليها من خلال قائمه Examples الموجوده فى قائمه File الرئيسيه كما في الصوره التاليه



## ملاحظات شخصيه :

هذه الصفحة مخصصة لكتابه ملاحظاتك الشخصية عن الفصل الثاني :

اكتب ملاحظاتك  
 هنا

## مُلْكُ الْحَفْرَالِ الْعَظِيمِ :

كُنْ شَدِيدَ التَّسَامُحَ مَعَ مَنْ خَالَفَ الرَّأْيِ،  
فَإِنْ لَمْ يَكُنْ رَأْيُهُ كُلُّ الصَّوَابِ فَلَا تَكُنْ أَنْتَ  
كُلُّ الْخَطَا بِتَشْبِيْخٍ بِرَأْيِكَ

فرانسوا ماري أرويه المعروف باسم فولتير

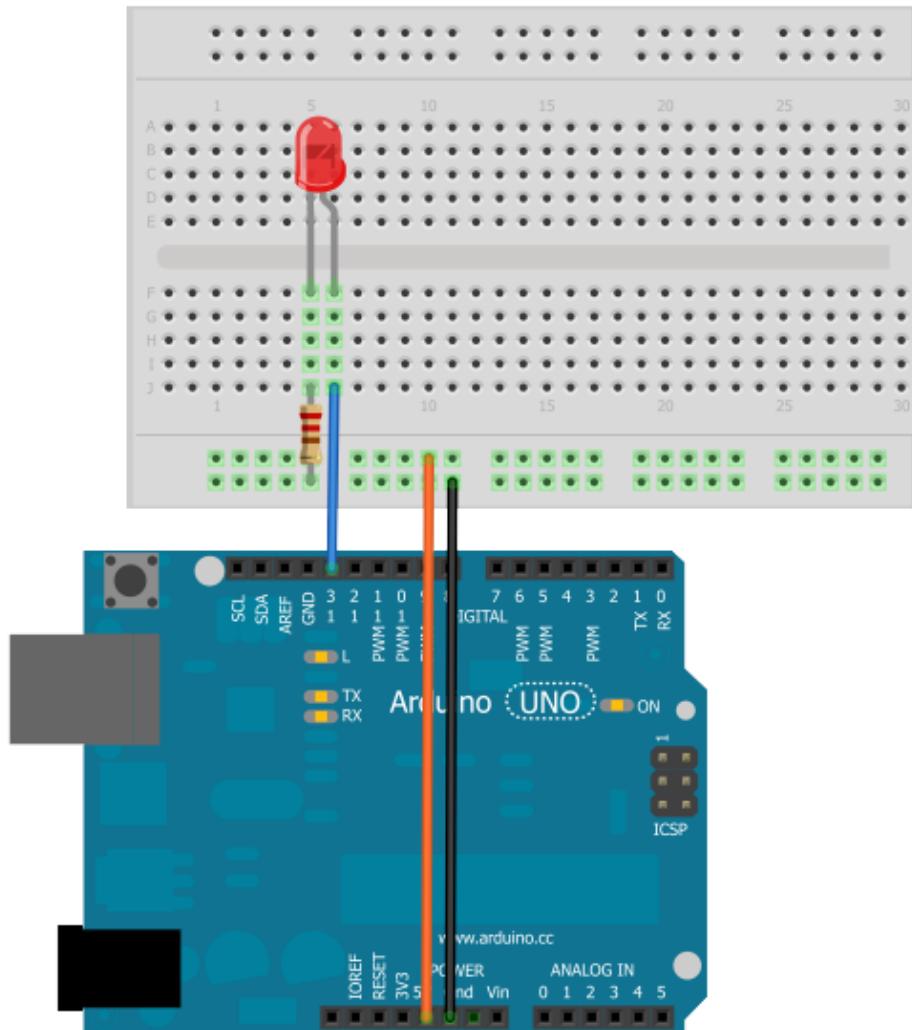
# الفصل الثالث

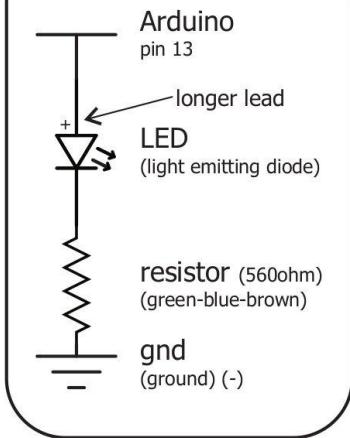
لنبدأ العمل مع اردوينو



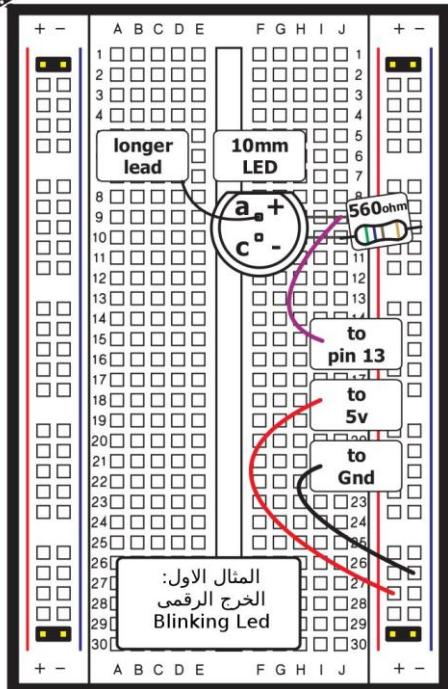
Let's get started

## المثال الأول: تشغيل دايدو ضوئي



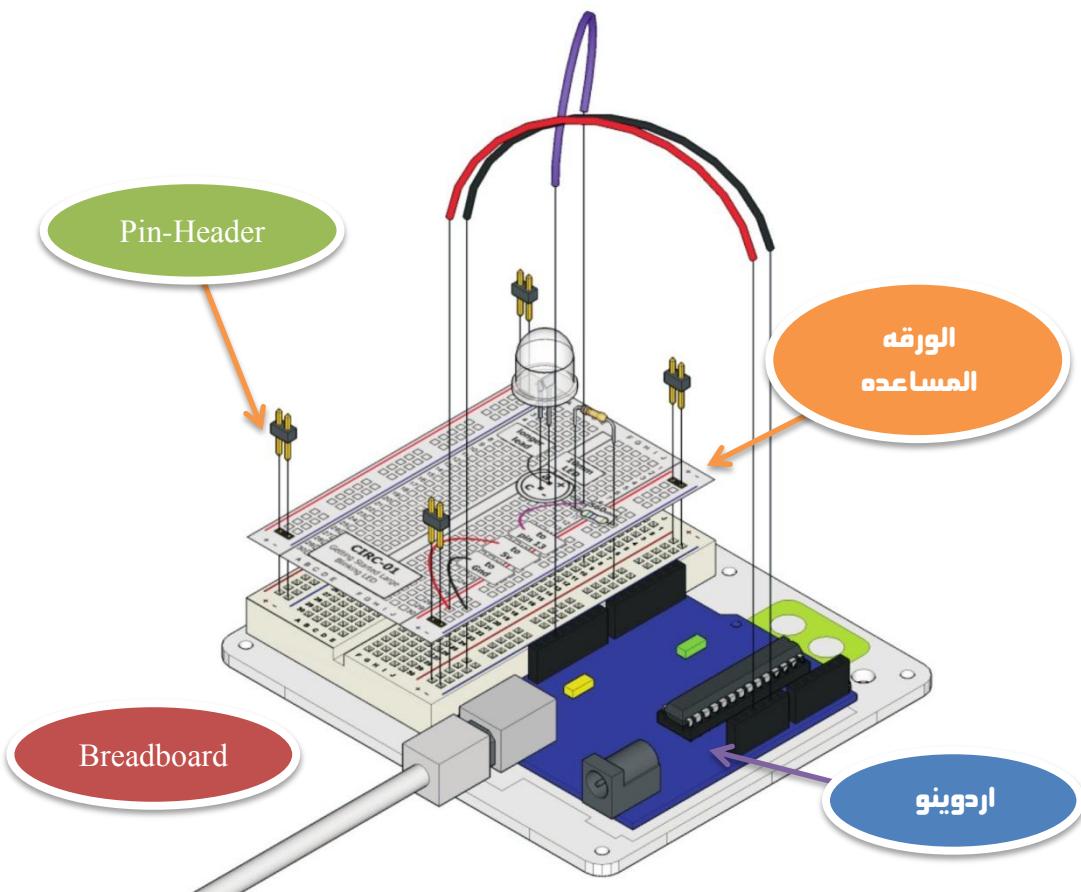
**Schematic****مكونات المثال (١):**

- ✓ بورده اردوينو Arduino Uno
- ✓ Breadboard
- ✓ مقاومه كهرييه بقيمه 560 او姆
- ✓ دايدود ضوئي قطر 10 ملي
- ✓ جهاز الكمبيوتر
- ✓ ورقه المساعده الموجوده في هذه الصفحة

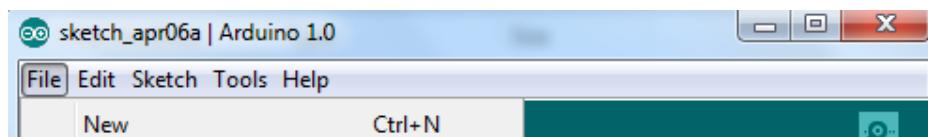


**ما هي الورقه المساعده:** هي ورقه يتم طباعتها لترشدك الى كيفية وضع المكونات الالكترونيه على لوحة التجارب Breadboard

**الخطوه الاولى:** قم بطبعه هذه الصفحة ملحوظه عند الطباعه لا تقوم بتكبير او تصغير حجم الصفحة واستخدم مقياس 100% ثم قص ورقه المساعده بعد ذلك ضع الورقه فوق Breadboard وثبتها باستخدام 4 وصلات pin-header ثم ابدأ فى توصيل المكونات الالكترونيه فوقها كما في الصورة



**الخطوة الثانية:** من بيئه التطوير Arduino IDE قم بعمل ملف جديد من شريط الاوامر السريعه و اكتب الاوامر البرمجيه التالية في الجزء المخصص لكتابه الاكوا德



```
// Example 01 : Blinking LED
```

```
const int LED = 13;
```

```
void setup()
{
    pinMode(LED, OUTPUT);
}
```

```
void loop()
{
    digitalWrite(LED, HIGH);
    delay(1000);

    digitalWrite(LED, LOW);
    delay(1000);
}
```



**الخطوة الثالثة:** بعد ان تنتهي من كتابه الكود البرمجى اضغط على زر تأكيد Verify فى الشريط الاوامر السريعه فى أعلى الـ IDE ثم انتظر حتى تظهر رسالة في الاسفل تخبرك بأن البرنامج قد انتهى من تجهيز الكود ليتم رفعه على **بورده الاردوينو**

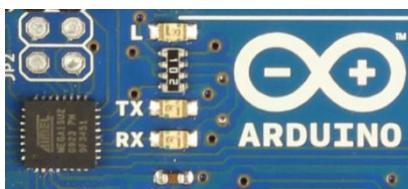
**تأكد** ان كابل الاردوينو متصل

بالحاسب ثم قم بالضغط على زر "رفع"



"Upload" و ذلك حتى يتم بدأ عملية تحميل البرنامج الى المتحكم الدقيقه وانتظر عده ثوانى ستلاحظ عندها ان الـ leds الموجوده على الـ **بورده** (مكتوب بجانبها Tx/Rx) بدأت تضئ بسرعه عاليه ثم تقف قليل و هذه الاضائه تعنى ان الـ **بورده** بدأت تستقبل بيانات البرنامج الذى قمنا بكتابته ، بعد ذلك يبدأ البرنامج و يجعل الليد الذى

وضعناد يضئ و ينطفئ بانتظام كل ثانية .



**مموك** ☺ اول مشروع لك باستخدام اردوينو، والان سنقوم بتشريح و فهم الكود البرمجي المستخدم في هذا المثال.

## شرح الكود البرمجي .....



const int LED = 13;

يقوم هذا الامر بعمل ثابت **constant** قيمته = 13 و اسمه LED



**سؤال طالما كتبنا هذا الثابت ؟؟**



نقوم باستخدام الثوابت حتى تسهل علينا تسمية المخارج والمداخل

(**Input/Output Pins**) للمتحكمه الدقيقه، فمثلاً في هذا البرنامج

قممنا بتعريف هذا الثابت حتى نعبر عن المخرج رقم 13 بأسم LED و

بذلك لو اردنا ان نأمر المتحكمه في اي جزء من اجزاء البرنامج لاحقاً

بأن تشغيل الاليد الموجود على المخرج رقم 13 او أن تقوم بإطفاء ذلك الاليد لنحتاج

عندها لذكر الرقم وإنما سنذكر كلمه **LED** مباشره و ستفهم المتحكمه الدقيقه إننا نقصد

بهذا المخرج رقم 13

```
Void setup ()  
{  
pinMode(LED,OUTPUT);  
}
```

**السطر الثاني من اوامر**

البرنامج هو تعريف وظيفه

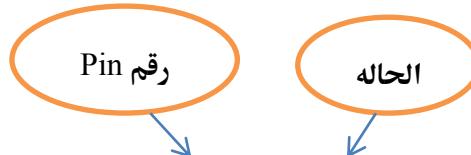
ال pin رقم 13 والى نرمز لها

بأسم **LED**

**فى برامج المتحكمات الدقيقه يتكون البرنامج من 3 مراحل و هى كالتالى:**

**المرحلة الاولى:** ان تقوم بتعريف اي متغيرات او ثوابت فى البرنامج الذى سوف نقوم بكتابته

**المرحلة الثانية:** تميز المتحكمات الدقيقه ان جميع الـ **Digital Pins** يمكن ان تستخدم اما كمخرج او مدخل و فى هذا المرحلة تقوم بضبط اعدادات المخارج و المداخل و تعريف المتحكمه ان الـ Pin رقم 1 او 12 او 13 مثلا سوف تستخدم كمخرج او ان الـ Pin رقم 5 و 6 سوف يستخدمان كمدخل للبيانات و تعتبر الدالة `( )` هي المرحلة الثانية.



**يتم** استخدام الامر `pinMode(pin number, state);` لتحديد وظيفه الـ Pin و هل ستعمل كمدخل ام كمخرج حيث يتم كتابه رقم او اسم الـ Pin اذا قمنا بتعريف ثابت مخصص سابقا، ثم يتم كتابه حالة التشغيل INPUT او OUTPUT ولاحظ انه يجب كتابه حروف كلمه (مدخل او مخرج) بصورة capital.

**ايضا** يجب ان تكتب جميع اعدادات المخارج والمداخل بداخل اقواس الدالة `{ } {write your configurations here}` على سبيل المثال اذا اردنا ان نخبر المتحكمه ان الـ pin العاشره والحاديه عشر والثالثه عشر هم مخارج وان الـ pin الثانيه هي مدخل فستقوم بكتابه الاوامر التالية:

```

void setup()
{
pinMode(11,OUTPUT); ←
pinMode(12,OUTPUT); ←
pinMode(13,OUTPUT); ←
pinMode(2,INPUT); ←
}

```

**لاحظ :** انه يجب بعد ان ننتهي من كل امر يجب ان نضع علامه الفاصله المنقوطة (;) و اذا نسيتها فى اي امر سوف تظهر لك رسالة خطأ عندما تقوم بعمل تأكيد للبرنامج verify

**المرحلة الثالثة:** في هذه المرحلة نقوم بكتابه البرنامج نفسه و ما نريد من المتحكمه ان تتعمل بالضبط و في حالة البرنامج السابق ستقوم المتحكمه بتشغيل الليد لمده ثانية ثم تطفئه لمده ثانية وهكذا مرار و تكرار.

```
void loop()
```

```

{
digitalWrite(LED, HIGH); ← قم بتشغيل LED
delay(1000); ← ثم انتظر 1000 ملي ثانية

digitalWrite(LED, LOW); ← قم بطفاء LED
delay(1000); ← ثم انتظر 1000 ملي ثانية
}

```

**يتم** كتابه برنامج المتحكمه داخل الدالة `void loop () {your program}` ، في هذا المثال استخدمنا الامر: `digitalWrite(LED, HIGH)` وهو المسؤول عن تطبيق جهد كهربى اما  $HIGH = 5$  Volt او  $LOW = 0$  حيث تقوم بكتابه اسم المخرج او لا ثم الجهد الذى نريد تطبيقه و عندما نريد تشغيل الليد نستخدم `HIGH` لتطبيق جهد 5 فولت و عند اطفائه نستخدم `LOW` لتطبيق جهد = صفر.

اما بالنسبة للأمر: `delay (1000)` فهو يستخدم لاخبار المتحكمه كم من الوقت يجب ان تنتظر قبل تنفيذ الامر التالي و بهذا الامر يمكننا ان نتحكم فى كم من الوقت يجب ان يعمل او يطفئ الليد .

و كما شاهدنا في المثال السابق:

```
digitalWrite(LED, HIGH);
delay(1000);
```

و تلك الاوامر تعنى ان المتحكمه ستطبق جهد يساوى 5 فولت `HIGH` على المخرج `LED` ثم تنتظر 1000 ملي ثانى و نلاحظ انه عند كتابه الوقت فى برامج المتحكمات الدقيقه يتم استخدام الملي ثانية `millisecond` والتى تعنى جزء من الف من الثانية و يمكننا استخدام النانو `Nano-second` والميكرو ثانية `Micro-Second`

بعد ذلك تقوم المتحكمه بتطبيق جهد `Low` يساوى صفر فولت والذى يعني ان الليد سوف ينطفئ ثم يظل منطفئ 1000 ملي ثانية اخرى = واحد ثانية .

## ماذا عن اول سطر في المفاصيل؟؟

**يحتاج** المبرمجين عند كتابة اي برنامج مهما كانت اللغة البرمجية الى نظام للتعليقات comments و التعليقات هى جمل تساعد المبرمج على فهم و تنظيم الكود البرمجى و لا تتحسب فى اجزاء البرنامج ، بمعنى اخر سيقوم Arduino IDE بتجاهل التعليقات عن نقل البرنامج الى الصوره التنفيذية و نقله و تشغيله على المتحكمه الدقيقه .

يمكنك كتابة التعليقات فى اردوينو فى اي مكان تشاء و يمكن اعتبار اي جملة انها تعليق ان قمت باضافه "://" قبل بدايه السطر.....مثال:

```
// Hello I'm Abdallah
// I'm Using comments
// Arduino is Sooooo funny
```

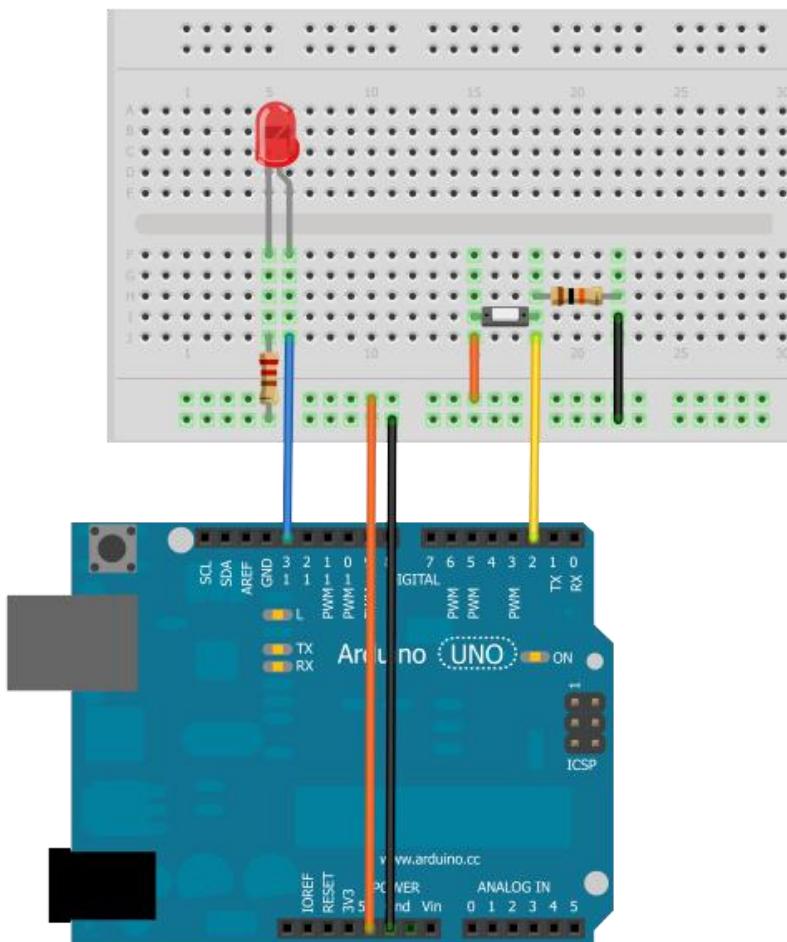
هناك اسلوب اخر لكتابه التعليقات الطويله و هي باستخدام الشرطه المائله و النجمه فى اول الكلام ثم كتابه نجمه و شرطه مائله فى نهايه الكلام .....مثال:

```
/* Hello I'm Abdallah and I Love Arduino, and you
are going to learn The best microcontroller system in
the world (^_^) */
```

**ملحوظه:** فى الطريقه الاولى لكتابه التعليقات يجب ان تكتب // قبل كل سطر تريده ان يتجاوزه بالبرنامج باعتباره تعليق ، لكن فى الطريقه الثانية نكتب \*/ فى بدايه النص ثم نكتب ما نريد من كلام و نصوص و فى النهايه نختتم ب/\* ، لذلك تستخد الم طريقه الثانية فى التعليقات الطويله و الطريقه الاولى فى التعليقات القصيره التي غالبا ما تكون سطر واحد.

## المثال الثاني: تطوير للمثال السابق ليعمل فقط عند ضغط مفتاح (سويتش) من نوع Push button

في هذا المثال سنقوم بتطوير الفكرة السابقة و سنجعل اليد يعمل في حالة اذا ضغط المستخدم على زر من نوع push button وهو اشهر انواع السويتشات المستخدمة في الأجهزة الإلكترونية.



## مكونات المثال (2):

- بورده اردوينو Arduino uno
- Breadboard
- Push button
- Led 10 mm
- مقاومه 10 كيلو اوم
- مقاومه 560 اوم
- بعض السلاوك للتوصيل

قم بوضع السويتش على لوحة التجارب ثم وصل احدى طرفيه بالمدخل 5v باستخدام سلك على بورده اردوينو والطرف الاخر بالمقاومة العشره كيلو اوم ومن نفس النقطه قم بتوصيل سلك للمدخل رقم 2 على اردوينو بورد و بعد ذلك قم بتوصيل طرف المقاومه الثاني بالمدخل الارضي .

### ما الهدف من هذه الخطوه ؟؟

تعمل تلك الدائمه على توفير دخل رقمي Digital input ، عند ضغط الزر تنطلق نبضه بقيمه 5 فولت والتى تعتبرها المتحكمه الدقيقه اشاره من نوع HIGH و عند ترك الزر ينفصل التيار الكهربائي ويصبح الدخل يساوى صفر فولت و الذى تعتبره المتحكمه LOW و سوف نستغل هذا الامر لتعديل المثال الاول لكي يعمل فقط عندما يكون الدخل HIGH

**الخطوه الثانية :** كتابه الأكواد البرمجية التالية في

نصيحة: قم بكتابه الأوامر البرمجيه بنفسك ولا تأخذها نسخ و لصق من الكتاب وذلك حتى تدرب نفسك على كتابه الكود

```
// Example_2_Blinking_with_Switch
```

```
const int ledPin = 13;
const int buttonPin = 2;
int val;
```

```
void setup ()
```

```
{
pinMode(ledPin, OUTPUT);
pinMode(buttonPin, INPUT);
```

```
}
```

```
void loop()
```

```
{
val = digitalRead(buttonPin);
if (val == HIGH)
```

```

{
digitalWrite(ledPin, HIGH);
delay(1000);
digitalWrite(ledPin, LOW);
delay(1000);
}
```

```
else {digitalWrite(ledPin, LOW); } }
```

**المرحلة الاولى: تعريف المتغيرات و الثوابت**

**المرحلة الثانية: تعريف و تحديد المدخلات و المخارج**

**المرحلة الثالثه: كتابه اوامر البرنامج الرئيسيه**

**الخطوة الثالثة:** قم بعمل تأكيد للكود البرمجي عن طريق الضغط على علامه تأكيد Verify وبعد ان تتأكد من ان البرنامج صحيح ولا يحتوى على اخطاء برمجيه وتم تحويله الى الصوره التنفيذية بنجاح، قم بعد ذلك برفعه للمتحكمه الدقيقه عن طريق زر Upload

## شرح الكود البرمجي .....



**المرحلة الاولى:** قمنا بتعريف متغير اسمه ledPin والذى سيرمز الى Pin 13 و كذلك قمنا بتعريف المتغير المتغير buttonPin والذى سيرمز الى السويفش الموجود

```
int ledPin = 13;
int buttonPin = 2;
int val = 0;
```

على المدخل الرقمي السابع و قمنا بتعريف المتغير val والذى سوف نستخدمه لاحقا لتخزين حالة الدخل على المدخل الرقمي الثاني 2 pin

**المرحلة الثانية:** فى هذه المرحلة جعلنا المتحكمه تعامل pin 13 والتي يرمز لها باسما ledPin على أنها مخرج و سنستخدمه لتشغيل الليد الموصل به، ثم قمنا بجعل المتحكمه تعامل المخرج pin 2 على انه مدخل رقمي وظيفته استقبال الاشارات

الرقميه  
HIGH or LOW

```
void setup()
{
pinMode(ledPin, OUTPUT);
pinMode(inPin, INPUT);
}
```

## المرحلة الثالثة:

```
val = digitalRead(buttonPin);
```

في هذا الأمر ستقوم المتحكمه بقياس قيمه الجهد على المدخل 2 pin و تسجل القيمه في المتغير val

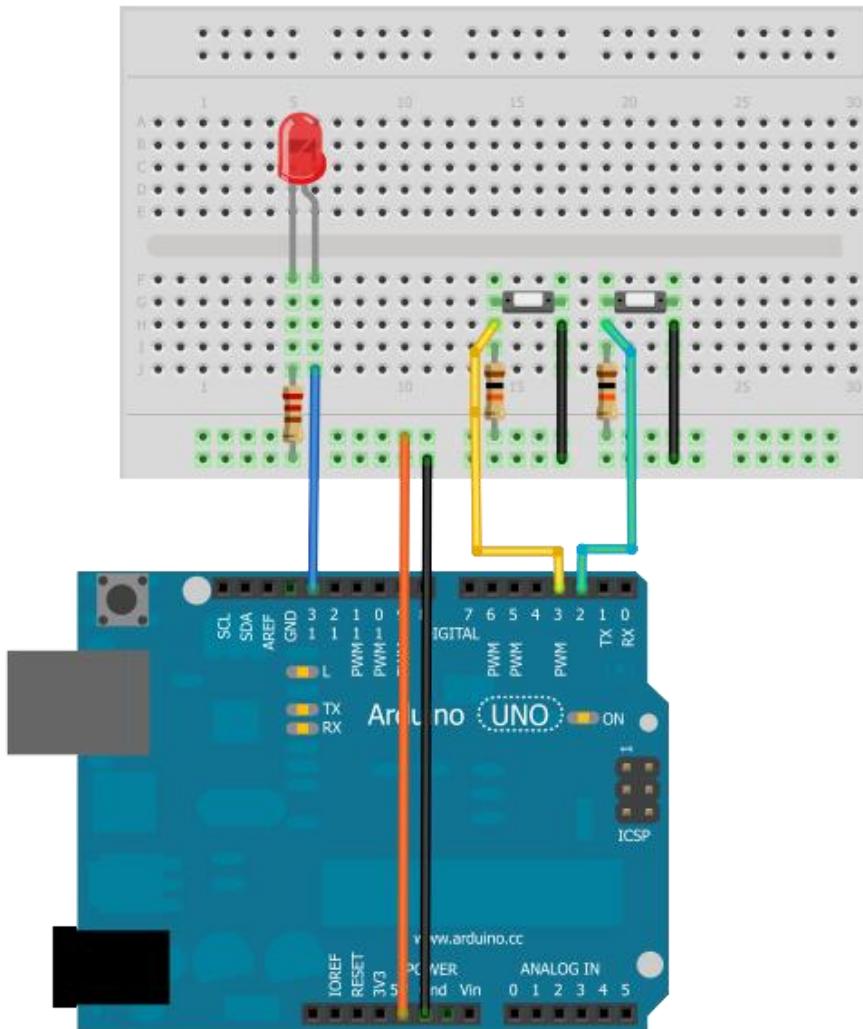
ويتم ذلك باستخدام الأمر `digitalRead()` اذا كان المستخدم ضغط على السويتش  
فأن القيمه ستكون 5 فولت = HIGH و ان لم يضغط ستكون القيمه صفر فولت = LOW

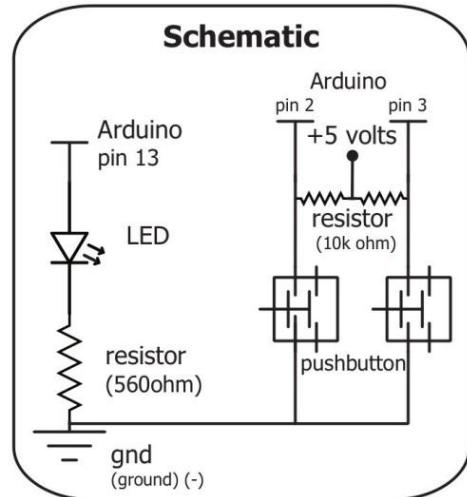
```
if (val == HIGH)
{
    digitalWrite(ledPin, HIGH);
    delay(1000);
    digitalWrite(ledPin, LOW);
    delay(1000);
}
else {digitalWrite(ledPin, LOW); }
```

في هذا الجزء استخدمنا واحد من اشهر الاوامر في لغات البرمجه وهو الامر `( if....else )` يستخدم ذلك الامر لعمل مقارنات و جعل المتحكمه تفعل شيء ما اذا حدث شيء اخر محدد وفي هذا المثال استخدمنا الامر if كالالتالي :

- ستقيس المتحكمه قيمه الجهد على المدخل السابع و تخزن القيمه في المتغير val
- اذا كانت القيمه تساوى HIGH و التي تعنى 5 فولت ستقوم المتحكمه بتشغيل الليد لمده ثانية و اطفائه لمده ثانية و هكذا باستمرا.
- ما لم تكن القيمه تساوى HIGH و الجهد المطبق على المدخل يساوى صفر ستقوم المتحكمه بتطبيق جهد يساوى صفر على الليد و بذلك لن يعمل الليد وسيظل منطفي.

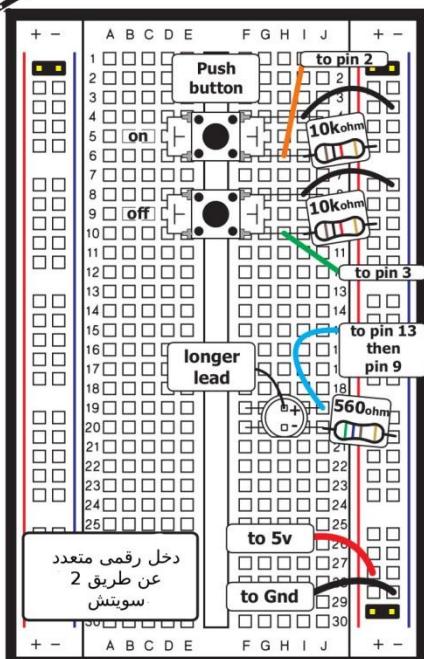
## المثال الثالث: تطوير المثال السابق يعمل بـ 2 سويتش من نوع Push Button واحد من أجل التشغيل ON والثاني من أجل الغلاق OFF



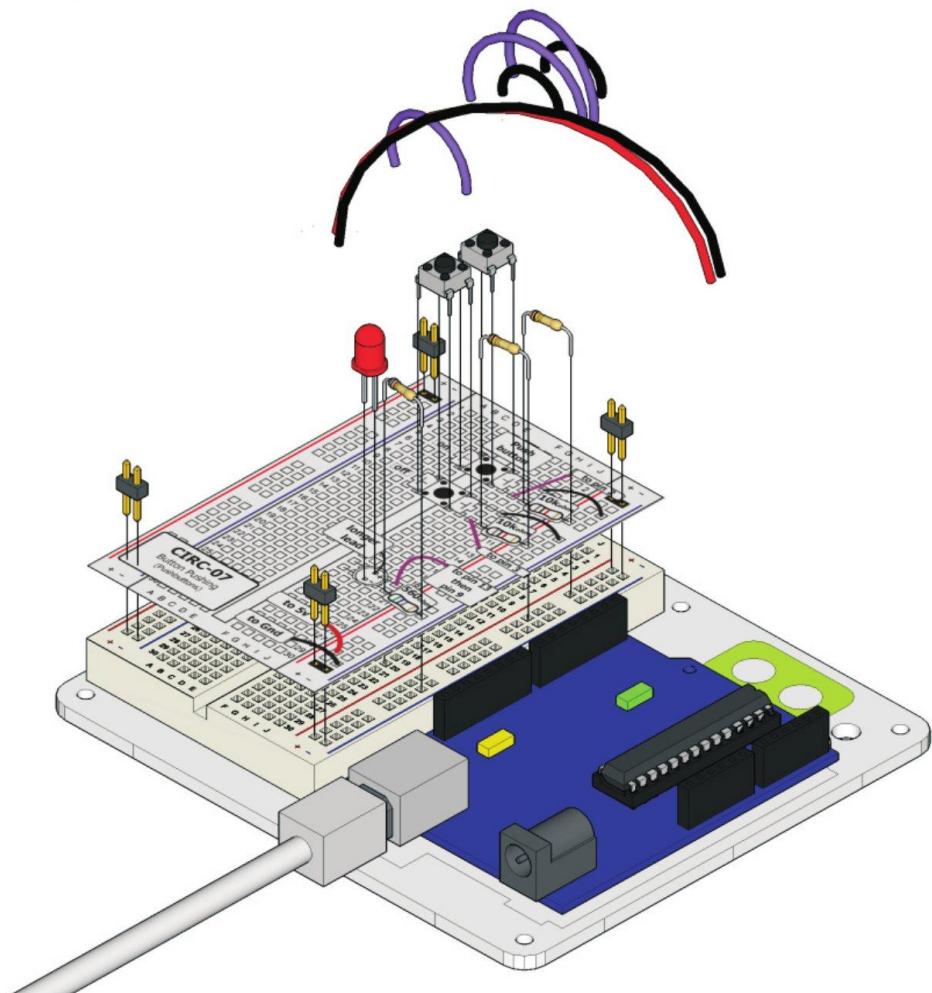
**مكونات المثال(3):**

- Arduino uno
- Breadboard
- Led 5 mm
- عدد 2 مقاومه قيمه 10 كيلو اوم
- مقاومه واحده قيمه 560 اوم
- عدد 2 سويتش
- اسلاك توصيل
- نسخه مطبوعه من الورقه المساعده

فى هذا المثال سندمج ما تعلمناه فى كلا المثالين الاول والثانى ، سنسخدم 2 سويتش من نوع **push button** وسيتولى السويتش الاول عملية تشغيل الليد و سيتولى السويتش الاخر عملية اطفائه.



**الخطوه الأولى :** قم بطبعه ورقه المساعده وقصها ثم ثبتها على breadboard باستخدام 4 من Pin-Headers ثم قم بتوصيل المكونات الالكترونيه على الورقه المساعده كما في الصوره



**الخطوة الثانية:** من بيئه التطوير Arduino IDE قم بعمل ملف جديد من شريط الاوامر السريعه و اكتب الاوامر البرمجيه التالية في الجزء المخصص لكتابه الاكواد و بعد الانتهاء ارفع البرنامج الى المتحكمه الدقيقه على البورده و جرب الضغط على السويفتش الاول و الثاني و شاهد ماذا سيحدث

```
//Example_3_On_Off_with_2_Switches_1
```

```
const int ledPin= 13;
```

```
const int inputPin1 = 2;
```

```
const int inputPin2 = 3;
```

```
void setup ()
```

```
{
```

```
pinMode(ledPin, OUTPUT);
```

```
pinMode(inputPin1, INPUT);
```

```
pinMode(inputPin2, INPUT);
```

```
}
```

```
void loop ()
```

```
{
```

```
if (digitalRead(inputPin1) == HIGH)
```

```
{digitalWrite(ledPin, LOW); }
```

```
else if (digitalRead(inputPin2) == HIGH)
```

```
{digitalWrite(ledPin, HIGH); }
```

```
}
```

تحديد المدخل  
والمخرج  
للدائرة

اوامر البرنامج  
الرئيسية

**في هذا المثال** استخدمنا تعديل بسيط على الأمر if و هو أضافه الأمر else if بدلًا من else فقط و هو أمر يمكننا من أضافه اكثر من احتمال واحد في جمله if .

مثال:

If (you like science) {you should read books}

else if (you like sport) {you should play football}

else if (you like art) { you should take art lessons}

else { you should search for a hobby to do in your summer}

كما ترى في هذا المثال وضعنا 3 احتمالات و خيار آخر كتالى :

- **إذا** كنت تحب العلوم عليك أن تقرأ بعض الكتب
- **أما إذا** كنت تحب الرياضة عليك أن تلعب كره القدم
- **أما إذا** كنت تحب الفن عليك أن تتعلم دروس الرسم
- **و إذا لم** يكن لديك هواية عليك أن تبحث عن هواية لتشغل بها وقت الإجازة الصيفية

وفي البرنامج السابق استخدمنا احتمال أن المستخدم سوف يضغط على السويتش الأول فنقوم المحكمه الدقيقه بتشغيل الدايمود الضوئي والاحتمال الثاني هو أن يضغط المستخدم السويتش 2 فنقوم المحكمه بإطفاء الدايمود الضوئي ... وهكذا

**و الأن ما رأيك أن نعدل المفهوم قليل (^\_^)**

```
//Example_3_On_Off_with_2_Switchs_2
const int ledPin= 13;
const int inputPin1 = 2;
const int inputPin2 = 3;

void setup ()
{
pinMode(ledPin, OUTPUT);
pinMode(inputPin1, INPUT);
pinMode(inputPin2, INPUT);
}

void loop ()
{

if (digitalRead(inputPin1) == HIGH)
{
digitalWrite(ledPin, HIGH);
delay(400);
digitalWrite(ledPin, LOW);
delay(400);
}

else if (digitalRead(inputPin2) == HIGH)
{
digitalWrite(ledPin, HIGH);
delay(100);
digitalWrite(ledPin, LOW);
delay(100);
}

}
```

## مراجعة سريعة على ما تعلمناه في الامثله السابقه

<code>void setup ( )</code> {.....}	الدالة المسئولة عن ضبط اعدادات المتحكم و تحديد المدخل والخارج
<code>void loop ( )</code> {.....}	يتم كتابه البرنامج الرئيسي داخل هذه الدالة و يقوم البرنامج باعاده نفسه مره تلو الاخرى كلما اتم تنفيذ جميع الخطوات
<code>int name = value;</code>	امر يستخدم لتعريف متغير بقيمه معينه
<code>cons int name = value;</code>	امر يستخدم لتعريف ثابت غير قابل للتغيير يشبه الامر السابق مع اضافه const
<u>example:</u> <code>const int led = 13;</code>	
<code>pinMode (pin number, State);</code>	الامر المسئول عن تحديد وظيفه كل pin و ضبطها للعمل كمدخل او كمخرج INPUT or OUTPUT
<u>example1:</u> <code>pinMode(11,INPUT);</code> <u>example2:</u> <code>pinMode(led,OUTPUT);</code>	
<code>digitalWrite (pin number, State);</code>	الامر المسئول عن اخراج جهد على احدى pins بصورة رقميه (1) HIGH or LOW(0)
<u>example:</u> <code>digitalWrite(13,HIGH);</code>	
<code>digitalRead (pin number);</code>	الامر المسئول عن قراءه الجهد المطبق على احدى pins بصورة رقميه حيث يقوم بقياس الجهد اذا كان 5 فولت حتى 3.8 فولت يعتبر HIGH و اذا كان
<u>example:</u> <code>digitalRead(4);</code>	

	ما بين صفر حتى 1.8 فولت يعتبر ( LOW )
<b>delay (time);</b>  <u>example1:</u> <code>delay(100);</code> <u>example2:</u> <code>delay(1000);</code>	تأخير بزمن محدد : يستعمل هذا الامر في تحديد المدة الزمنية للامر الذي يسبقه في التنفيذ
<b>if (the condition) {.....what to do}</b>  <b>else if (another condition) {...what to do}</b>  <b>else (last condition) {....what to do}</b>	اذا حدث ... او اذا حدث.. افعل ..... جمله شوطيه تستخدم لتنفيذ امر ما عندما يتحقق شرط معين و يمكن استخدام الشروط التالية:  == يساوي != لا يساوي <= أقل من او يساوي >= أكبر من او يساوي < أقل من > أكبر من

أشهر أنواع المتغيرات التي يمكن استخدامها في اردوينو كال التالي:

النوع	مثال	القيمة
Integers	int led = 13;	أي عدد صحيح من سالب 32768 حتى موجب 32768
Float	float sensor = 12.5;	يستخدم مع الأعداد التي تحتوي على كسور
character	char name= hello;	حروف نصية أو كلمات
long	long variable = 99999999;	رقم طويل جدا يتراوح بين سالب 2,147,483,648 حتى موجب نفس الرقم
byte	byte number= 44;	رقم صغير يتراوح بين (0) إلى 255

لمزيد من المعلومات عن أنواع المتغيرات التي يمكن استخدامها داخل برمج اردوينو قم بزيار الصفحة التالية

<http://www.arduino.cc/en/Reference/VariableDeclaration>

## ملاحظات شخصيه :

هذه الصفحة مخصصة لكتابه ملاحظاتك الشخصيه عن الفصل الثالث :

اكتب ملاحظاتك  
 هنا

## مُلْكُ الْقُوَّالِ الْعَظِيمِ :

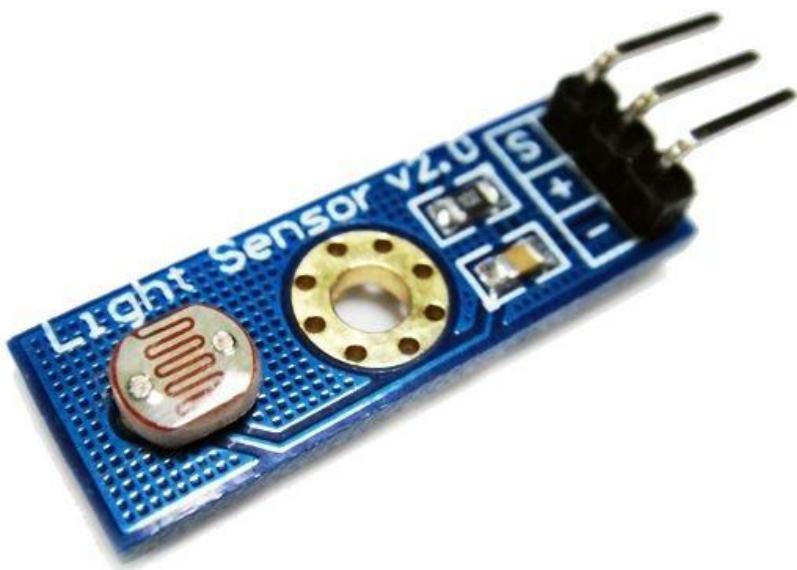
ذو الْعَقْلِ يَشْقَى فِي التَّعْيِمِ بَعْقَلِهِ  
وَأَخُو الْجَهَالَةِ فِي الشَّقَاوَةِ يَنْعَمُ

أبو الطيب المتنبي

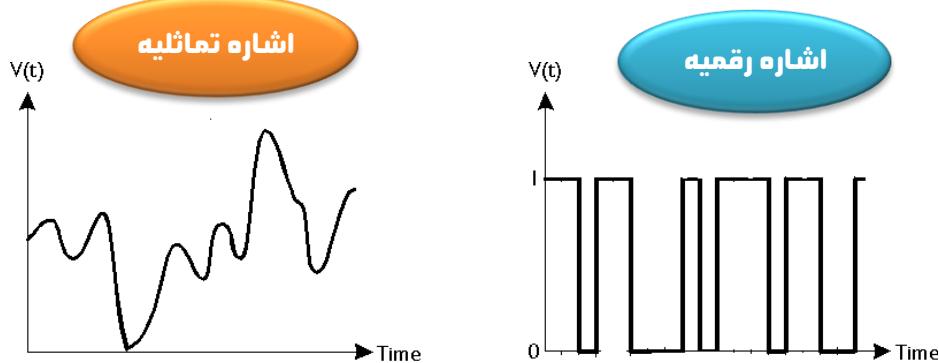
# الفصل الرابع

الدخل والخرج التماضي واستخدام المستشعرات

Analog Inputs/Outputs and Sensors



**يعرف** الدخل او الخرج التماثلى بأنه اي فرق جهد تبدأ قيمته من صفر وقابل للتغير دائمًا و ليس له حدود قصوى معينه ، على عكس الدخل الرقمى والذى اما يكون 5 فولت HIGH (1) واما يكون صفر فولت LOW (0)



## ما هي أهمية الاشارات التماثليه ؟؟ Analog Signals

يعبر الدخل التماثلى عن الاشارات المتغيرة مثل فرق الجهد الناتج من مستشعر (حساس sensor) معين نتيجة التغير في عامل معين، على سبيل المثال : حساس للضوء او ما يعرف باسم المقاومه الضوئيه Photo-Resistor و هى مقاومه كهربائيه تتغير قيمتها بتغير مقدار الضوء الذى تتعرض له و اذا قمنا بتوصيل هذا المقاومه على مصدر ثابت للفولت و استخدامنا المتر AVO-meter لقياس فرق الجهد المطبق عليها .. سنجد ان فرق الجهد سيتغير دائمًا بتغير مقدار الضوء الساقط عليها



**و يمكننا ان** نستغل تلك الظاهرة فى قياس اي عامل بيئى معين باستخدام مستشعرات مناسبه والتى تقيس ذلك العامل البيئى و تحوله الى اشاره كهربائيه تماثليه قابله للقياس مثل: الحرارة، الضوء، الرطوبة، سرعه الرياح، القوه، العزم، التسارع، المجال المغناطيسي... الخ



تحتوى شريحة-p 328 Atmega المستخدمه فى بورده Arduino Uno على 6 مداخل تستطيع قياس مقدار فرق الجهد الكهربائي المتغير (دخل تماثلي) وهي مرقمه من A0 حتى A5 و يمكنها قياس فرق جهد بدقة تصل الى 4.8 مللى فولت و هذا يعني تقريبا خمسه اجزاء من الالف جزء من واحد فولت و هي دقه عاليه تكفى للكثير من التطبيقات الدقيقه

**في هذا الفصل** سنتعرف على بعض اشهر المستشعرات مثل المقاومه الضوئيه و مستشعر الحرارة مثل LM35 أو TMP36 هو عبارة عن ترانزistor خاص يتاثر بالحرار و يعطى تغير في الجهد الكهربى (الفولت) متناسب مع التغير في درجة الحرارة .. حيث يعطي 10 مللى فولت (عشره اجزاء من الف فولت) لكل 0.1 درجة مئوية.... كما سنتعرف على الدخل والخرج التماثلي وكيفيه استخدام المستشعرات مع اردوينو

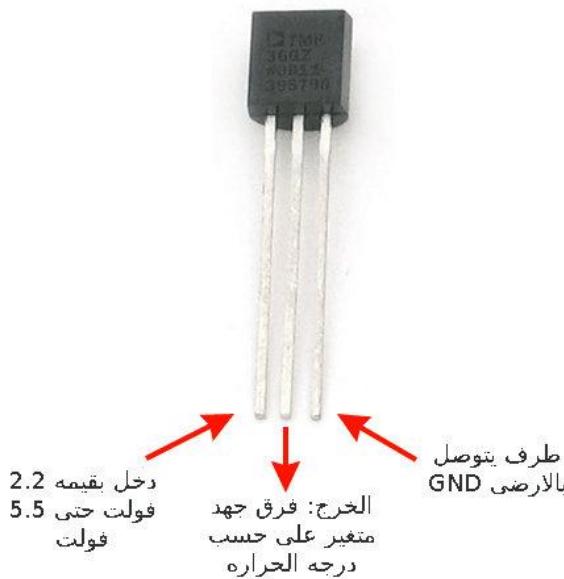


## كيف تولد الحساسات الإشارة التماثلية ؟؟

لنأخذ حساس الحرارة كمثال: حساس الحرارة مثل TMP36, TMP35, LM35 ما هو الا ترانزستور ذو حساسية عالية للتغير في الحرارة و من المعروف أن الترانزستور يصنع من مادة السيلكون التي تتغير مدي قابليتها لتوصيل الكهرباء بتغير الحرارة حيث يزداد التيار الكهربائي المار فيها بزيادة الحرارة و يمكننا إجراء تجربة بسيطة لنتعرف أكثر على طريقه عمل ذلك الحساس.

يحتوى الحساس على ثلاثة أرجل وهى كالتالى:

- 1- منفذ الدخل و يتم توصيل جهد ثابت بين 2.2 فولت حتى 5.5 فولت
- 2- منفذ الخرج وهو المنفذ الذى نحصل منه على قراءة الحساس
- 3- منفذ الأرضي و يتم توصيله بأى نقطه ارضى GND



سنحتاج لهذه التجربة المكونات التالية:



-1 عدد 2 بطاريه مقاس AAA بقيمه 1.5 فولت

-2 عدد 1 حامل بطاريات لتوصيل 2 بطاريه

-3 جهاز قياس متعدد الاستخدامات AVO  
Meter

-4 حساس حراري من أي الأنواع التالية:  
TMP 35, TMP36, LM 35

احضر 2 بطاريه بقياس AAA بقيمه 1.5 فولت وضعهم في  
علبه حامل البطاريات لتحصل على قيمه كلية تساوى 3 فولت  
ثم وصل الطرف الموجب لحامل البطارية (سلك لونه احمر)



بالطرف الأول من ناحيه اليسار للحساس الحراري (طرف الدخل الثابت)

ثم قم بتوصيل الطرف الأرضي لحامل البطارية (سلك لونه اسود) بالطرف الأخير  
للحساس الحراري ، بعد ذلك قم بضبط الـ AVO على وضع قياس الفولت ووصل  
الطرف الأحمر بالمخرج للحساس الكهربائي و الطرف الأسود للطرف الأرضي للحساس

الكهربائي



لاحظ قراءة فرق الجهد الظاهر على الـ AVO meter ستجد أنها 0.76 فولت تقريباً  
الآن قم بوضع يدك على الحساس و اضغط عليه قليلاً (تعمل تلك الحركة على رفع درجة حرارة الحساس و من المعلوم أن درجة حرارة الجسم تساوى 37 درجة مئوية تقريباً وهي أعلى من درجة حرارة جو الغرفة )

ثم لاحظ التغير في قراءة الفولت الناتج كما في الصورة التالية:

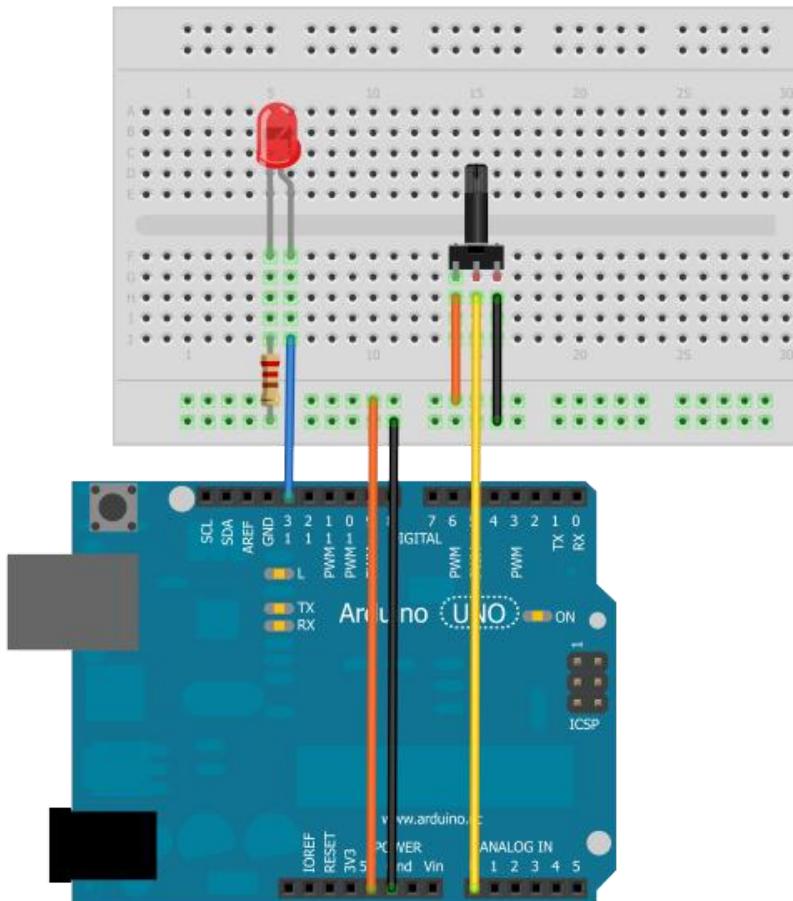


**سنجد أن قميّه الفولت قد أزدادت بسب حراًءه اليد**

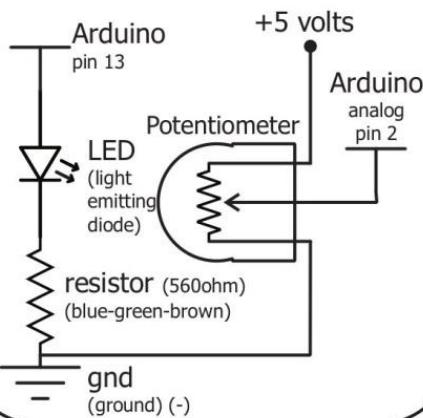
تعمل جميع الحساسات بأنواعها المختلفة بنفس المبدأ حيث يؤدي التغير في عامل بيئي معين إلى تغير المقاومة الكهربائية الداخلية مما يؤدي إلى التغير في فرق الجهد الخارج من الحساس مما يولد أشاره تماثليه متغيره يمكن قياسها ، والآن لنبدأ مع امثله استخدام الحساسات مع اردوينو

## المثال الرابع: استخدام مقاومه متغيره للتحكم فى سرعه اضائه دايدود ضوئي Potentiometer

في هذا المثال سنستخدم مقاومه متغيره للحصول على فرق جهد يمكن تغييره (دخل تماثلي) وسنجعل الدايدود الضوئي يضيء وينطفئ بسرعه تعتمد على قيمه الدخل التماثلي

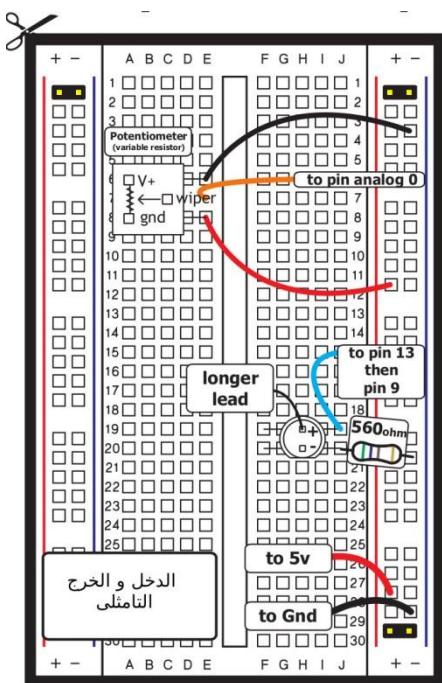


## Schematic



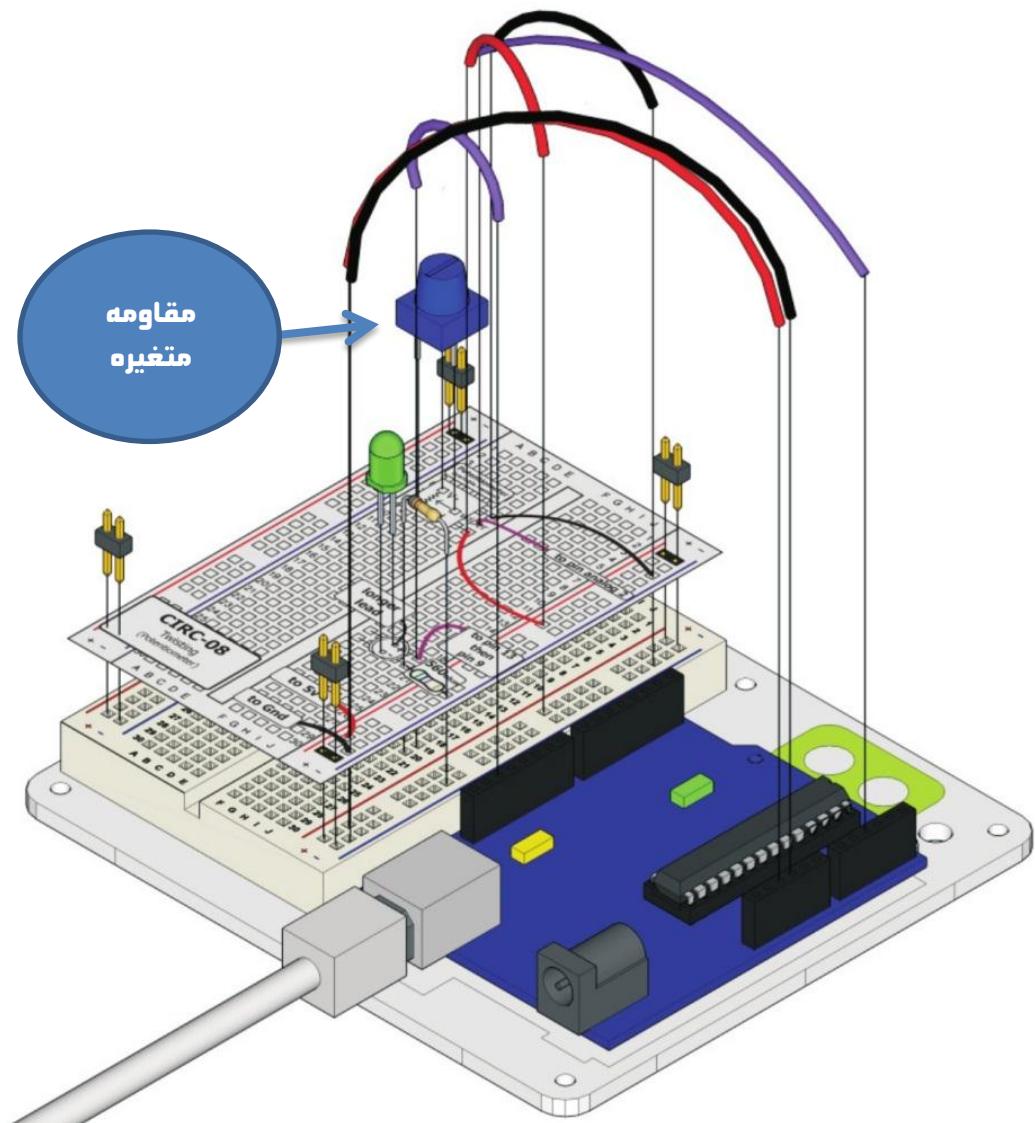
## مكونات المثال (٤):

- ✓ بورده اردوینو Arduino Uno
  - ✓ لوحة تجارب Breadboard
  - ✓ دايدود ضوئي led 5mm
  - ✓ مقاومه 560 او姆
  - ✓ مقاومه متغيره 10 اطراف 3 كيلوا
  - ✓ أسلاك توصيل
  - ✓ ورقه المساعدة المطبوعة



## قم بطباعة الورقة المساعدة ثم وصل المكونات الالكترونية على لوحة التجارب

**الملحوظه:** لا يشرط ان تستخدم مقاومه متغيره صغيره الحجم مثل التى فى الصوره ولك الحرية فى استخدام اي مقاومه متغيره لها 3 اطراف بأى حجم واي قيمه مثل 100 كيلو او 200 كيلو او م، ايضا جميع المقاومات المتغيره ذات 3 اطراف يكون الطرف الاوسط هو الطرف المتغير و الذى سنوصله بالمدخل التماذلي A0 على بورده اردوينو كما فى الصوره التالية



**بعد الانتهاء** من توصيل المكونات الالكترونية على لوحة التجارب و بورده اردوينو قم بعمل ملف جديد و اكتب الاوامر البرمجيه التالية ثم قم برفع البرنامج الي بورده اردوينو

```
//Example_4_Variable_Resistor_sensor
const int sensorPin = A0;
const int ledPin = 13;
int sensorValue;
```

```
void setup ()
{
  pinMode(ledPin, OUTPUT);
}

void loop ()
{
  sensorValue = analogRead(sensorPin);
  digitalWrite(ledPin, HIGH);
  delay(sensorValue);
  digitalWrite(ledPin, LOW);
  delay(sensorValue);
}
```

متغير لتخزين قيمة الجهد  
التماثلي

## شرح الكود البرمجي

**فى هذا المثال** سنتعرف على واحد من اهم الدوال البرمجية الموجودة فى لغه اردوينو **analogRead(pin number)** تقوم هذه الدالة بقراءه فرق الجهد بصورة تماثلية و تستطيع المتحكمه الدقيقه ان تقيس فرق جهد من 4.8 مللى فولت حتى 5 فولت تقريباً و تقوم بتحويل الاشاراه التماثلية الى قيمه رقميه من صفر الى 1024 و تسمى هذه العمليه **Analog to digital converting** باسم تحويل الاشاره من تماثلية الى رقميه

**مثلاً** اذا كان الجهد الداخل الى A0 يساوى القيم التالية:

رقمي	4.8 مللى فولت = 1
رقمي	48 مللى فولت = 10
رقمي	480 مللى فولت = 100
رقمي	واحد فولت = 208.33
رقمي	2 فولت = 416.66
رقمي	5 فولت = 1024

ويستخدم الامر التالي في تخزين قيمة القراءه الناتجه داخل المتغير sensorValue

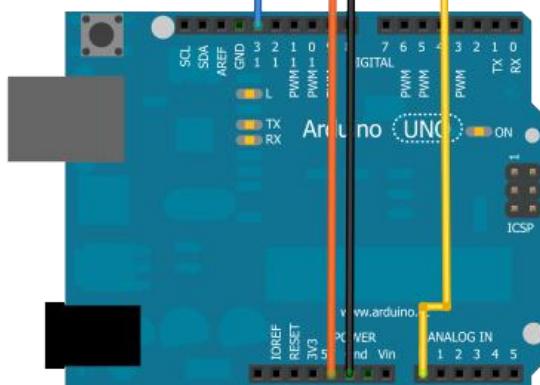
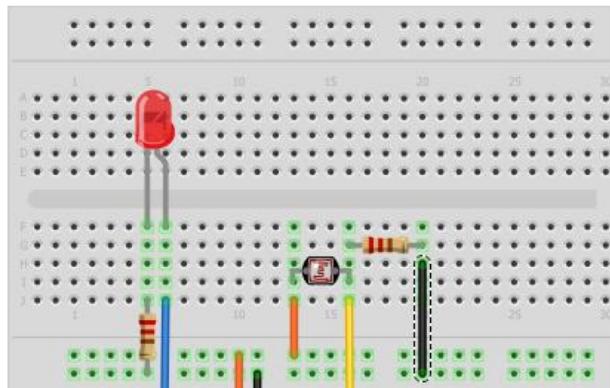
**sensorValue = analogRead(sensorPin);**

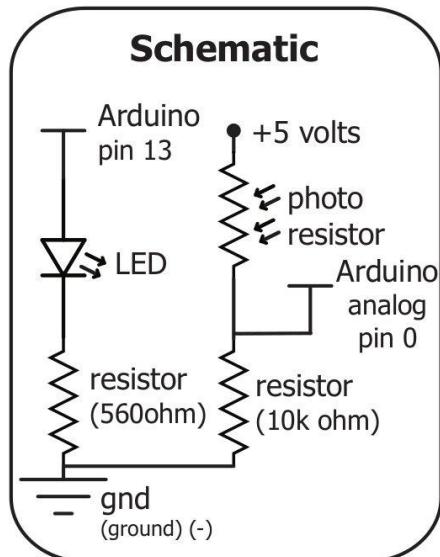
**بعد ذلك** تقوم المتحكمه بتشغيل الدايدود الضوئي فتره زمنيه تساوى قيمة هذا المتغير و تقوم بأطفائه بنفس الزمن مثل المثال الاول led blinking ولكن الزمن في هذا المثال متغير تبعا لقراءه الجهد التماشى على المدخل A0.

في هذا المثال استخدمنا مقاومه متغيره يمكننا ان نغير قيمتها عن طريق لف (تدوير) عمود الدوران الخاص بها وبذلك نحصل على اشاره تماثليه (تسمى تلك العمليه تغير منحكم به من الانسان) ، في المثال القادم سنتعلم نوع جديد من المقاومات المتغيره وهي المقاومه الضوئيه التي تتغير قيمتها بتغير كثافه الضوء الساقط عليها ويسمى هذا تغير بيئي و سوف نستغل ظاهره التغير في المقاومه بتغير الضوء لتحول المقاومه الى مستشعر للضوء light sensor

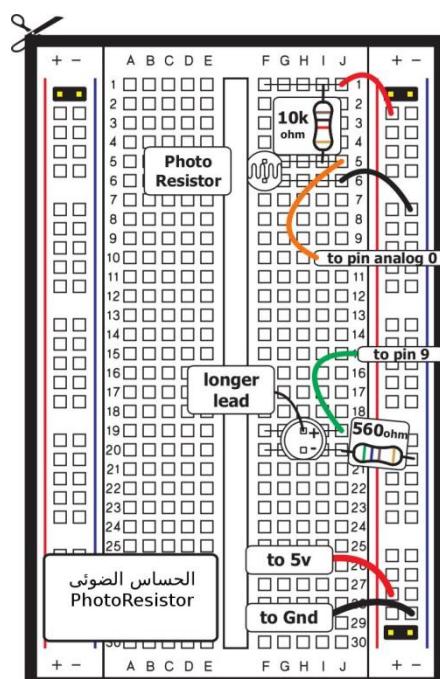
## المثال الخامس: المقاومه الضوئيه (حساس الضوء التماضي) Photo resistor as light sensor (Photoresistor)

**تصنع** المقاومات الضوئية عادة من نوع خاص من السيليكون المعالج، و يعتبر السيليكون المعالج ماده شبهه موصله للكهرباء و تتأثر مدي توصيليته للكهرباء بالضوء لذلك يستخدم في الحساسات الضوئية والخلايا الشمسية .

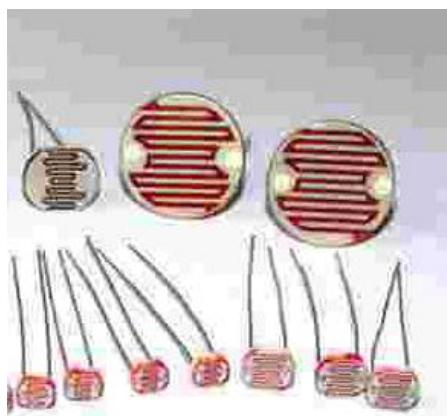


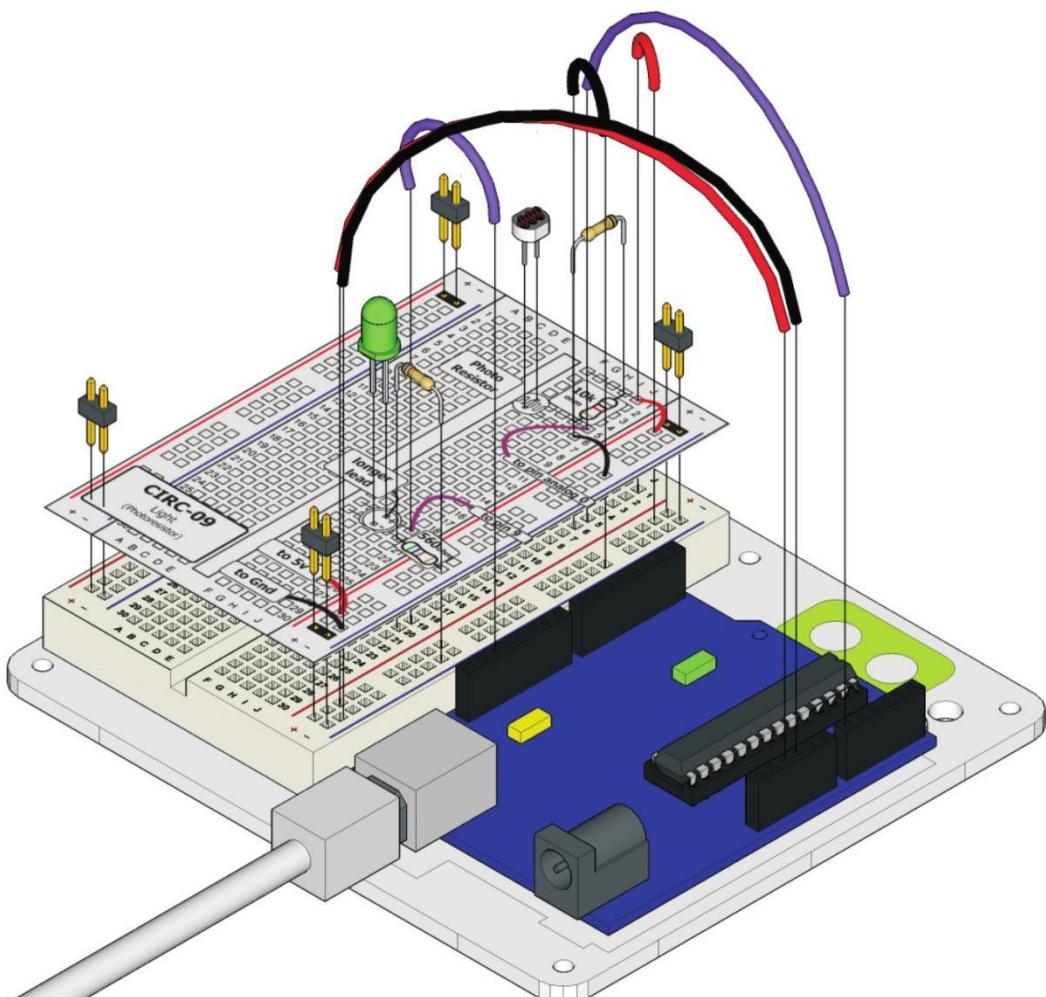
**مكونات المثال (٥):**

- ✓ بورده اردوینو Uno
- ✓ لوحة تجارب Breadboard
- ✓ دايدود ضوئي led 5mm
- ✓ مقاومه 1560 او姆
- ✓ مقاومه ضوئيه
- ✓ اسلاك توصيل
- ✓ ورقه المساعده المطبوعه



توفر المقاومات الضوئية في احجام مختلفه تبعا  
للمقدار حساسيتها للضوء و يمكن استخدام اي  
وفر حجم مت





```
//Example_5_Light_Sensor_led  
const int lightPin = A0;  
const int ledPin = 9;  
int lightLevel;
```

```
void setup ()  
{  
    pinMode(ledPin, OUTPUT);  
}
```

```
void loop ()  
{  
    lightLevel = analogRead(lightPin);  
    lightLevel = map(lightLevel, 0, 900, 0, 255);  
    lightLevel = constrain(lightLevel, 0, 255);  
    analogWrite(ledPin, lightLevel);  
}
```

قم بتنفيذ البرنامج ورفعه على بورده المتحكمه الدقيقه ثم شاهد ماذا سيحدث للدايود الضوئي عند تعریض المقاومه الضوئيه للضوء، ثم حاول ان تقلل مقدار الضوء الساقط على المقاومه الضوئيه عن طريق تغطيتها بيده و شاهد ماذا سيحدث الى الديود الضوئي.

### رقم المخرج

### قيمة الخرج



الآن سنتعرف على الدالة `analogWrite (pin number, value);` و هي المسئولة عن توليد خرج تماثلي `analog Output` و يمكن تطبيقها على المخارج التي تدعم خاصية التعديل الرقمي المعتمد على عرض النسبة (Pulse-Width modulation) وهي المخارج التي يكتب بجانبها علامه (~). و هي 6 مخارج كالتالي: [3,5,6,9,10,11](#).

## ما هي خاصية التعديل الرقمي ؟؟ PWM

خاصية يمكنك من انتاج فرق جهد على هيئة موجه قابلة للتغير عن طريق اشاره رقميه، وبذلك تستطيع أن تحول الأوامر الرقمية إلى موجه تماثلية و يمكن استغلال تلك الخاصية في توليد جهد متغير قيمته بين صفر فولت حتى 5 فولت عن طريق استخدام قيم رقميه من 0 حتى 255 مثلا:

$$255 = 5 \text{ volt}$$

$$128 = 2.5 \text{ volt}$$

$$0 = 0 \text{ volt}$$

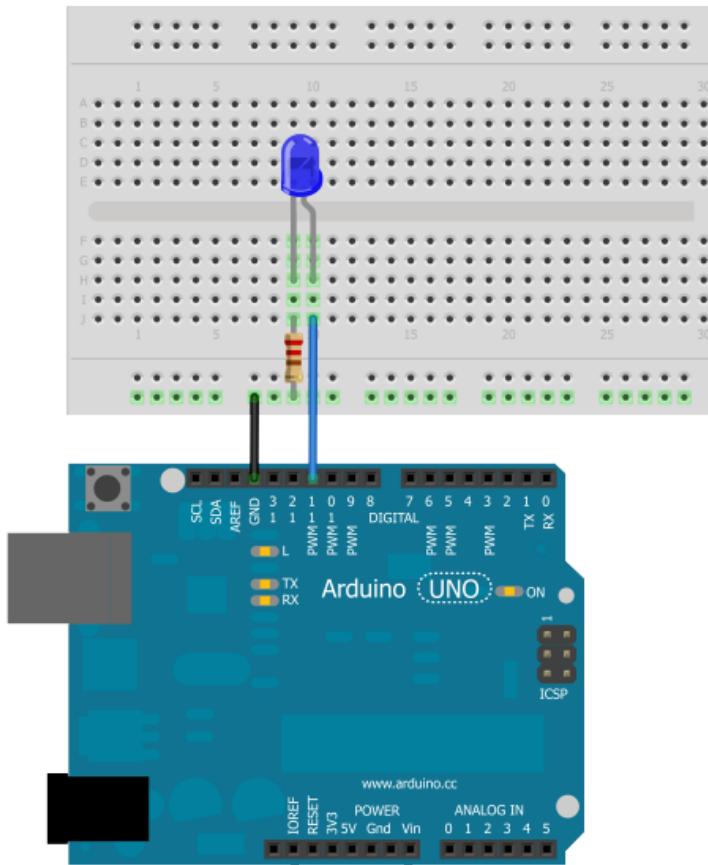
## كيف نستفيد من هذه الخاصية ؟؟

الكثير من المكونات الالكترونيه والكهريه تتفاعل مع القيم المختلفه من فرق الجهد بصور مختلفه مثل الديايد الضوئي، مثلا عندما تطبق 3 فولت على الديايد الضوئي ستجد ان اضافته ضعيفه و اذا رفعت الجهد الى 4 فولت ستجد الاضافه قد ازدادت و اذا وصلت الى 5 فولت ستجد الاضافه وصلت الى اعلى معدل سطوع.. وهناك مكونات كهربيه مثل

المحرك الكهربى (الموتور) والذى تزداد سرعته بزيادة الفولت الكهربى.  
في هذا المثال وصلنا الدايدود الضوئي على المخرج رقم 9 (الذى يدعم PWM) واستخدمنا مقاومه ضوئيه حتى تستشعر كميه الضوء الموجود حولها ثم ترسل الاشاره التمايليه الى اردوينو فيقوم بتشغيل الليد بنائا على الاشاره التمايليه من المقاومه الضوئيه .

لنأخذ مثال آخر على خاصيه التعديل الرقمي

## المثال السادس: المؤثرات الضوئية باستخدام PWM



في هذا المثال سنقوم بتوصيل دايمود ضوئي ازرق (اخترت اللون الأزرق لأن أضاءته عالية مقارنة بالألوان الأخرى و يمكنك اختيار اللون الأبيض كذلك) إلى المنفذ رقم 11 والذى يدعم خاصية PWM و سنستغل تلك الخاصية في عمل بعض المؤثرات الضوئية الجميلة

والآن نأتي إلى الكود البرمجي:

```
// Example_6_fading_led_light
const int ledPin = 11;
int i = 0;

void setup()
{
    pinMode(ledPin, OUTPUT);
}

void loop()
{
    for (i = 0; i < 255; i++)
    {
        analogWrite(ledPin, i);
        delay(10);
    }
    for (i = 255; i > 0; i--)
    {
        analogWrite(ledPin, i);
        delay(10);
    }
}
```

العداد الأول

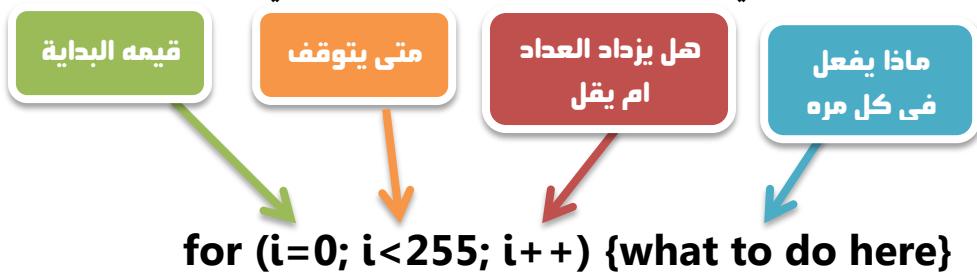
العداد الثاني

بعد الانتهاء من كتابة الكود و رفعه إلى المتحكم الدقيقة.. لاحظ ماذا يحدث للدايمود

الضوئي وكيف تزداد إضاءته بهدوء و تنخفض بهدوء ☺

## سنتعلم الأن أحدى أهم الأوامر البرمجية وهي حلقات التكرار

في هذا المثال شاهدنا امر جديد وهو `for (i = 0; i < 10; i++)` يستخدم هذا الأمر في عمل ما يسمى بالحلقة المتكررة loop ويمكن استخدام الحلقات في أداء وظيفه تكرر أكثر من مره بعد عدد معروف من المرات أو الي ما لا نهاية و يمكن استغلال الحلقات في عمل ما يسمى بالعدادات counters وهي برمج صغيره تقوم بالعد التزايدى أو التناصي كما يريد المبرمج ، و تكون صيغه الأمر كتالى:



في المثال السابق استخدمنا عداد يقوم بعد الأرقام ابتدائياً من صفر حتى 255 وفي كل مره يضيف واحد ويخزن القيمة في المتغير (i) ثم يقوم بتطبيق جهد تماثلي بقيمته على الدايوه الضوئي باستخدام الأمر `analogWrite(ledPin,i)` وبذلك نحصل على ذلك التأثير الضوئي الجميل حيث نرى الدايوه الضوئي يبدأ في الوميض بصورة خافتة ثم تزداد شده الإضاءة تدريجياً بزيادة قيمة المتغير (i) في العداد الأول.

**في العداد الثاني** استخدمنا نفس المبدأ لكن بصوره معكوسه بحيث يبدأ العداد من 255 ويظل الرقم يتناقص بمقدار واحد حتى يصل إلى الصفر وفي كل مره يسجل العداد قيمة المتغير (a) ويطبق جهد تماثلي على الدايوه الضوئي بقيمه (a) وهذا ما يجعل الدايوه الضوئي يبدأ في الوميض بشده ثم تنخفض شده إضاءته تدريجيا بتغير قيمه (a)

### قد يتسائل البعض لماذا استخدمنا الأمر delay(10); داخل كل عداد ؟؟

الأجابة تمثل في التحكم في سرعه اردوينو اثناء تطبيق الجهد التماثلى ولكن يتضح المعنى بصوره عمليه جرب ان تغير الرقم و تجعله مره = 20 بحيث يكتب الأمر delay(20); و مره أخرى تجعله 50 و مره أخرى تجعله = 5 و شاهد ماذا سيحدث للدايوه الضوئي.

## ملاحظات شخصيه :

هذه الصفحة مخصصة لكتابه ملاحظاتك الشخصيه عن الفصل الرابع :

اكتب ملاحظاتك  
 هنا

## مُلْكُ الْعِلْمِ الْعَظِيمُ :

العلم مغرس كل فخر فافتخر  
واحدر يفوتك فخر ذاك المدرس

واعلم بأن العلم ليس يناله  
من همه في مطعم أو ملبس

الإمام الشافعى

# الفصل الخامس

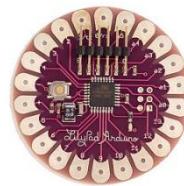
أرسال البيانات و الأوامر من اردوينو إلى الحاسوب الآلي و العكس

Computer interfacing with Arduino



## جميع بوردات اردوينو (lilypad, mini)

توفر إمكانية أرسال و استقبال البيانات مباشره مع الحاسب الآلي computer interface وذلك عن طريق منفذ USB و يمكن استغلال هذه الخاصية في الكثير من المشاريع الرائعة، وبالنسبة إلى البويرات التي لا تدعم هذه الخاصية فهي



Lilypad

تحتاج إلى ما يسمى بالـ FTDI interface وهي شريحة صغيرة تستخد لتبادل البيانات بين الحاسب الآلي و المتحكمات الدقيقة

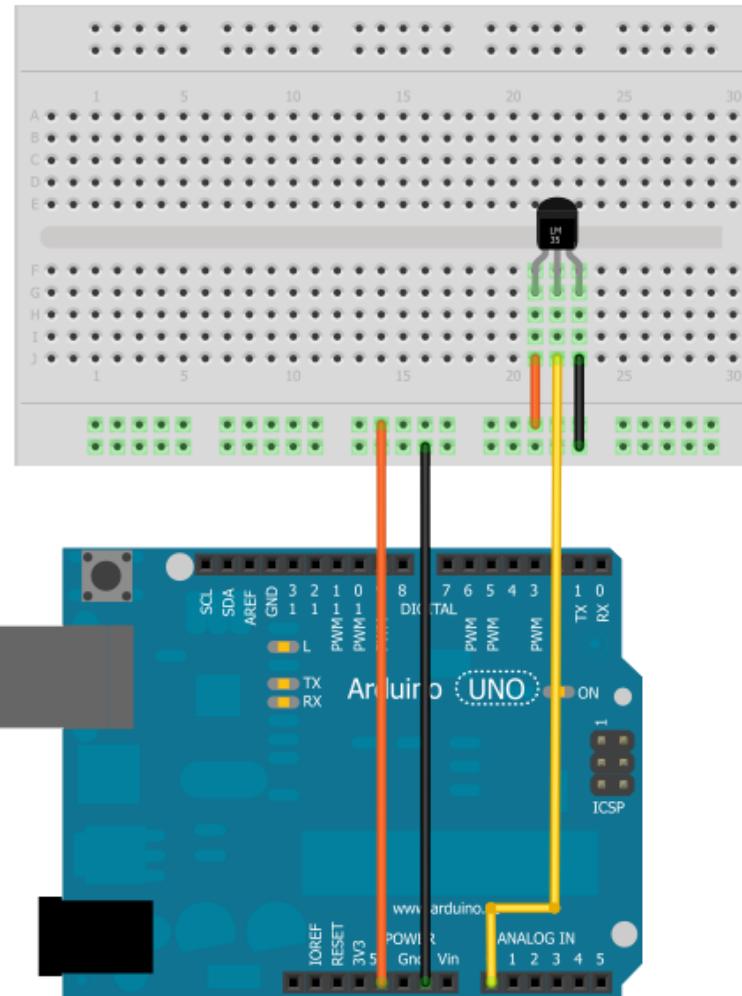


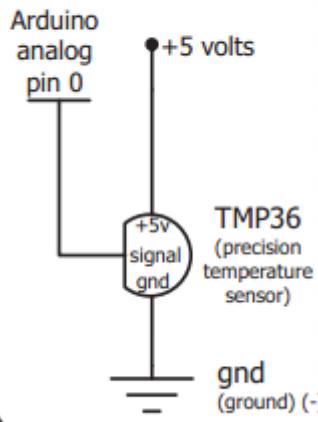
FTDI chip

**يتحدث** الفصل الخامس عن إمكانية استخدام اردوينو في الحصول على معلومات من الحساسات المختلفة وأرسال النتائج لعرضها على الحاسب الآلي أو أخذ أوامر من الحاسب الآلي لتنفيذها على المكونات الإلكترونية الموصولة على البوير.

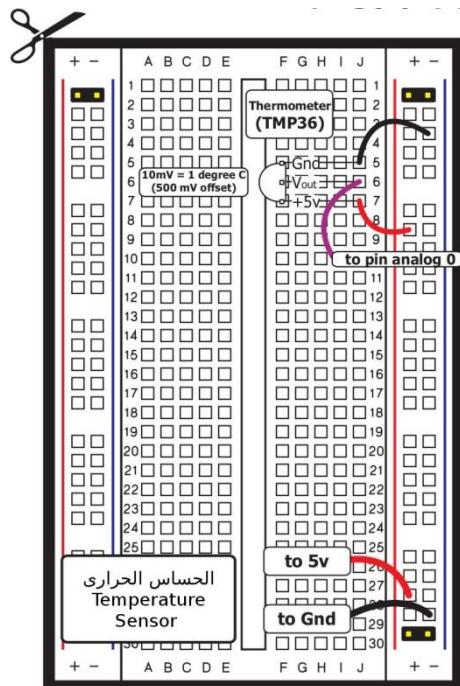
**في الأمثلة السابقة** كنا قد استخدمنا اردوينو في قراءة بعض الحساسات مثل الضوء والحرارة وإظهار ناتج الحساس على دايمود ضوئي ليوضح مقدار هذه القراءة أما الأن فسنستخدم خاصيه serial interface لأرسال نتائج القراءة للحساسات إلى الحاسب الآلي حيث يمكن قرائة النواتج بالأرقام وإجراء بعض العمليات الحسابية عليها.

## المثال السابع: استخدام حساس الحرارة و أرسال درجة الحرارة إلى الحاسوب الآلي

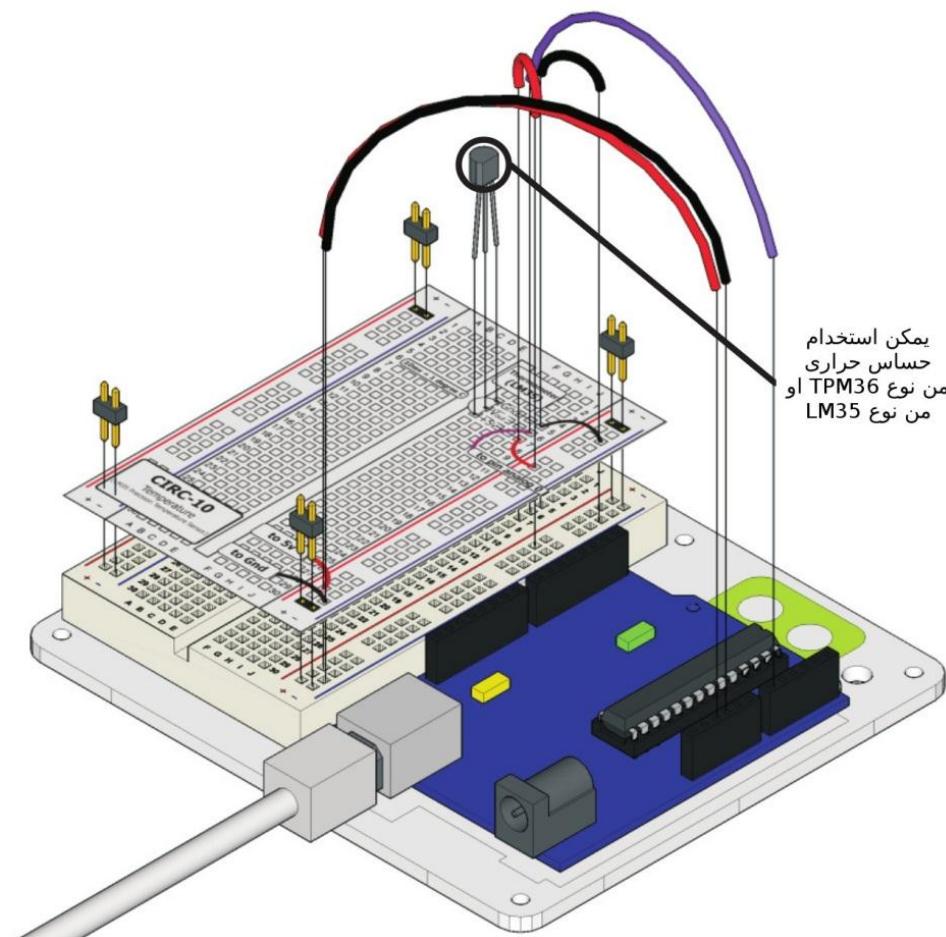


**Schematic****مكونات المثال (7):**

- ✓ بورده اردوینو Uno
- ✓ لوحة تجارب Breadboard
- ✓ حساس حراري TMP36 و LM35
- ✓ أسلاك توصيل
- ✓ كابل التوصيل بالUSB
- ✓ ورق المساعدة المطبوعة



لاحظ انه في حالة استخدامك لحساس حراري مختلف عن النوعين المذكورين في الأعلى فإن المداخل والمخارج الخاصة بالجهد قد تختلف و يمكنك معرفتها من خلال قراءة الداتا شيت الخاصة بالحساس المستخدم



بعد الانتهاء من توصيل المكونات على لوحي التجارب قم بكتابه الأكواط التالية وتأكد من صحتها عن طريق زر Verify ثم ارفعها إلى بورده اردوينو

```
// Example_7_Temprature_Sensor

const int sensorPin = A0;

int reading;

float voltage;

float temperatureC;

void setup( )

{ Serial.begin(9600); }

void loop ( )

{

reading = analogRead(sensorPin);

voltage = reading * 5.0/1024;

Serial.print(voltage);

Serial.println(" volts");

temperatureC = (voltage - 0.5) * 100 ;

Serial.println("Temperature is: ");

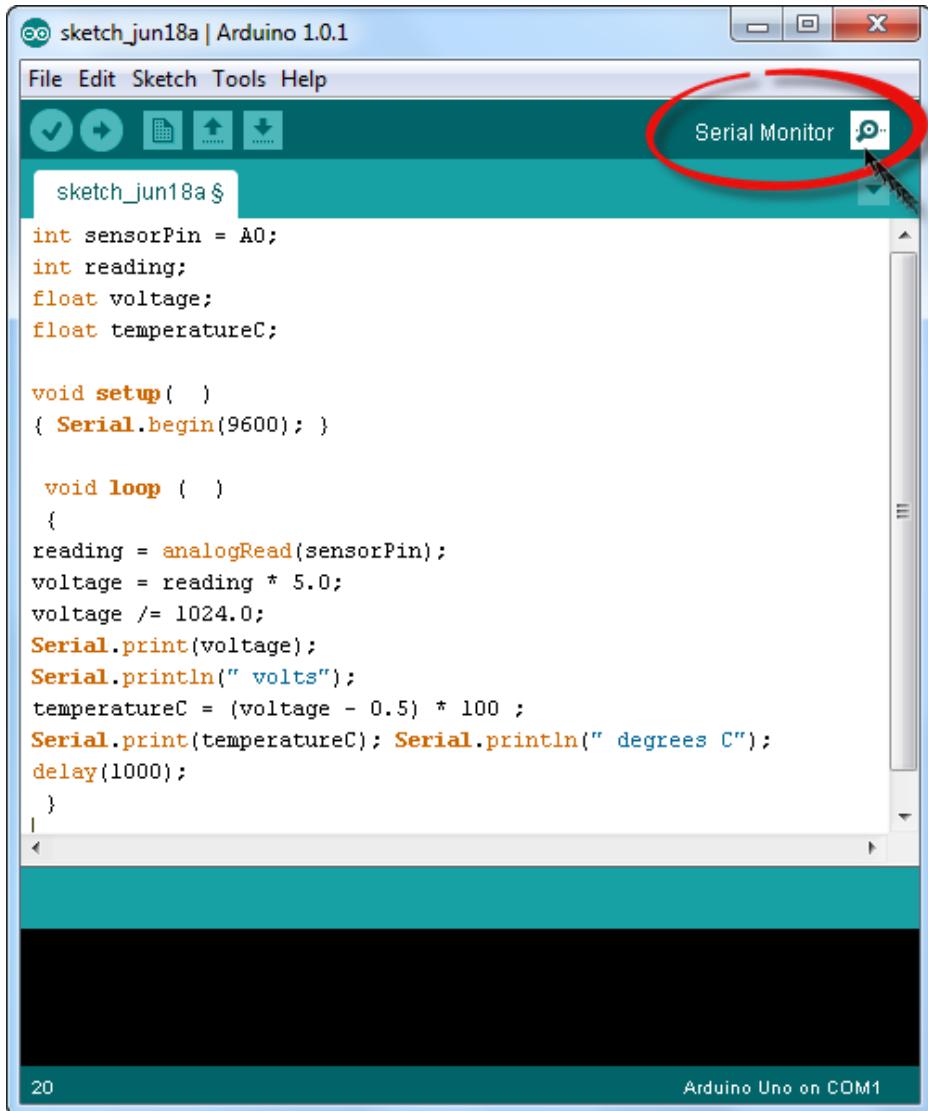
Serial.print(temperatureC);

Serial.println(" degrees C");

delay(1000);

}
```

بعد الانتهاء من كتابة ورفع الكود البرمجي إلى المتحكم يقوم بالضغط على زر Serial موجود في شريط الأوامر السريعة كما في الصورة التالية:



ستظهر صفحه خاصه تستقبل البيانات من بورده اردوينو و تقوم بعرض درجه الحرارة كل ثانية كما سنلاحظ أن الدايمود الضوئي الموجود على بورده اردوينو المكتوب بجانبه X بدأ في الوميض بسرعه وذلك يعني أن اردوينو يرسل بيانات إلى الحاسب الآلي ...

```

0.29 volts
Temperature is:
-21.19 degrees C
0.28 volts
Temperature is:
-21.68 degrees C
  
```

Autoscroll      No line ending      9600 baud

الآن حاول ن ترفع درجه حراره الحساس عن طريق تقرير شمعه مشتعلة أو عود ثقاب مشتعل أو أي مصدر حراري متوفّر (حذاري أن تمّس النار الحساس الحراري فذلك قد يتلفه ، هناك حساسات حرارية خاصة يمكنها تحمل الحرارة الشديدة لكن الحساس المستخدم في هذا المثال يستطيع تحمل حراره تصل إلى 150 درجه مئوية تقريبا)

ملحوظه: علامه الشرطة (-) التي تكتب قبل درجه الحرارة لا تعنى سالب درجه مئوية و إنما خطأ برمجي مؤقت في أول السطر عند استقبال بيانات من اردوينو

## شرح الكود البرمجي

**يستخدم الأمر Serial.begin(9600);** في جعل بورده اردوينو تبدأ التواصل مع الحاسب الآلي عن طريق منفذ USB و بهذا يمكن للاردوينو أرسال أو استقبال بيانات من وإلى الحاسب.

نلاحظ انه هناك متغيران و هما temperatureC و voltage تم تعريفهم باستخدام الأمر float بدلا من استخدام الأمر int و ذلك لان الحساس الحراري يقيس درجه الحرارة بدقة عالية تصل إلى 0.1 درجه مئوية و من المتوقع أن تكون قيمه الجهد الكهربائي الناتج عنه و درجه الحرار بالكسور العشرية و ليس الأعداد الصحيحة لذلك استخدامنا الأمر float لجعل هذه المتغيرات تقبل قيمه تحتوى على كسور عشرية .

**يستخدم الأمر reading = analogRead(sensorPin);** في تسجيل قراءة الدخل التماثلي على المتغير reading و هي القيمة الناتج من sesnorPin و الذي يرمز إلى المدخل A0 .

كما ذكرنا سابقا ان المحكمة الدقيقة تقوم بتحويل الاشارة التماثلية إلى قيمه رقميه من صفر الى 1024 لذلك استخدمنا الأمر التالي لتحويل هذه القيمة الرقمية إلى قيمه فولتية نستطيع فهمها

```
voltage = reading * 5/1024;
```

بعد تحويل القيمة الرقمية إلى قيمه فولتيه استخدمنا الأمر Serial.print(voltage);  
لأرسال هذه القيمة إلى الحاسب الآلي لعرضها على بيئه التطوير باستخدام الـ Serial  
ثم نستخدم الأمر Serial.println ("voltage"); يقوم بطباعه كلمه Monitor  
بعد كل قيمه فولت يعرضها على شاشه الحاسب الآلي voltage

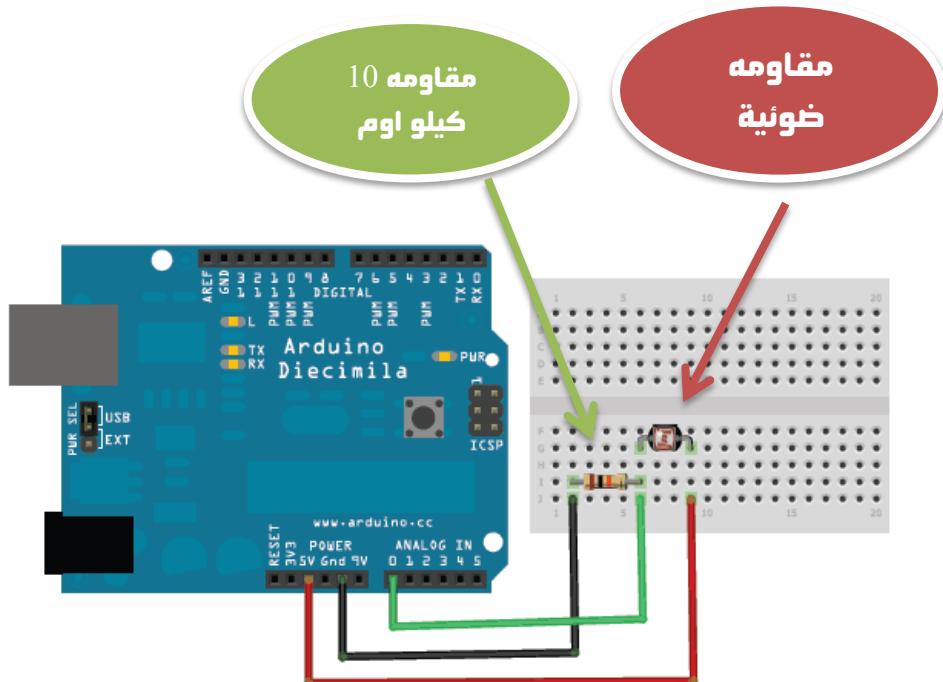
نستخدم الأمر temperatureC = (voltage - 0.5) \* 100; لتحويل قيمه الفولت  
إلى درجه حراره بوحده سي لزيوس Celsius ثم نكرر نفس اوامر Serial.print مع  
درجة الحرارة لعرضها على شاشه الحاسب الآلي

```
Serial.print(temperatureC);
Serial.println(" degrees C ");
```

الأمر الأخير هو delay(1000); وذلك لنجعل المتحكمه الدقيقه تنتظر ثانيه واحده  
قبل أن تعيد أرسال درجه الحرارة و الفولت إلى الحاسب الآلي مره أخرى.

## المثال الثامن: استخدام حساس الضوء و عرض شدء الإضاءة على الحاسب الآلي

هل تذكر المثال الخامس، سنتقوم بأعاده تنفذ ذلك المثال ولكن بدلا من عرض شده الإضاءة على الديايد الضوئي led ، سنعرضها على الحاسب الآلي.



```
// Example_8_Light_Sensor_2

const int photocellPin = A0;

int photocellReading;

void setup(void)
{ Serial.begin(9600); }

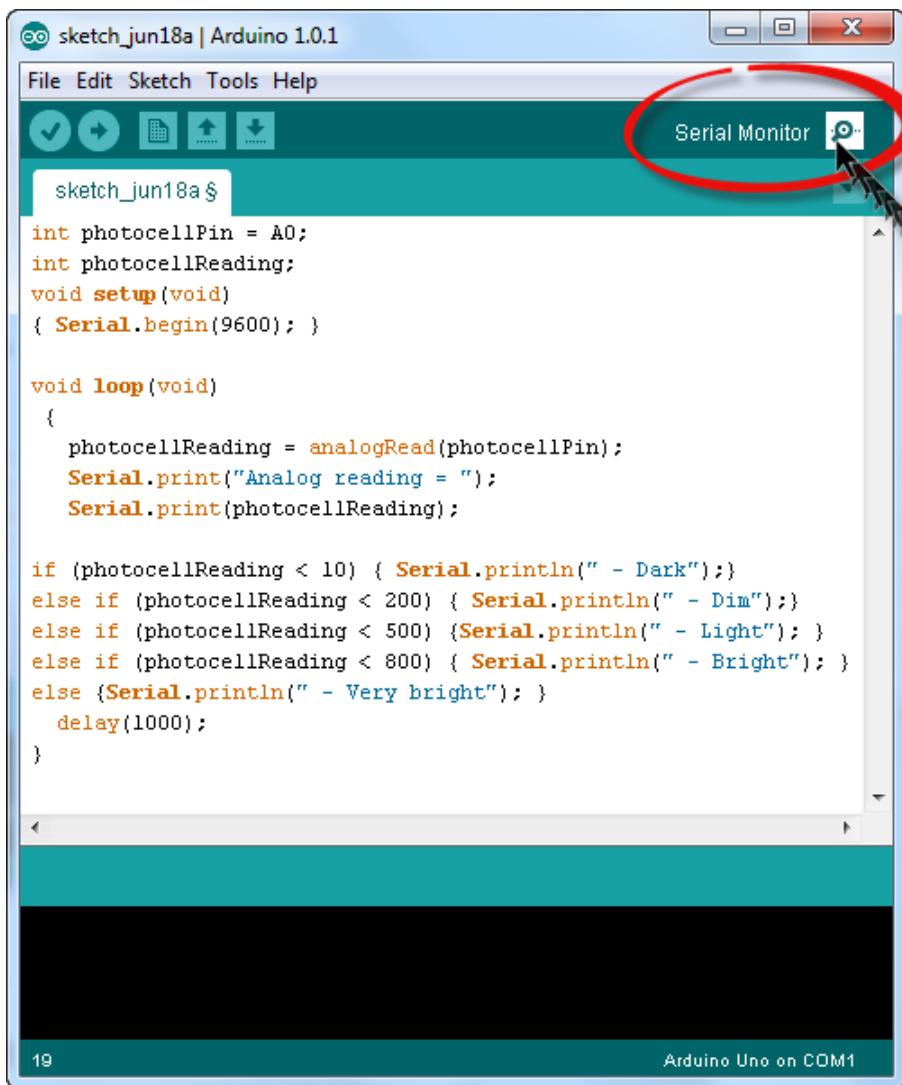
void loop(void)
{
    photocellReading = analogRead(photocellPin);

    Serial.print("Analog reading = ");
    Serial.print(photocellReading);

    if (photocellReading < 10) { Serial.println(" - Dark");}
    else if (photocellReading < 200) { Serial.println(" - Dim");}
    else if (photocellReading < 500) {Serial.println(" - Light");}
    else if (photocellReading < 800) { Serial.println(" - Bright");}
    else {Serial.println(" - Very bright");}

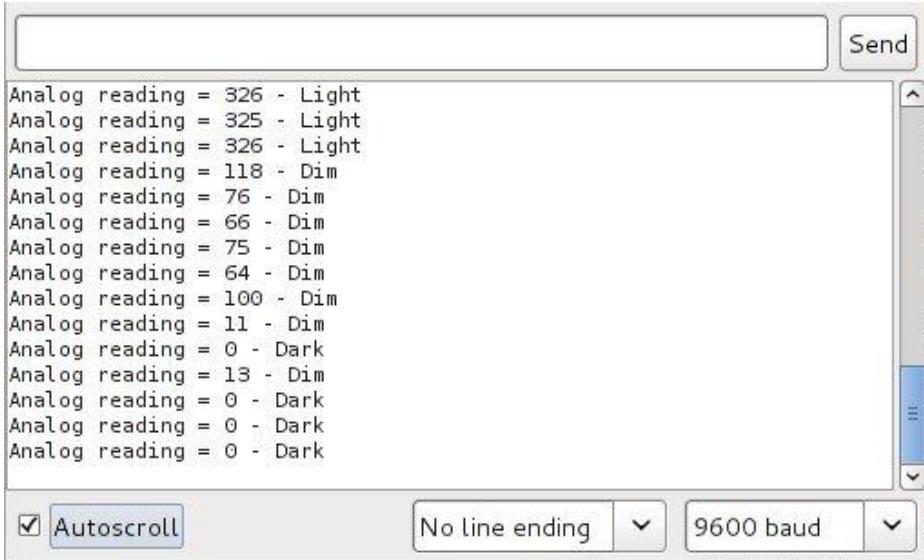
    delay(1000);
}
```

**بعد الانتهاء** من كتابه ورفع الكود البرمجي إلى المتحكم نقوم بالضغط على زر **Serial monitor** الموجود في شريط الأوامر السريعة كما في الصورة التالية :



**الآن حاول** أن تغير مستوى الإضاءة الذي تتعرض له المقاومة الضوئية عن طريق :

- تعریضها للضوء الساطع مباشره
- غططيتها بقطعة قماش خفيفة
- غططيتها بيديك لتجحّب عنها الضوء تماماً



```
Analog reading = 326 - Light
Analog reading = 325 - Light
Analog reading = 326 - Light
Analog reading = 118 - Dim
Analog reading = 76 - Dim
Analog reading = 66 - Dim
Analog reading = 75 - Dim
Analog reading = 64 - Dim
Analog reading = 100 - Dim
Analog reading = 11 - Dim
Analog reading = 0 - Dark
Analog reading = 13 - Dim
Analog reading = 0 - Dark
Analog reading = 0 - Dark
Analog reading = 0 - Dark
```

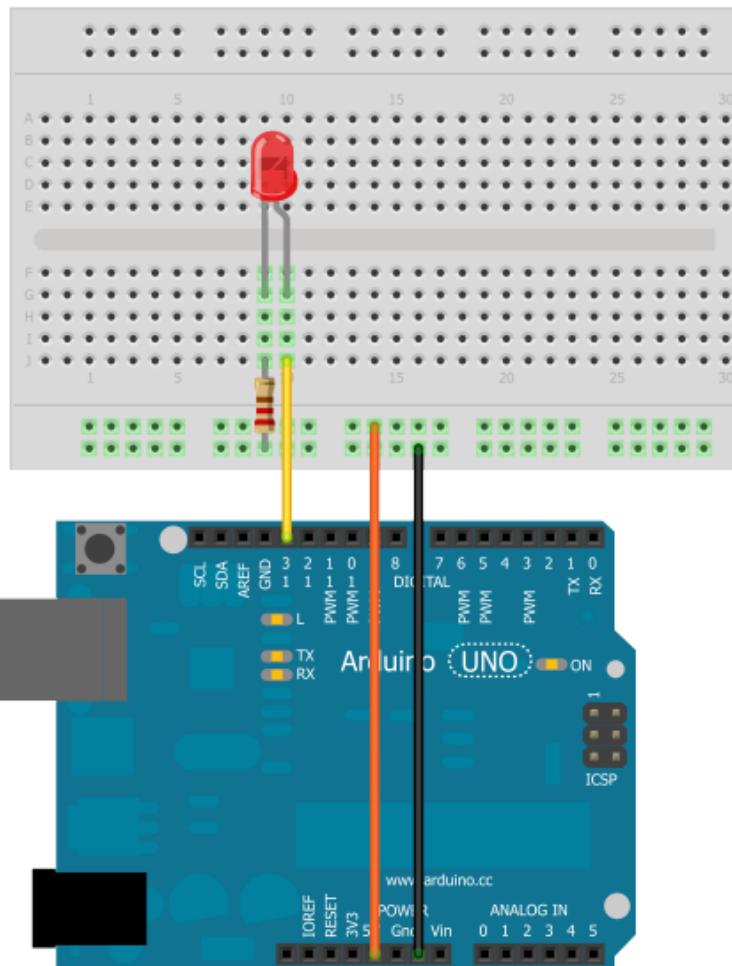
Serial monitor settings at the bottom:

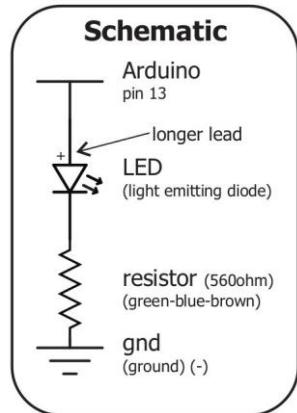
- Autoscroll
- No line ending
- 9600 baud

**ملحوظه:**

- تعنى إضاءة خافتة Dim
- تعنى ظلام دامس Dark
- تعنى إضاءة معتدلة Light
- تعنى إضاءة شديدة Bright Light

# المثال التاسع: تشغيل دايوود ضوئي عن طريق استقبال أمر من الحاسب الآلي



مكونات المثال (٩):

- ✓ بورده اردوینو Arduino Uno
- ✓ لوحة تجارب
- ✓ دايدود ضوئي led 5mm
- ✓ مقاومه 560 او姆
- ✓ أسلاك توصيل

يعتبر المثال التاسع تطوير للمثال الأول والثاني ، وفي هذا المثال سنستخدم الحاسب الآلي في التحكم في الدايدود الضوئي بدلاً من السويتش، سنجعل اردوينو يستقبل امر التشغيل والإطفاء عن طريق الـ USB باستخدام خاصيه الـ Serial Monitor في بيئة Arduino IDE

الكود:

```
// Example_9_Computer_Interfacing

int ledPin=13;
int value;

void setup ()
{
    Serial.begin(9600);
    pinMode(ledPin,OUTPUT);
}

void loop ()
{
    value = Serial.read();
    if (value == '1') {digitalWrite(ledPin,HIGH);}
    else if (value == '0') {digitalWrite(ledPin,LOW);}
}
```

**بعد الانتهاء** من كتابة الكود البرمجي ورفعه إلى اردوينو اضغط على أيقونة serial Monitor ستجد شريط كتابي في أعلى الصفحة قم بكتابه 1 واضغط زر أرسال وشاهد ماذا سيحدث للدایود الضوئی ، ثم اكتب الرقم صفر واضغط على زر أرسال مره أخرى وشاهد ماذا سيحدث ...



في هذا المثال استخدمنا الأمر `Serial.read()` وهو الامر المستخدم في قراءة البيانات المرسلة من الحاسب الآلي إلى اردوينو عبر منفذ USB و قمنا بإضافة `value=` قبل هذا الأمر و ذلك حتى تقوم المتحكمه بقراءة ما يرسل من الـ USB، تخزن هذه القيمة في المتغير `value`

ثم استخدمنا جمله `if.. else if` لوضع شرط:

- اذا كانت قيمة المتغير `value == 1` تقوم المتحكمه بتشغيل الدايمود الضوئي
- أما اذا كانت قيمة المتغير `value == 0` تقوم المتحكمه بإطفاء الدايمود الضوئي

**الآن حاول أن تعدل الكود بنفسك و تزيد عدد امدادات التي يمكن التحكم بها عن طريق الحاسب الآلي**

**مصادر اضافيه للمعلومات:**

لمزيد من المعلومات عن الأوامر المستخدمة في الـ Serial communication تفضل الرابط التالي لمراجع الأوامر الرسمي:

<http://arduino.cc/en/Reference/serial>

## ملاحظات شخصيه :

هذه الصفحة مخصصة لكتابه ملاحظاتك الشخصيه عن الفصل الخامس :

اكتب ملاحظاتك  
 هنا

# ଶାନ୍ତିକାଳ ପାଦଗୀ

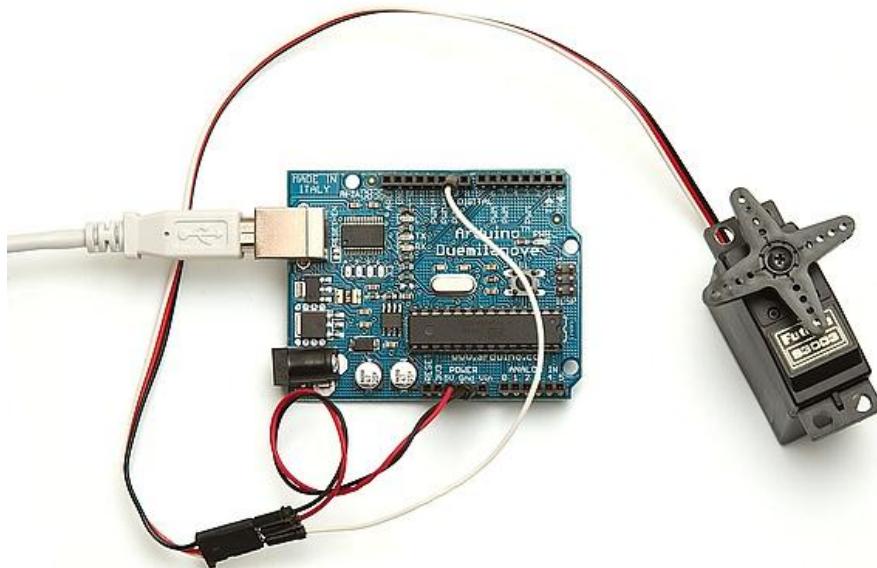
## المعرفه قوه في يد من يعلمها

أَلْبَرْ أَبْنَسْتِينْ

# الفصل السادس

استخدام المحركات

The Motors



تعتبر المحركات من أهم العناصر الداخله فى المشاريع الالكترونويه و هى العنصر المسؤول عن تحويل الطاقه الكهربائيه الى طاقه حرکيه فى صوره دوران .



ستجد المحركات فى كثير من الأجهزه الالكترونويه التي يوجد بها حركه ميكانيكيه مثل: الروبوتات بأنواعها المختلفه سواء كانت صناعيه او روبوتات للترفيه، مشغلات الأقراص، ألعاب الأطفال.. الخ

تنقسم المحركات الكهربائيه الى نوعين رئيسين و هما :

- محركات التيار المستمر (DC – Servo – Stepper)
- محركات التيار المتردد (3 Phase – 1 Phase)

فى هذا الفصل سنتعرض للنوع الأول و هو المحركات التي تعمل بالتيار المستمر مثل ال DC Motor و ال Servo Motor و هما أشهر أنواع المحركات المستخدمه فى مشاريع المتحكمات الدقيقه بأنواعها المختلفه.

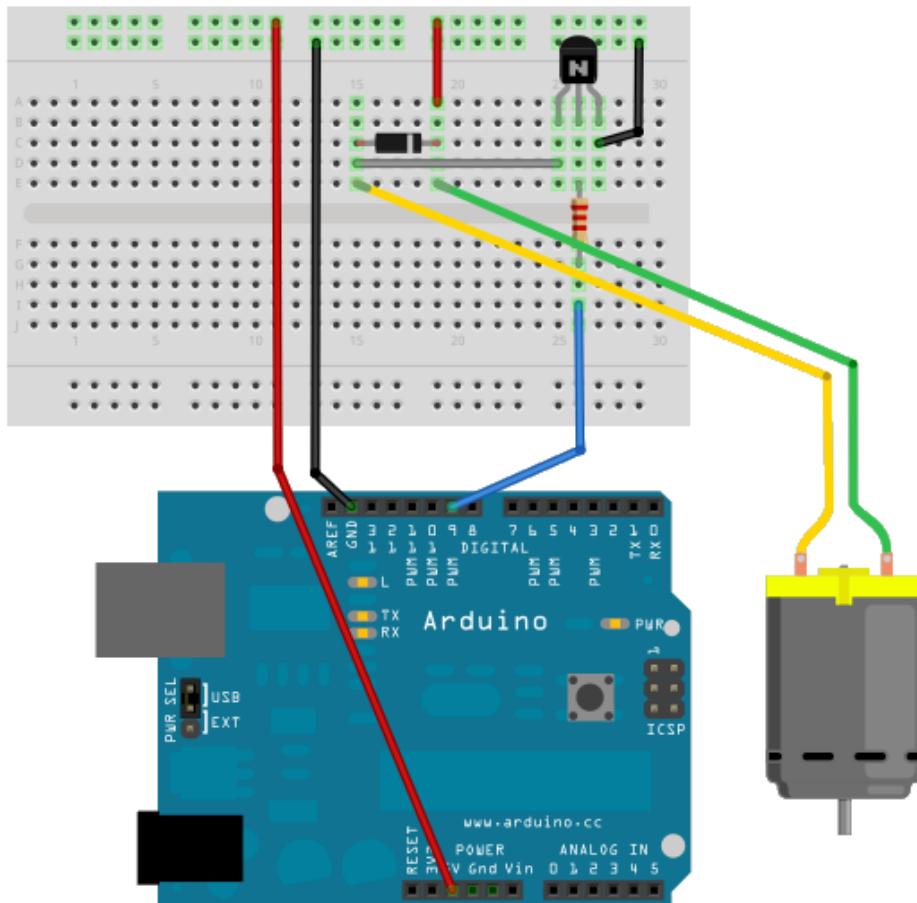


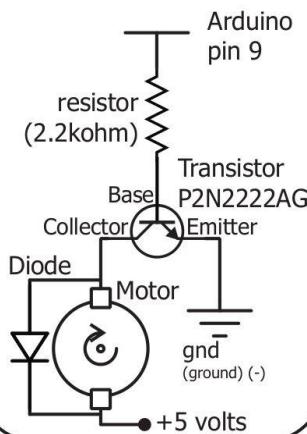
Servo Motor



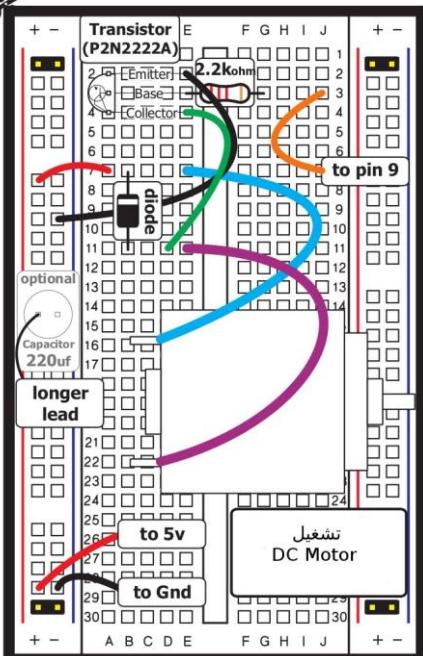
DC Motor

## المثال العاشر: استخدام محرك التيار المستمر DC



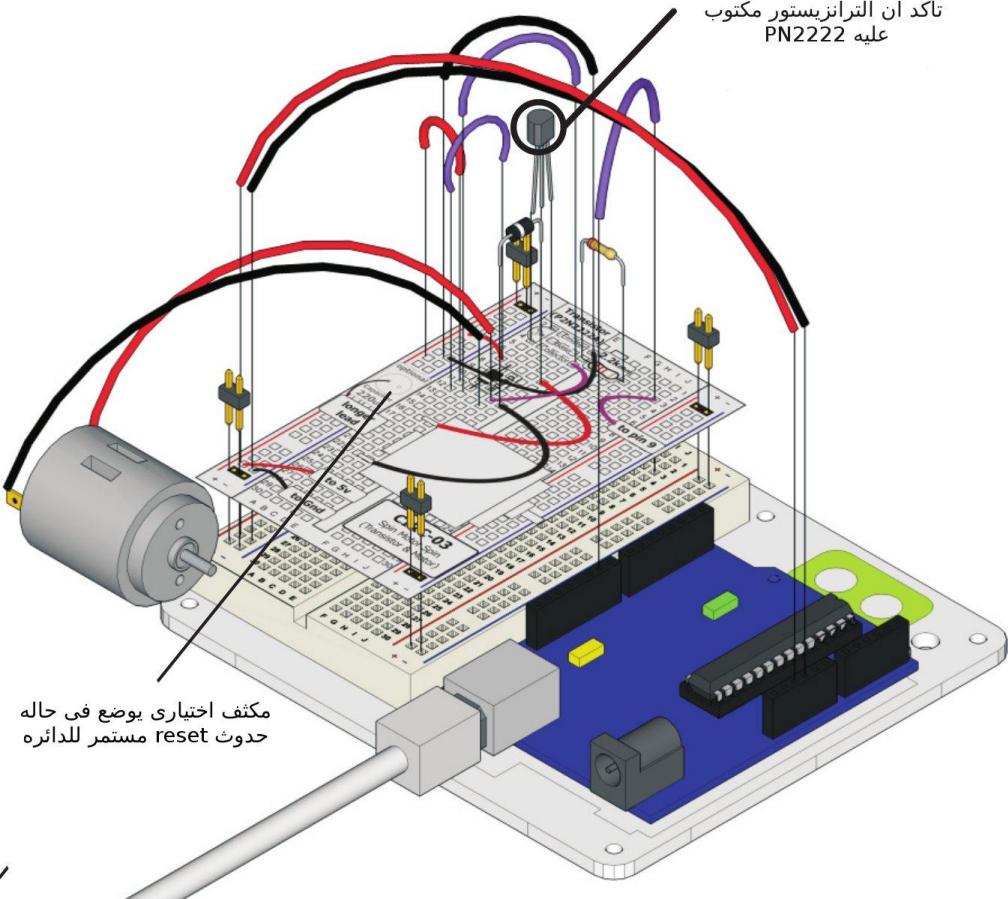
**Schematic****مكونات المثال (7):**

- ✓ بورده اردوینو Uno
- ✓ لوحة تجارب Breadboard
- ✓ محرك تيار مستمر DC motor
- ✓ ترانزستور PN2222 و 2N2222
- ✓ دايد 1N4001 او اي بديل
- ✓ مقاومه 2.2 كيلو اوم
- ✓ أسلاك توصيل
- ✓ كابل التوصيل بار USB
- ✓ ورق المساعدة المطبوعة



فى هذا المثال سوف نستخدم محرك تيار مستمر من الحجم الصغير والذى يوجد عادة فى لعب الأطفال ويعمل بفرق جهد يبدأ من 3 فولت واقصاه 9 فولت وستجده مثل هذه المحركات متوفره فى محلات المكونات الألكترونيه او فى اي من لعب اطفال القديمه اللى تحتوى على محركات ☺

تأكد ان الترانزستور مكتوب  
عليه PN2222



بعد الانتهاء من توصيل المكونات على لوح التجارب قم بكتابه الأكواد التالية ثم ارفعها  
إلى بورده اردوينو

```
//Example_10_DC_Motor
int motorPin = 9;
int onTime = 2500;
int offTime = 1000;

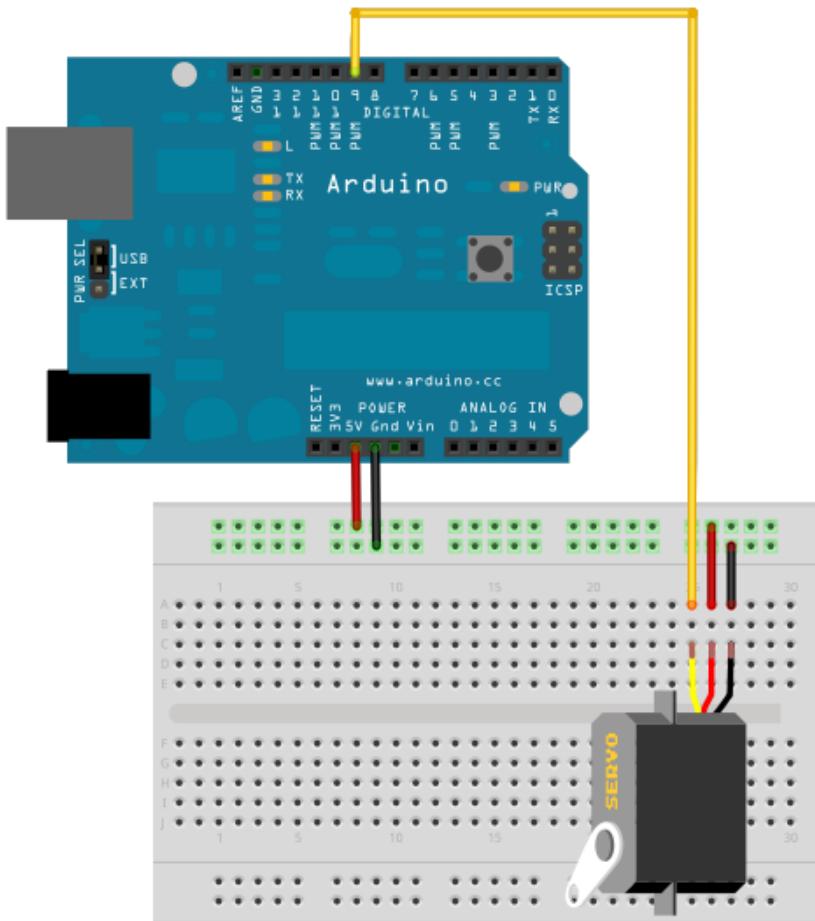
void setup ()
{pinMode(motorPin, OUTPUT); }

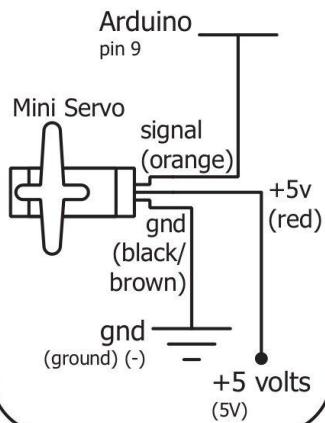
void loop ()
{
analogWrite(motorPin,100);
delay(onTime);
digitalWrite(motorPin, LOW);
delay(offTime);

analogWrite(motorPin,190);
delay(onTime);
digitalWrite(motorPin, LOW);
delay(offTime);

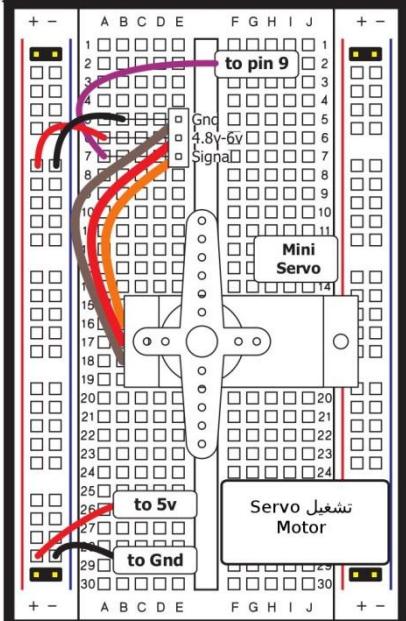
analogWrite(motorPin,255);
delay(onTime);
digitalWrite(motorPin, LOW);
delay(offTime);
}
```

## المثال الحادى عشر: استخدام محرك سيرفو

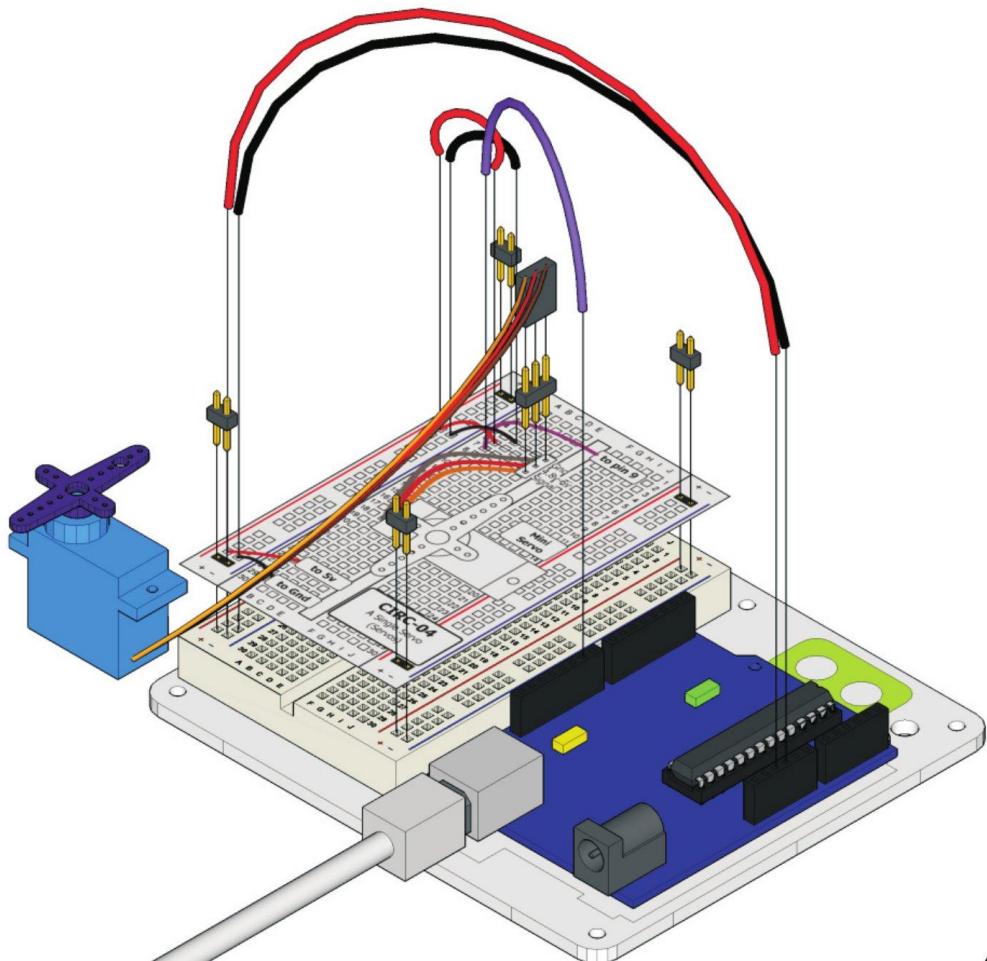


**Schematic****مكونات المثال (7):**

- ✓ بورده اردوینو Uno
- ✓ لوحة تجارب Breadboard
- ✓ محرك تيار مستمر DC motor
- ✓ ترانزستور PN2222 او 2N2222
- ✓ دايوه 1N4001 او اي بديل
- ✓ مقاومه 2.2 كيلو اوم
- ✓ أسلاك توصيل
- ✓ كابل التوصيل بار USB
- ✓ ورق المساعدة المطبوعة



فى هذا المثال سوف نستخدم محرك تيار مستمر من الحجم الصغير والذى يوجد عادة فى لعب الأطفال ويعمل بفرق جهد يبدأ من 3 فولت واقصاه 9 فولت وستجده مثل هذه المحركات متوفرة فى محلات المكونات الألكترونیه او فى اي من لعب اطفال القديمه التي تحتوى على محركات ☺



بعد الانتهاء من توصيل المكونات على لوح التجارب قم بكتابه الأكواود التالية ثم ارفعها إلى بورده اردوينو (ملحوظه: يمكنك ان تجد الكود البرمجي في قائمه الأمثله الجاهزه فى برنامج اردوينو (examples - servo - sweep

مكتبة اضافية

```
//Example_11_Servo_Motor  
#include <Servo.h>  
Servo myservo;  
int pos = 0;
```

```
void setup()  
{  
    myservo.attach(9);  
}
```

```
void loop()  
{  
  
    for(pos = 0; pos < 180; pos += 1)  
    {  
        myservo.write(pos);  
        delay(15);  
    }  
    for(pos = 180; pos>=1; pos-=1)  
    {  
        myservo.write(pos);  
        delay(15);  
    }  
}
```

## ملاحظات شخصيه :

هذه الصفحة مخصصة لكتابه ملاحظاتك الشخصيه عن الفصل السادس :

اكتب ملاحظاتك  
 هنا

## مُلْكُ الْحَفْرَالِ الْعَظِيمِ :

الجبال قد تصمد امام الزلازل لكنها لن تقاوم قطرات الماء التي تهطل بانتظام في هدوء وتكرار ، فأجعل عملك مثل قطرات المياه حتى تتغلب على جميع العوائق

حَلْمُهُ صِنْبَرِيُّهُ فَرِيمُهُ

# الفصل السابع

وسائل الادخال و الاصدار امتطورة

Advanced Inputs/Outputs



**فى الفصل** السابقه اعتمدنا بشكل اساسي على ابسط وسائل الادخال والاخراج للتعامل مع اردوينو مثل السويتش(ادخال Input) والدايود الضوئي (اخراج Output)



فى هذا الفصل سنستعرض بعض المكونات الاكثر تطورا للتفاعل مع المتحكمات الدقيقه فى مشاريع واقعيه يمكن استخدامها فى حياتنا اليوميه، سيتطرق هذا الفصل الى شرح المكونات التالية:

- شاشات عرض الكريستال السائل LCD liquid crystal Display:
- لوحة ادخال الارقام Keypad
- مصفوفه الديايد الضوئي Led matrix
- Relays المتممات



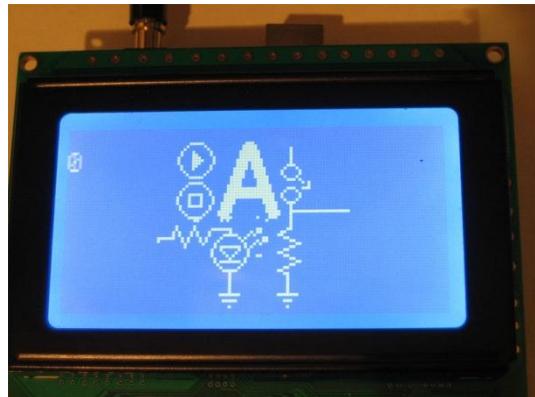
شاشات عرض الكريستال السائل : تتكون هذا الشاشات من زجاج الكريستال المعالج و توفر هذه الشاشات بأحجام و أنواع مختلفة و سوف نستعرض منها نوعين اساسيين و هما:

- شاشات العرض المعتمدة على الحروف Character LCD
- شاشات العرض المعتمدة على الرسومات Graphical LCD

Character LCD



Graphical LCD



توفر شاشات عرض المعتمدة على الحروف Character LCD امكانية اخراج اي نصوص تتكون من حروف او ارقام او رموز (مثل التي تكتب على لوحة المفاتيح في الحاسوب الآلى) و توفر بأحجام مختلفة والوان مختلفة مثل:

Green 16x2 lcd

Blue 16x2 lcd

Greens 20x4 lcd

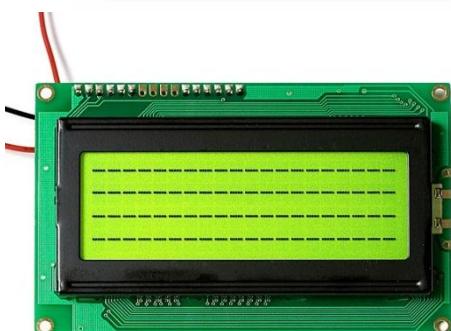
و يمثل الرقم 16x2 عدد السطور (2) الحروف التي يمكن كتابتها في كل سطر (16) حرف كما توفر بالوان مختلفة كما في الصور التالية:



Green 16x2 LCD



Blue 16x2 LCD

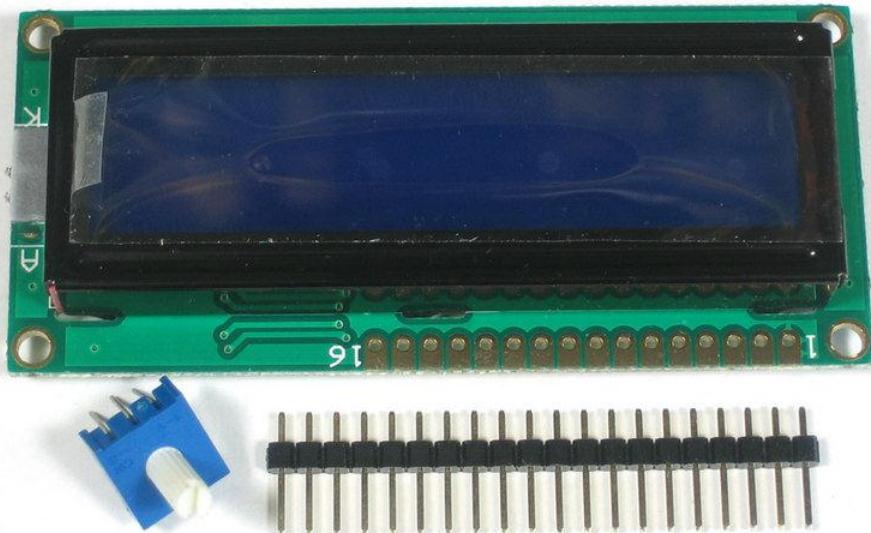


Green 20x4 LCD



Black 16x2 LCD

# المثال العاشر: توصيل شاشه Character LCD بمقاس 16x2

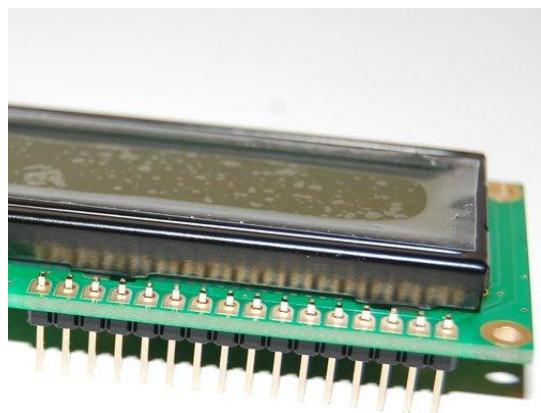
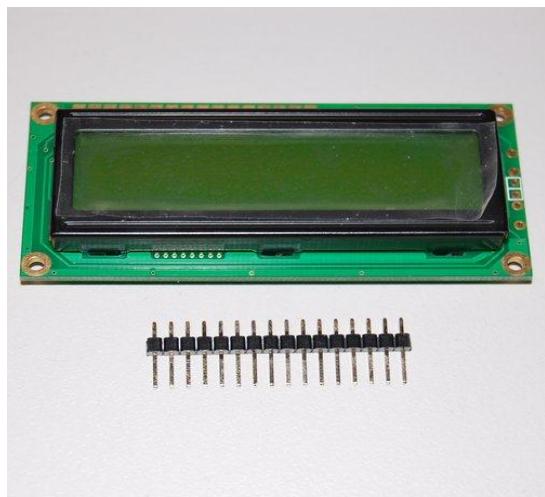


## سنحتاج المكونات التالية:

- شاشه بقياس 2x16 لون اضافه ازرق (او اي لون تفضله انت )
- اعمده توصيل نحاسيه قصيره 16 نقطه copper Pin Headers 16 point
- مقاومه متغيره بقيمه 10 كيلو او姆
- مکواه لحام
- قصدیر لحام

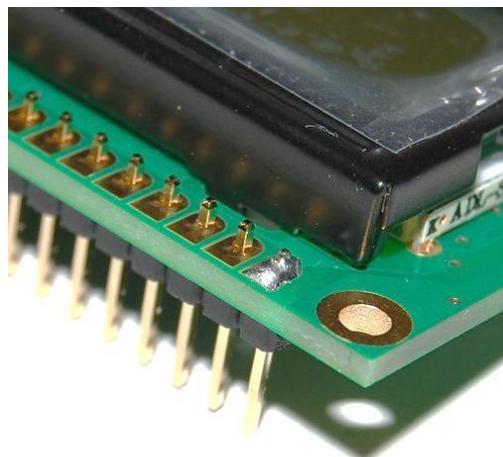
الخطوة الأولى هي لحام نقاط التوصيل Pin Headers بنقاط لحام الشاشة و يتم استخدام مكواه اللحام والقصدير في هذا العملية:

اذا اردت تعلم لحام المكونات الالكترونية توجهه الى اللينك التالي  
<http://www.aaroncake.net/electronics/solder.htm>



قم بلحام اول نقطه توصيل باستخدام الكاويه والقصدير وانتظر 5 ثوانى حتى تبرد نقطه التوصيل

تنبيه: لا تحاول لمس نقطه اللحام بيدهك بعد اللحام مباشره فقد يؤدى ذلك الى حدوث حروق للجلد من حرارة اللحام



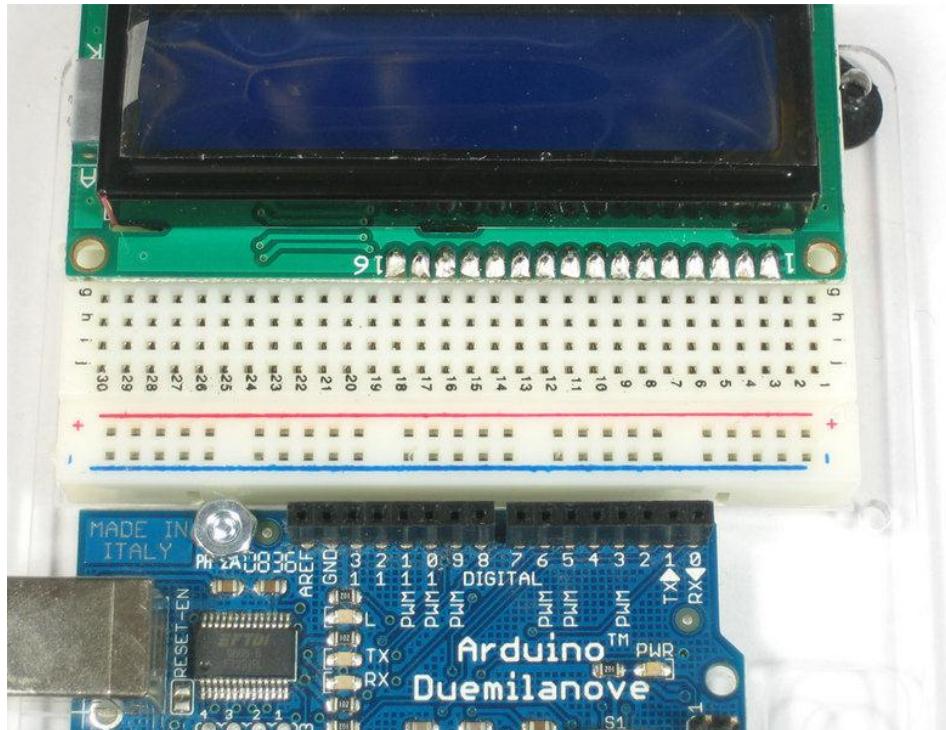
بعد ذلك قم بلحام نقطه التوصيل الاخيرة وذلك حتى تقوم بتثبيت pin headers وذلك من كلا الجانبيين



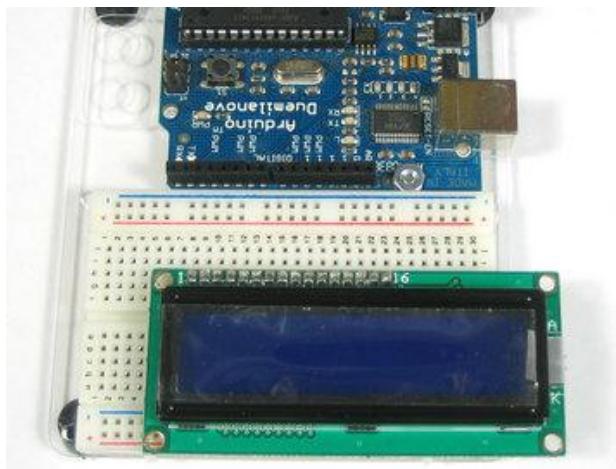
قم بتكرار نفس العملية السابقة مع جميع نقاط التوصيل حتى تنتهي من 16 نقطه كما في الصورة التاليه



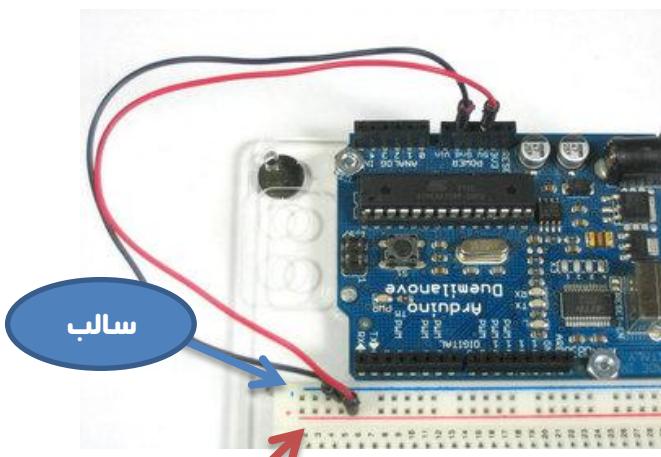
بعد الانتهاء من اللحام قم بتركيب pin header على لوحة التجارب كما في الصورة



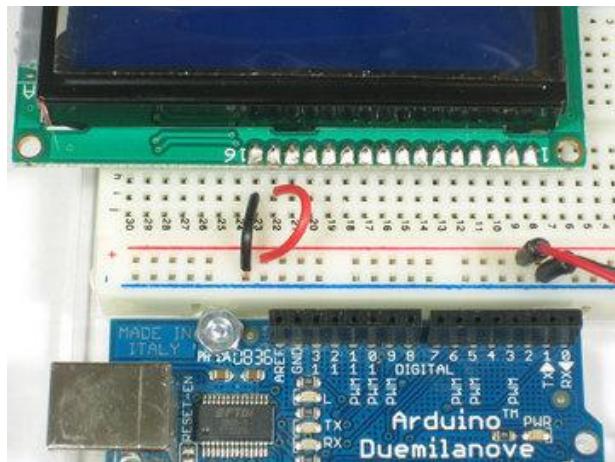
يفضل استخدام قاعده تثبيت بلاستيكية تستخدم فى تثبيت اردوينو ولوحة التجارب معا  
المثل التى تستخدم فى الصوره التاليه



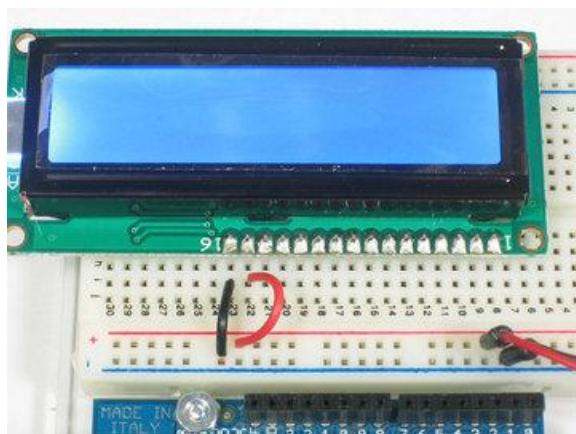
قاعدہ التثیت  
البلاستیکیہ تحمل  
اردوینو و لوحة  
التجارب و الشاشہ  
معا



- الان نقوم بتوصيل المخرج 5v على خط الموجب الاحمر والطرف GND على الخط الارضي فى لوحة التجارب



- بعد ذلك نقوم بتوصيل المدخل رقم 16 فى شاشه العرض على الخط الارضى و نوصل المدخل رقم 15 على الخط الموجب 5 كما في الصوره التاليه :

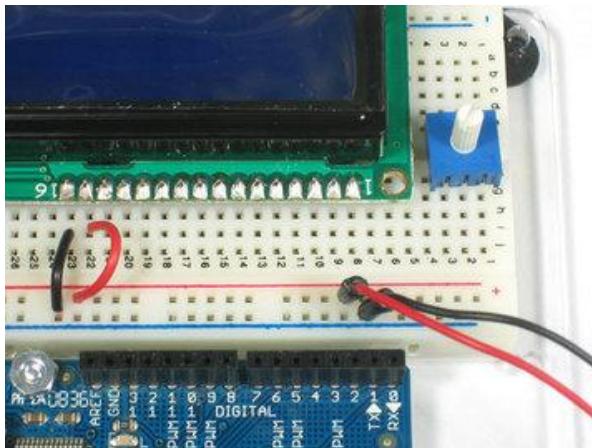


- قم بتوصيل اردوينو بالبطاريه او كابل الـ USB ثم لاحظ ان الضوء الخلفي للشاشة بدأ في العمل كما في الصوره التاليه :

ملحوظه يختلف لون الاضاءه على حسب الشاشه المستخدمه و توفر الالوان التاليه:

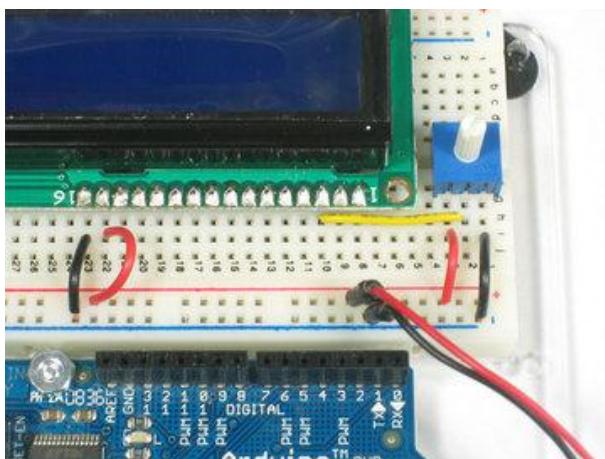
**الاحمر - الأخضر - الأزرق - الأبيض**

## تركيب اطلاعه اطغيمه للتحكم فى شدة الاضاءه

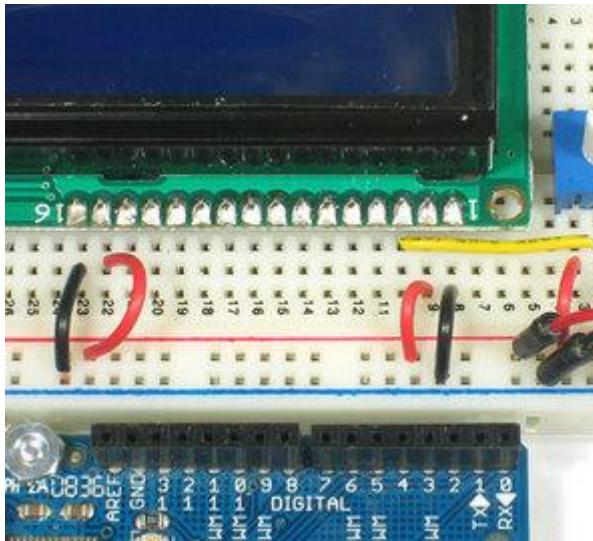


تعتبر تلك الخطوه اختياريه و يمكنك ان تهملها اذا اردت.

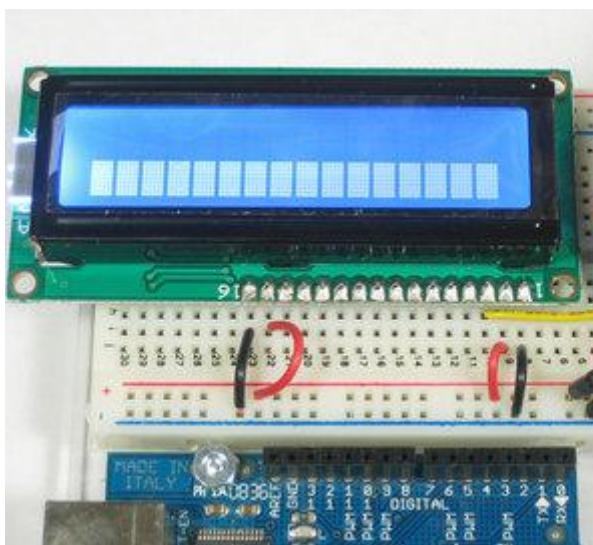
الهدف من تركيب المقاومه المتغيره هو التحكم فى التيار الكهربئي الداخل الى الششه وبالثالى التحكم فى شده السطوع (الاضاءه) الناتجه



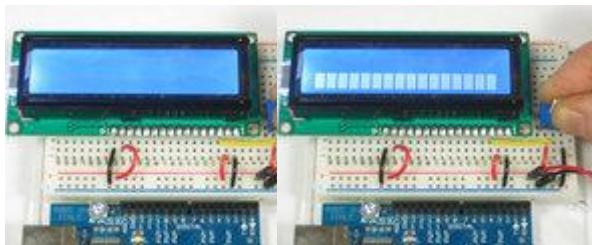
- قم بتوصيل احد الاطراف الجانبيه للمقاومه المتغيره بالخط الموجب والطرف الجانبي الاخر بالخط السالب ثم قم بتوصيل الطرف الاوسط للمقاومه المتغيره بالمدخل رقم 3 على الششه كما في الصوره التاليه :



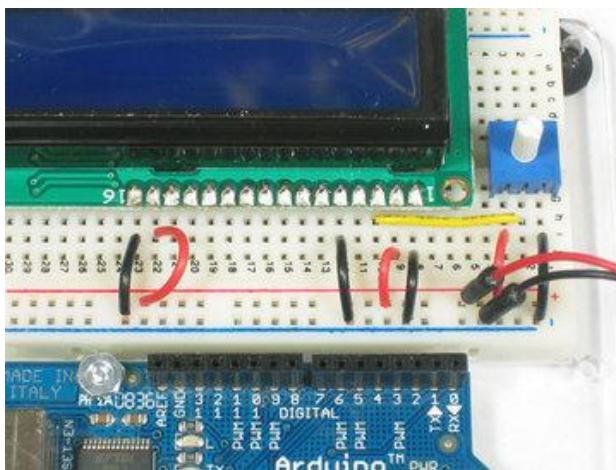
- قم بتوصيل المدخل رقم 1 على شاشه العرض بالخط الارض والمدخل رقم 2 بالخط الموجب على لوحة التجارب



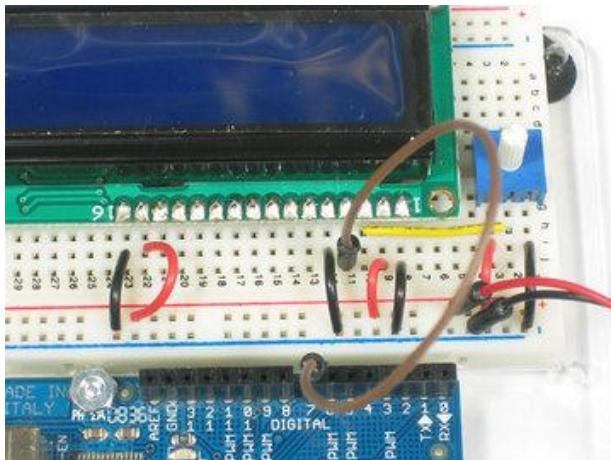
الان قم بوصيل البطاريه مره اخرى وقم بتدوير عصا الدوران فى المقاومه المتغيره ولاحظ ماذا سيحدث للاضاءه الخلفيه لشاشه العرض



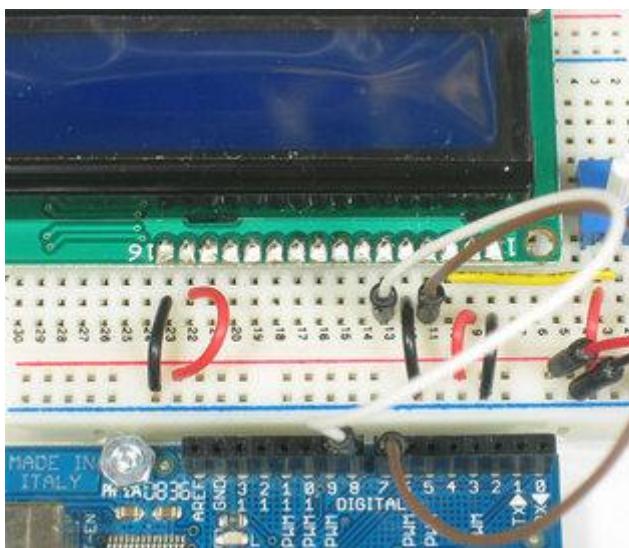
جميع الخطوات السابقة كانت تهدف لتوسيع الشاشه بالطاقة و بالمقاومة الضوئية التي تحكم في مدى سطوع الشاشه و شده الاضائه الخلفيه ، الخطوات التاليه ستوضح كيف توصل منافذ نقل البيانات من اردوينو الى الشاشه.



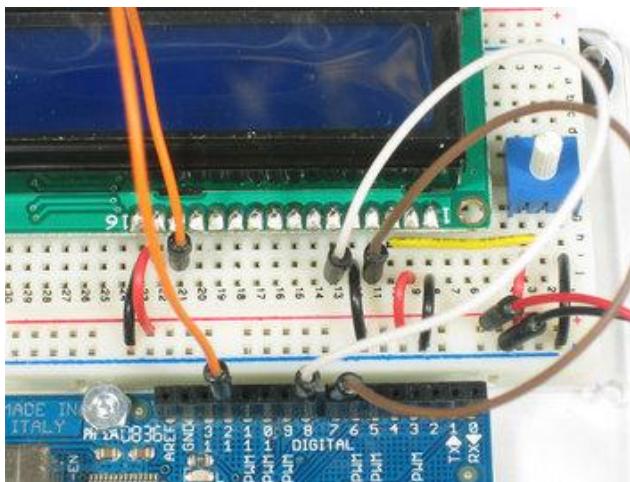
في بعض المشاريع قد تستخدم المدخل رقم 5 والذي يسمى RW لكن في حالتنا هذا سنقوم بتوصيله بالخط الأرضي



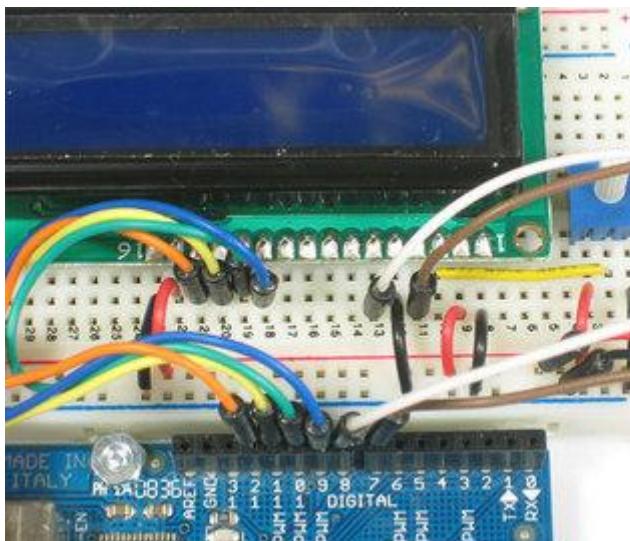
قم بتوصيل المدخل رقم 4 في شاشه العرض على المخرج رقم 7 في بورده اردوينو



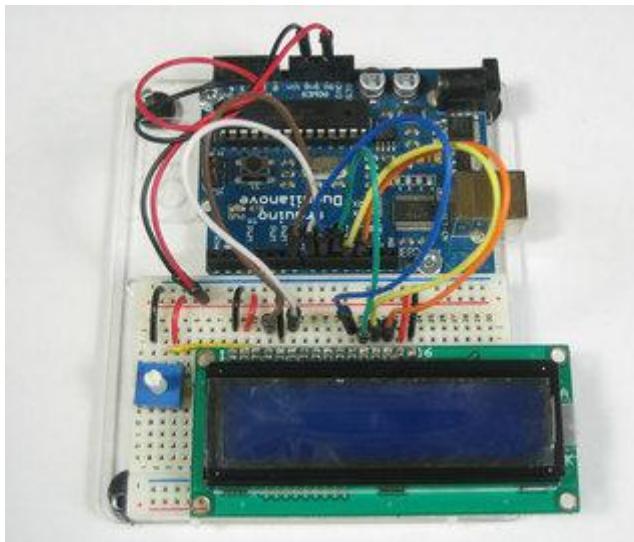
بعد ذلك نقوم بتوصيل المدخل رقم 6 في شاشه العرض على المخرج رقم 8 في بورده اردوينو



قم بتوصيل المدخل رقم 14 في شاشه العرض على المخرج رقم 12 في بورده اردوينو



الخطوه الاخيره هي توصيل المدخل رقم 13,12,11 بالمخرج رقم 10,10,10 الموجوده على اردوينو بنفس الترتيب كما في الصوره التاليه



الشكل النهائي بعد  
توصيل جميع الاطراف  
اللازمه لتشغيل الشاشه  
مع اردوينو.

### الآن حان وقت كتابه الكود البرمجى

توفر بيئه تطوير اردوينو العديد من الامثله الجاهزه و يمكنك ان تستخدمن مثال الششه  
الجاهز من خلال فتح قائمه الامثله كالتالى:

**File→Examples→LiquidCrystal→HelloWorld**

سنحتاج ان نعدل السطر البرمجى التالى :

```
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

ليصبح هكذا:

```
LiquidCrystal lcd(7, 8, 9, 10, 11, 12);
```

بعد تعديل السطر قم بتاكيد و رفع الكود الى بورده اردوينو كما فعلت فى الامثله السابقه

الكود البرمجي بعد التعديل:

```
//Example_12_LCD_16x2

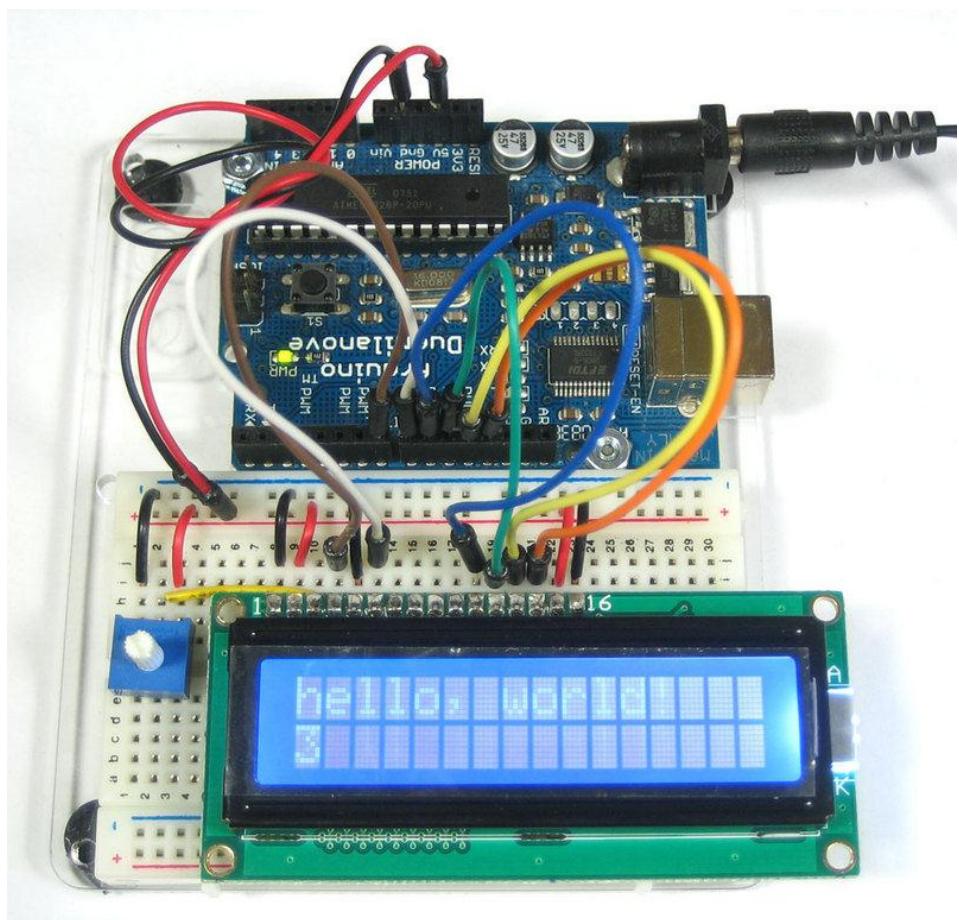
#include <LiquidCrystal.h>

LiquidCrystal lcd(7, 8, 9, 10, 11, 12);

void setup()
{
    lcd.begin(16, 2);
    lcd.print("hello, world!");
}

void loop()
{
    lcd.setCursor(0, 1);
    lcd.print(millis()/1000);
}
```

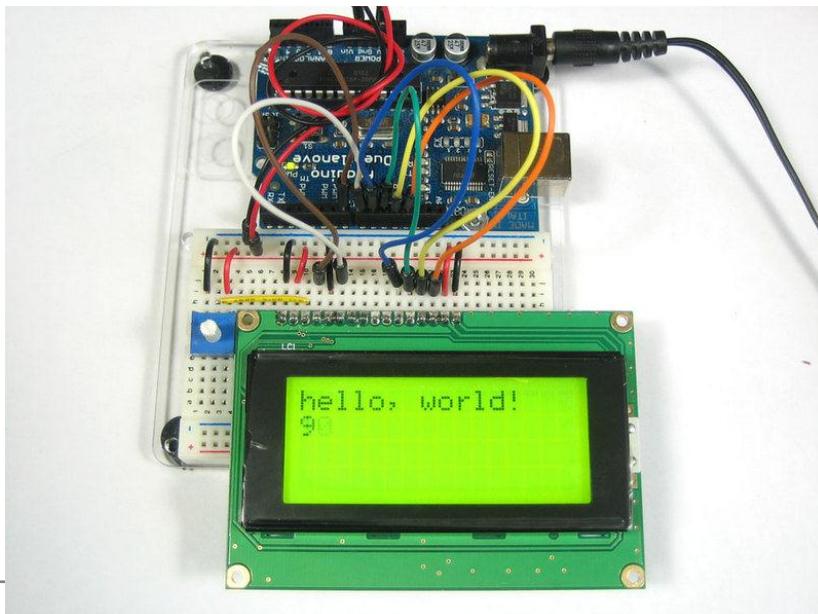
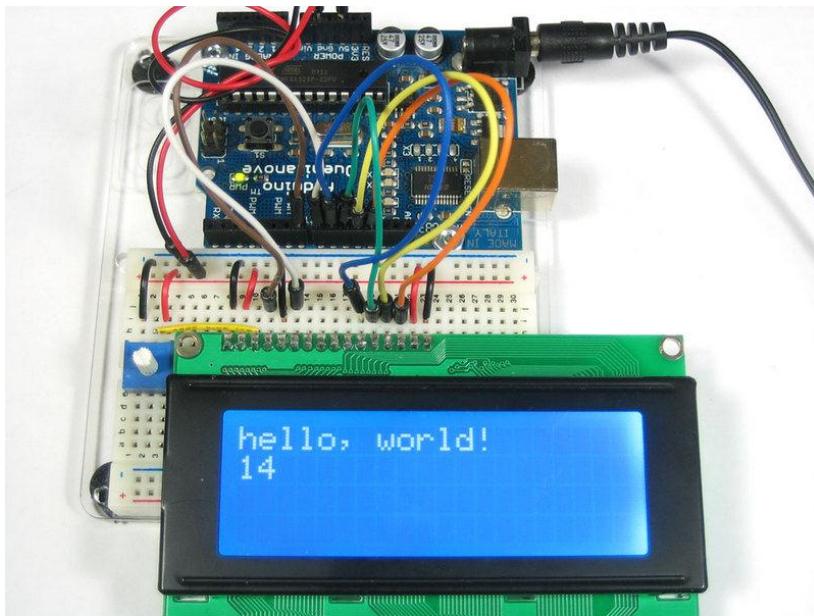
## الشكل النهائي للمثال العاشر (^\_^)

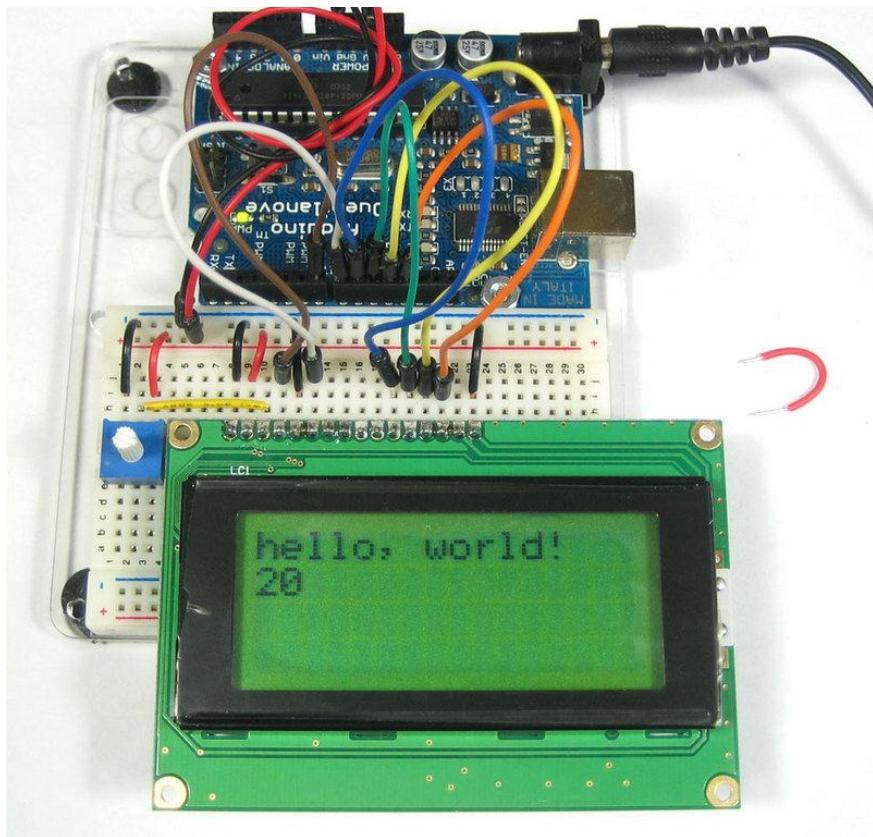


تستطيع تعديل مستوى سطوع الشاشه عن طريق تغير قيمه المقاومه المتغيره



كما يمكنك استخدام اي حجم ولون تفضله من الشاشات Character LCD مثل  
الحجم الاكبر فى عدد الحروف 20x4 كما فى الصور التالية





### مصادر اضافية للمعلومات:

لمزيد من المعلومات والأوامر المستخدمة مع شاشات LCD

<http://arduino.cc/en/Tutorial/LiquidCrystal>

عمل اشكال مرسومه على ششه من نوع character LCD

<http://www.instructables.com/id/Controlling-a-character-LCD-with-an-Arduino>

لتوسيع شاشه هواتف نوكيا الملونه Nokia 6100 او 2600

<http://www.instructables.com/id/How-To-Use-a-Nokia-Color-LCD/>

## استخدام لوحة الأرقام Keypad مع اردوينو



تعتبر لوحة الأرقام من أهم وسائل الادخال المستخدمة بكثرة في مشاريع المتحكمات الدقيقة والتي ستجدها حولك في العديد من الأجهزة الإلكترونية مثل الهاتف المنزلي ولوحة المفاتيح ولوحة التحكم في المصاعد الموجودة في البيوت متعددة الطوابق ... الخ

وتحتفل أشكال لوحات الأرقام تبعاً لحجمها وعدد الأرقام المتاحة ، في بعض اللوحات ستجد رموز إضافية مثل علامه النجمة (\*) وعلامة الشباك (#) أو حتى بعض الحروف الانجليزية مثل A,D,F وتعتبر لوحة الأرقام بحجم 4x3 و 4x4 هي أشهر لوحات الكتابة



3x4



4x4



4x4



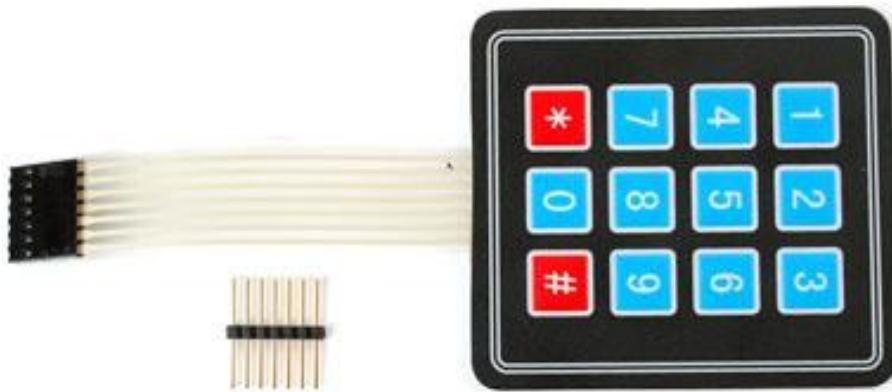
كما توفر بعض اللوحات المرنه والقابله للطى بسهوله والتى تصنع من نوع خاص من البلاستيك المرن و تميز بالنحافه حتى تبلغ ثخانه اللوحه بضعه مليمترات فقط و تميز بالسعر المنخفض (حوالى 2 دولار فقط )



سوف نشرح لوحه الارقام المرنه بمقاس 3x4 والتي تميز بالمواصفات التالية:

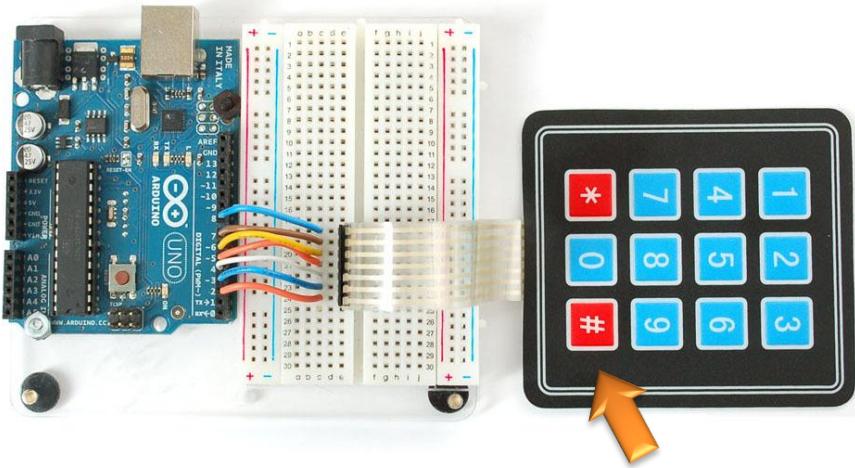
- Weight: 7.5 grams
- Keypad dimensions: 70mm x 77mm x 1mm (2.75" x 3" x 0.035")
- Length of cable + connector: 85mm
- 7-pin 0.1" pitch connector

## المثال الحادى عشر أستخدام لوحة ارقام



### مكونات المثال الحادى عشر:

- لوحة ارقام (مونه) بمقاس 3x4
- أعمدهه توصيل نحاسيه 7 نقاط (7) Pin-Headers
- اردوينو Uno
- Breadboard



قم بتوصيل مخارج اردوينو بدئاً من المخرج رقم 2 حتى المخرج رقم 8 بلوحة الأرقام مع ملاحظة ان يكون رمز # هو الرمز المتواجد امام المخرج رقم 2 في بورده اردوينو كما في الصورة بالضبط و بذلك تكون قد انتهينا من تجهيز لوحة الأرقام.

قبل ان نبدأ كتابه الكود البرمجي سيتوجب تحميل مكتبه لوحه الارقام من موقع اردوينو حيث لا تتوفر هذه المكتبه بشكل افتراضي في برنامج Arduino IDE على عكس مكتبه استخدام الشاشات من نوع Character LCD والتى تتوفر بشكل افتراضي داخل برنامج Arduino IDE

أولاً: توجهه الى اللينك التالي على موقع اردوينو الرسمي:  
<http://www.arduino.cc/playground/Code/Keypad>

ثم قم بتحميل مكتبه لوحة الأرقام كما في الشكل التالي:

The screenshot shows a web browser window with the URL <http://www.arduino.cc/playground/Code/Keypad>. The page content discusses the Keypad library version 3.0, which supports multi-keypresses by default. It also notes that the library is fully backwards compatible. Below this, it explains that external resistors or diodes are not needed because the library uses internal pullup resistors and ensures high-impedance pins for unused columns. A section on I2C port expanders is also mentioned.

## Download, install and import

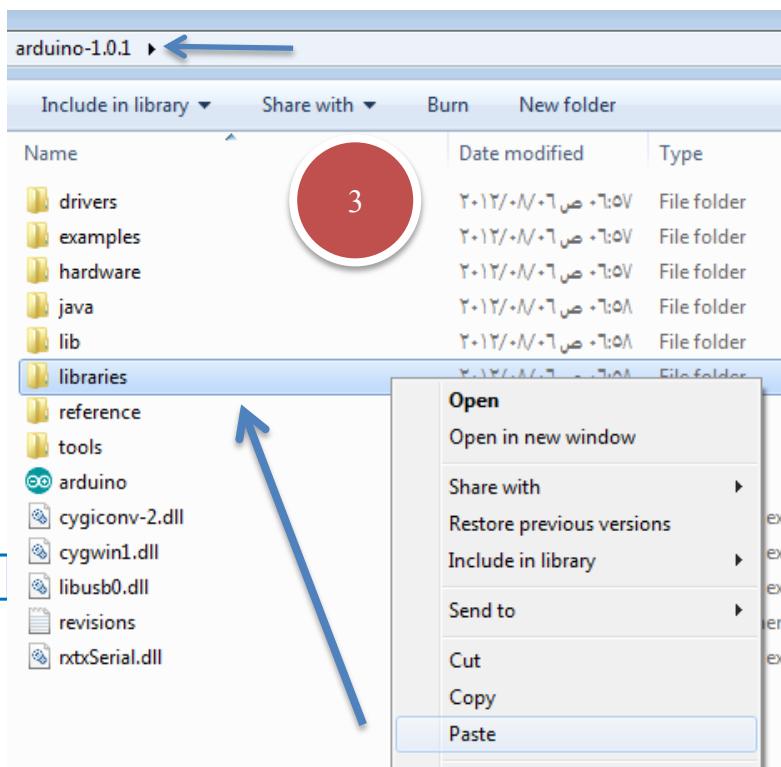
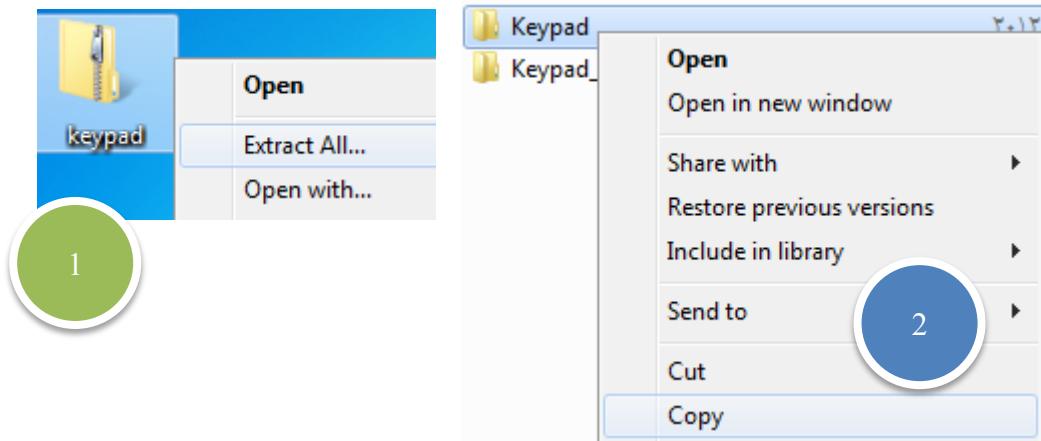
Download here: [keypad.zip](#)

Put the Keypad folder in "arduino\libraries\".

In the Arduino IDE, create a new sketch (or open one) and select from the menu bar "Sketch -> Import Library -> Keypad".

Once the library is imported, an "#include <Keypad.h>" line will appear at the top of your Sketch.

بعدها سيببدأ المتصفح بتنزيل مكتبه لوحة الأرقام و هي عباره عن ملف مضغوط اسمه keypad.zip ، الخطوه التاليه هي ان تقوم بفك ضغط الملف و نسخ الفولدر المسمى keypad و توجهه الى مكان تواجد برنامج اردوينو **Arduino IDE** حيث ستتجد فولدر اسمه libraries قم بفتح الفولدر و الصق keypad داخله كما في الصوره التاليه:



بعد الانتهاء من تجهيز المكتبة البرمجية نبدأ في كتابة الكود:

```
//Example_13_Keypad_Input
#include <Keypad.h>
const byte ROWS = 4;           ← عدد الصفوف
const byte COLS = 3;           ← الحروف و الرموز
char keys[ROWS][COLS] = {      ←
{'1','2','3'},                ←
{'4','5','6'},                ←
{'7','8','9'},                ←
{'#','0','*'}}                ←
```

};

byte rowPins[ROWS] = {5, 4, 3, 2};  
byte colPins[COLS] = {8, 7, 6};

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

```
void setup()
{
  Serial.begin(9600);
}
```

```
void loop()
{
  char key = keypad.getKey();

  if (key != NO_KEY) { Serial.println(key); }
}
```

عدد الصفوف

الحروف و الرموز

أمر يقوم بتفعيل  
استقبال الأرقام من  
لوحة الأرقام

بعد الانتهاء من كتابة الكود قم بتشغيل serial monitor واضغط على الأرقام في  
لوحة Keypad وانظر ماذا سيحدث !!

## مصادر اضافيه للمعلومات:

مكتبه لوحه الأرقام والأوامر البرمجيه :

<http://www.arduino.cc/playground/Code/Keypad>

كيف تصنع قفل الكترونى بأسستخدام اردوينو و السيرفو موتور:

<http://www.instructables.com/id/Access-control-with-Arduino-Keypad-4x4-Servo>

كيف توصل لوحه الأرقام بأردوينو عن طريق 3 أسلاك فقط بدلا من 7 أسلاك:

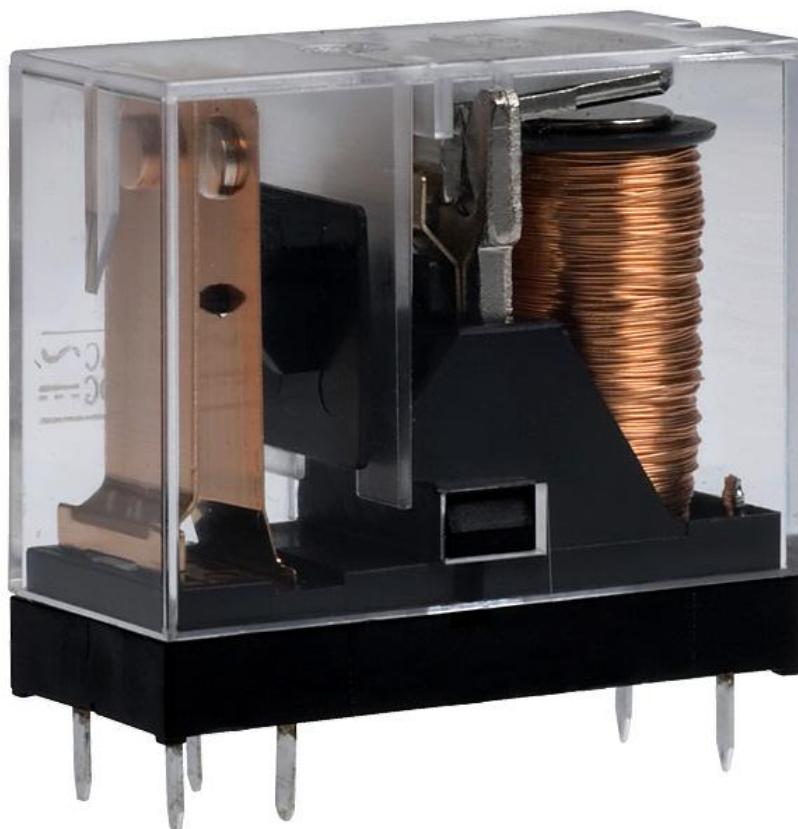
<http://www.instructables.com/id/Arduino-3-wire-Matrix-Keypad>

عمل نظام امنى (مفتاح الكترونى شامل) مكون من اردوينو ميجا و شاشه ولوحه ارقام و

قفل الكترونى (سيرفو موتور):

<http://www.instructables.com/id/Password-Lock-with-Arduino>

## استخدام المُرْجِل مع اردوينو Relay



يعتبر الريلاي من اهم العناصر المستخدمة في التحكم الالكتروني بأنواعه المختلفة

## ما هو ذلك العنصر؟ وكيف يمكن استخدامه؟ وما هي تطبيقاته؟ إيجابياته؟ سلبياته؟

مم يتكون؟

الريلاي هو عنصر ميكانيكي/إلكتروني، ويمكننا تخيله على شكل مفتاح أو زر كهربائي، داخلياً يتكون من جزئين رئيسيين:

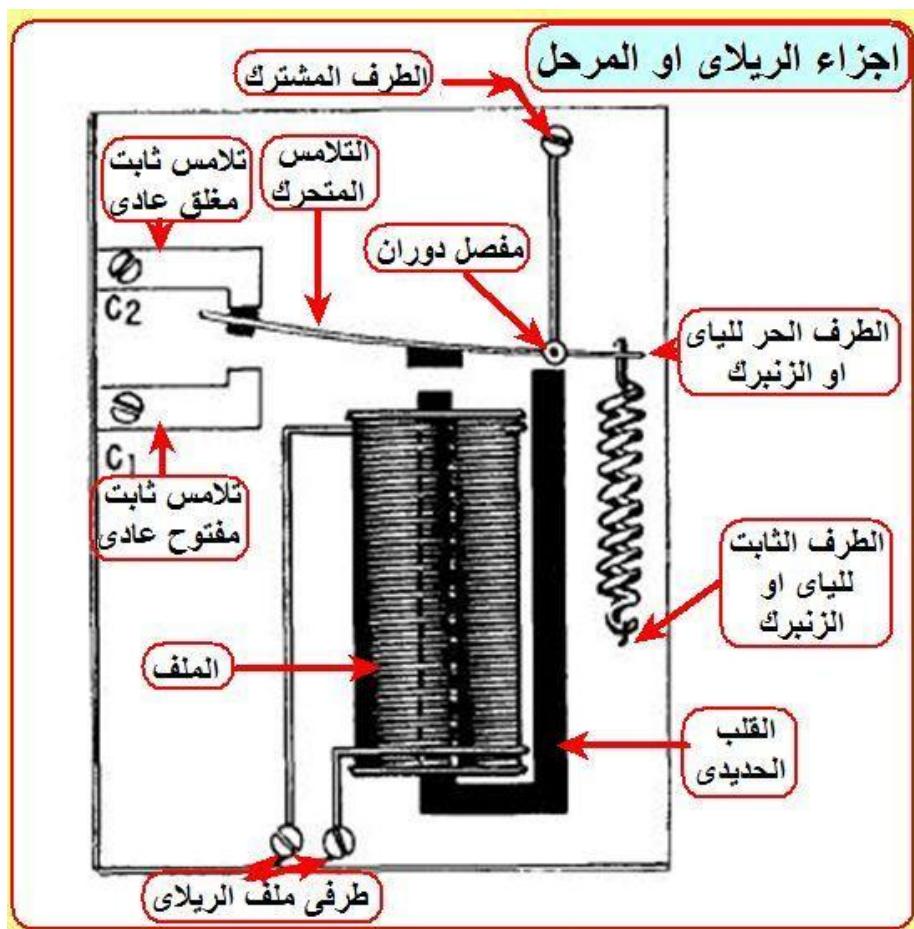
**الأول:** سلك ملفوف حول قالب حديدي، وفي مقدمة ذلك القالب يتوضع **الجزء الثاني**: وهو لسان أو قطعة معدنية مستطيلة الشكل تقع أمام الملف وهي بمثابة المفتاح فهي تحتوي على تفاصيل معدنية يتم من خلالها توصيل حمل كهربائي ليعمل الريلاي على فصله أو تشغيله بحسب وضع الملف في الريلاي.

## ما هو الرمز الإلكتروني للريلاي في الدارات الإلكترونية؟

يرمز للريلاي بالشكل التالي: حيث الجزء الأيسر هو الملف والأيمن هو رمز المفتاح الكهربائي



## التركيب من الداخل



**كيف يعمل؟**

عندما يتم تطبيق فرق جهد على طرفي الملف الداخلي للريلاي، سيمرن تيار في الملف ليتحول بواسطته الملف لمغناطيس كهربائي مولدا مجالاً مغناطيسياً سيقوم بدوره بجذب

ذلك اللسان أو القطعة المواجهة للملف بحيث تغلق التماسات الكهربائية، وعند فصل الجهد المطبق سيتلاشى النيار تدريجيا ليختفي ذلك المجال المغناطيسي، وهناك زنبرك سيقوم بإعادة اللسان لوضعه الطبيعي وفصل التماسات وفتح المفتاح الذي أغلق.

إذا بكل بساطة فالريلالي هو مفتاح كهربائي ميكانيكي، ما إن يتم تطبيق جهد على ملفه، يبدأ بالعمل ليجذب تماسات معدنية ستعمل على غلق أو فتح دارة خارجية متصلة بها.

## وما فائدته إذا؟ ألا يمكننا استخدام مفتاح كهربائي مباشر؟

تخيل معي أننا نحتاج ان نوصل اردوينو بحمل كهربائي بعمل على فرق جهد متعدد قيمته 110 فولت او 220 فولت. منطقيا، لا يمكننا تشغيل ذلك الحمل من خلال اردوينو لأن أقصى جهد يمكن توليده من اردوينو هو 5 فولت فقط ،



وهنا يأتي دور الريلالي فهو سيقوم بعملية تشغيل لأي حمل ذا استهلاك كبير للطاقة من خلال فرق جهد صغير، والسر يمكن وراء تشغيل الملف بجهد صغير مثل **5 فولت** لنقوم بتشغيل المصباح الذي سيتم توصيله على أطراف التماسات التي سيجذبها الملف داخل الريلالي.

**إذا للريلالي** فائدة عظيمة كونه سيعمل على ترحيل جهد وتيار صغير لقيادة حمل كبير. وأيضا ميزة الريلالي تكمن في قدرته على العزل التام ما بين دارة المصدر والحمل “الخرج”， فهم معزولان تماما ولكل واحد منهمما نقطة أرضية مرجعية مختلفة عن الأخرى،

وهذا له فائدة كبيرة في منع انتقال التشو Ish والجهود العابرة لدارتنا وبمعنى أصح حمايتها. أخيراً، يمكن استخدام الريلاي في التحكم عن بعد، أي وضع الريلاي بالقرب من الجهاز ومد سلك للدراة التي تصدر الأوامر.

## عيوبه:

لكل عنصر إلكتروني فوائد وسلبيات ، وكذلك الأمر بالنسبة للريلاي، فأهم سلبياته هي:

- بما أنه يعتمد على الملف - وهو حسي - فلذلك ستتولد فيه قوة دافعة كهربائية عكسية عند مرور تيار به (تعرف بقاعدة لنز) عند التوصيل وعند الفصل مما قد يؤدي لعطب الدارة المتصلة به. إلا أنه يمكننا تجاوز هذه المشكلة البسيطة بواسطة وضع عنصر الدايمود بين طرفي الملف وذلك لمنع عودة تلك القوة الدافعة العكسية .
- حدوث ارتدادات ميكانيكية عند كل تحويل من وضع الفصل إلى وضع التوصيل أو العكس. مما قد يؤدي لعطب الحمل
- يحتاج إلى دائرة موائمة لكي يعمل جيدا مع الأنظمة الإلكترونية . وعادة تكون هذه الدارة هي مكونة من ترانزستور
- العمر الافتراضي للريلاي صغير نسبيا خصوصا في الدوائر التي تتطلب عددا كبيرا من مرات الوصول والفصل وعادة ما تذكر قيمة عدد مرات الفصل والوصل القصوى لكل ريلاي في ملف المواصفات الفنية

## دارة عملية لاستخدام الريلاي مع الدارت الإلكترونية

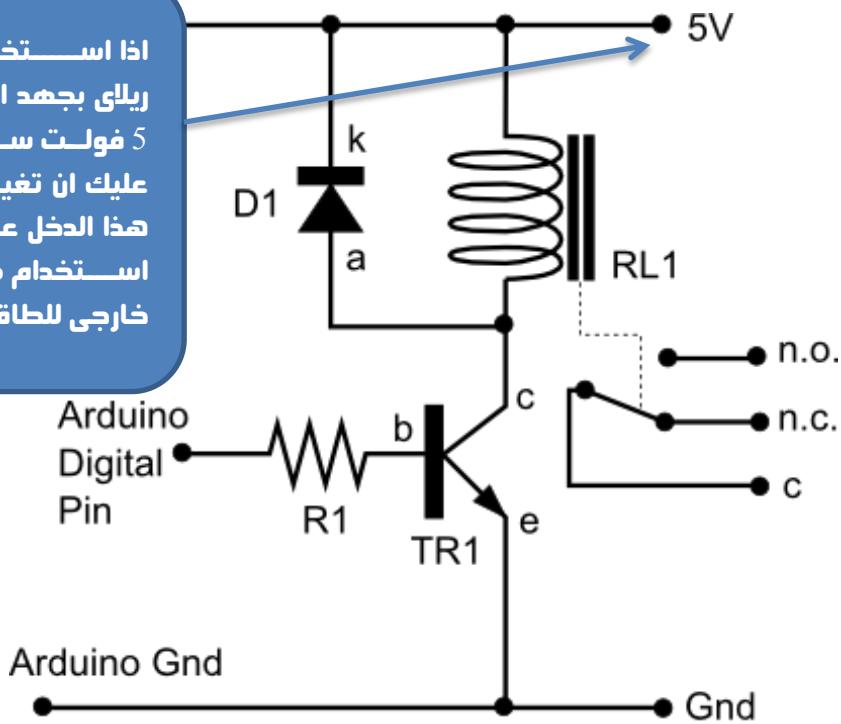
عند اختيار أي ريلاي فلا بد من الانتباه للأمور التالية كي نتمكن من الاختيار العلمي الصحيح ومعرفة البدائل التي يمكن استخدامها في حالة عدم توفر الريلاي المناسب

- فرق الجهد اللازم لتشغيل الملف الداخلي للريلاي: فمثلا لو كان لدينا دارة تعمل على فرق جهد 5 فولت فيجب اختيار ريلاي له فرق جهد لملفه هو 5 فولت، وفي حالة تغدر ذلك يمكننا استخدام جهد أكبر بعد استخدام دارة مواعمة سبق أن طرحتها في تدوينة الترانزستور.
- أقصى تيار لازم لتشغيل ملف الريلاي: وغالبا هذه القيمة لا تعطى بشكل واضح بل يتم ذكر مقاومة الملف بدلا منها، ومن خلال قسمة فرق جهد الملف على مقاومته يتم حساب التيار الأعظم. وهي قيمة مهمة جدا كي نعرف هل يمكننا تشغيل ووصل الريلاي بدارتنا مباشرة أو لابد من وضع دارة مواعمة جهد وتيار مرحلة الخرج أو التماسات، وهنا يجب معرفة ما يحتاجه الحمل ومقارنته بالريلاي وينبغي دائما أن يتم اختيار قيمة تيار عظمى تزيد عن القيمة المطلوبة بنسبة 5 إلى 10 %.
- العمر الافتراضي لتماسات الريلاي وعادة ما تعطى بملايين المرات

**ننتقل الآن للدارة العملية سنحتاج لهذه الدائرة المكونات التالية:**

- ريلاي يعمل على جهد 5 فولت (يمكن استخدام 9 او 12 فولت)
- دايو
- مقاومه 1 كيلو اوم
- ترانزستور من نوع 2N2222

اذا استخدمت اي  
ريل اي بجهد اعلى من  
5 فولت سيتوجب  
عليك ان تغير قيمة  
هذا الدخل عن طريق  
استخدام مطرز  
خارجي للطاقة



TR1 - BC548, 2N2222, 2N3704 transistor

D1 - 1N4004, 1N4007 diode

R1 - 1k  $\Omega$  resistor

RL1 - 5V relay

c - collector

b - base

e - emitter

n.c. - normally closed

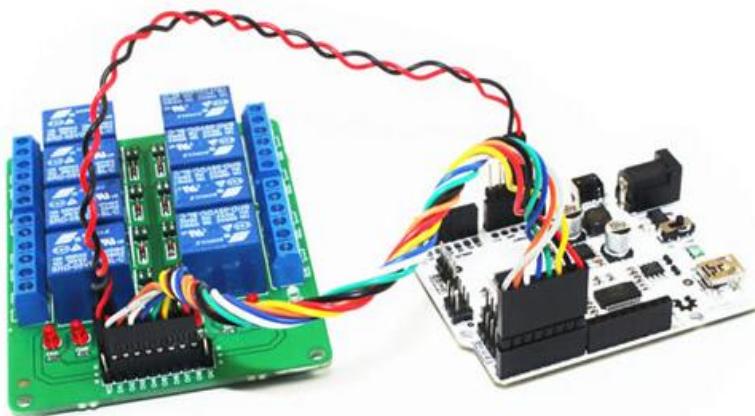
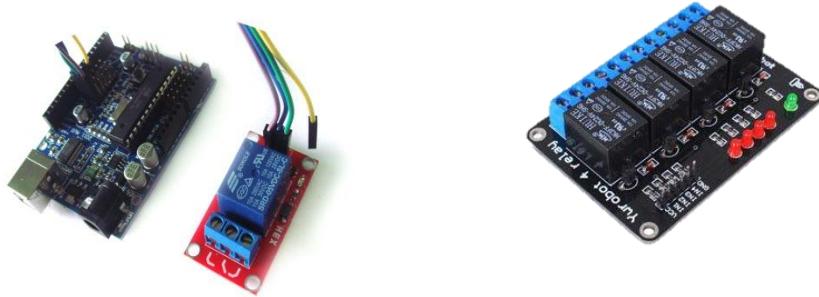
n.o. - normally open

c - common

a - anode

k - cathode

تتوفر في الأسواق العديد من دوائر المُرْحلات الجاهزة للأسستخدام Relay Boards مع اردوينو كما تتوفر بأحجام و أعداد مُرْحلات مختلفة مثل الصور التالية:



فى حال كان مشروعك يحتاج تشغيل احمال بفرق جهد كبير مثل 220 فولت فانى انصح بشده بأن تشتري بورده جاهزه بالحجم المناسب للمشروع الخاص بك و ذلك لأن هذه البوردات تصنع بجوده و دقه عاليه و هو ما تحتاجه فى المشاريع التي تعمل بجهد كبير تجنباً لحدوث اي اخطاء قد لا يُحمد عقباها.

### مصادر اضافيه للمعلومات:

هناك مشروع جميل يشرح كيفية استخدام الريالاي مع اردوينو لتجهيز منفذ للطاقة بجهد 220 فولت يعمل من خلال اشارات التحكم الصاده من اردوينو

<http://www.sparkfun.com/tutorials/119>

ملحوظه: مرجع الصور والمعلومات المذكوره عن المُرحل relay فى هذا الفصل

<http://muslimlead.com/?p=2121>

و تحتوى المدونه ايضا على العديد من الدروس الرائعه حول نظريات عمل العديد من المكونات الالكترونيه مثل الدايود, الثاني المشع للضوء, المكثف, الملف, الترازستور

يمكنك الضغط على اي اسم من اسماء العناصر المذكوره ليتم نقلك الى موضوع يشرح اساسيات العنصر بالتفصيل Ⓢ

## ملاحظات شخصيه :

هذه الصفحة مخصصة لكتابه ملاحظاتك الشخصيه عن الفصل السابع :

اكتب ملاحظاتك  
 هنا

## مُلْكُ الْقُوَّالِ الْعَظِيمِ :

سُبْحَانَكَ اللَّهُمَّ خَيْرَ مَعْلَمٍ  
عَلَمْتَ بِالْقَلْمِ الْقَرْوَنَ الْأُولَى

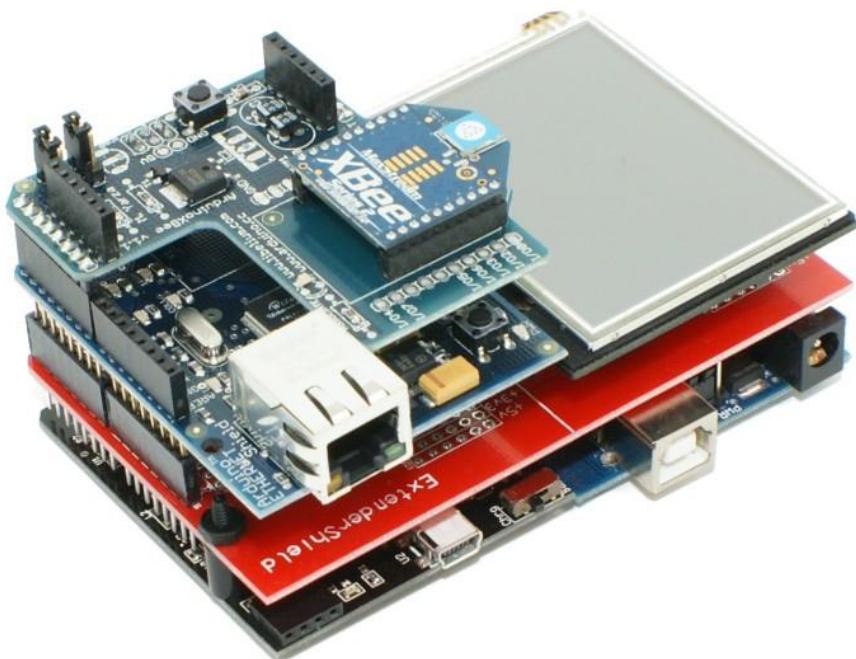
أَخْرَجْتَ هَذَا الْعَقْلَ مِنْ ظُلْمَاتِهِ  
وَهَدَيْتَهُ النُّورَ الْمُبِينَ سَبِيلًا

أمير الشعراء أحمد شوقي

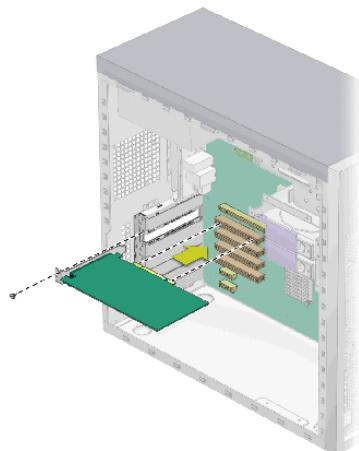
# الفصل الثامن

أغطية اردوينو الماجنة

Arduino Shields

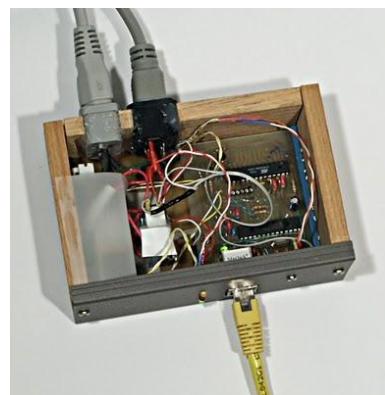


**من اكثـر المـعـيـزـات** المـشـيرـه في ارـدوـينـو هي توـافـر عـدـد كـبـير من الـاغـطـيهـ الـتي يـمـكـن تـركـيـبـها مـباـشـره عـلـى الـبـورـدـاتـ الـمـخـتـلـفـاتـ مـثـلـ Arduino uno وـ الإـصـدـارـهـ الـأـعـلـىـ فيـ Arduino Mega



تعـمل الـاغـطـيهـ بـنـفـسـ فـكـرهـ كـروـتـ PCIـ المـوجـودـهـ فـيـ الـحـاسـبـ الـآـلـيـ، عـلـىـ سـبـيلـ المـثالـ اـذـاـ لمـ توـافـرـ الـلـوـحـهـ الـأـمـ Motherboardـ كـارـتـ شبـكـهـ لـلـدـخـولـ عـلـىـ الـانـتـرـنـتـ فـأـنـكـ تـسـتـطـعـ بـكـلـ بـسـاطـهـ انـ تـشـرـىـ كـارـتـ شبـكـهـ network interface cardـ وـ قـوـهـ بـتـركـيـبـهـ وـ الدـخـولـ عـلـىـ الـانـتـرـنـتـ فـوـراـ .. هـكـذـاـ تعـملـ الـاغـطـيهـ اـرـدوـينـوـ

اـذـاـ كـانـ لـدـيـكـ خـبـرـهـ سـابـقـهـ مـعـ الـمـتـحـكـمـاتـ الـدـقـيقـهـ فـلـابـدـ اـنـكـ تـدـرـكـ مـدـىـ الصـعـوبـهـ وـ الـمـجـهـودـ الـمـبذـولـ حـتـىـ تـقـومـ بـتـوصـيـلـ الـمـتـحـكـمـ الـدـقـيقـ بـالـانـتـرـنـتـ اوـ شبـكـهـ محـليـهـ (ـحيـثـ سـتـقـومـ بـبـنـاءـ وـحدـهـ Ethernetـ مـنـ شبـكـهـ محـليـهـ)ـ وـ سـيـسـتـهـلـكـ ذـلـكـ الـكـثـيرـ مـنـ الـوقـتـ وـ الـصـفـرـ بـنـفـسـكـ)ـ وـ سـيـسـتـهـلـكـ ذـلـكـ الـكـثـيرـ مـنـ الـوقـتـ وـ الـمـجـهـودـ ،ـ اـمـاـ فـيـ حـالـهـ اـرـدوـينـوـ كـلـ مـاـ عـلـيـكـ فـعـلـهـ انـ تـشـرـىـ الغـطـاءـ جـاهـزـ وـ تـرـكـيـبـهـ مـباـشـرهـ عـلـىـ الـبـورـدـ الخـاصـهـ بـكـ وـ تـبـدـأـ فـيـ بـرـمـجـهـ مـشـروعـكـ بـكـلـ سـهـولـهـ



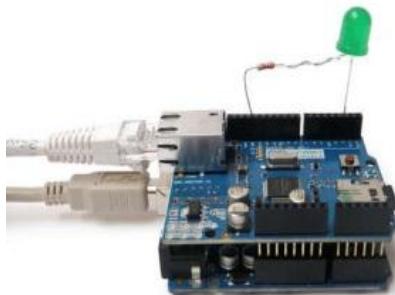
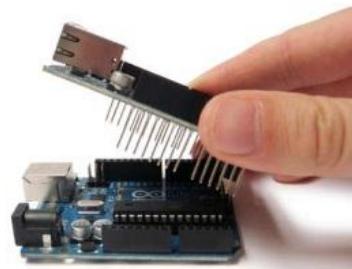
**تـقـوـفـ الـعـدـيدـ مـنـ الـاغـطـيهـ وـ سـنـسـتـعـرـضـ مـنـهـاـ التـالـيـ:**

## • غطاء اردوينو للشبكة السلكية Arduino Ethernet Shield •



**يوفّر** ذلك الغطاء امكانیه توصیل اردوینو بشبکات الحاسب الآلی و الانترنوت عن طريق توصیل کابل شبکه من نوع CAT5 مثل المستخدم فى الحاسب الآلی و يمكنك استغلال ذلك الغطاء فى عمل مشاريع التحكم عن بعد بأسستخدام الشبکه المحلیه او

الانترنوت فيمكنك مثلا توصیل عدد من الاجهزه الالكترونيه بالاردوینو والتحكم بها من اي مكان في العالم عن طريق الانترنوت ، وهناك بعض الاصدارات الحديثه من هذا الغطاء توفر ايضا امكانیه تركيب کارت ذاکره لتخزين الملفات و يمكن استغلال ذلك في

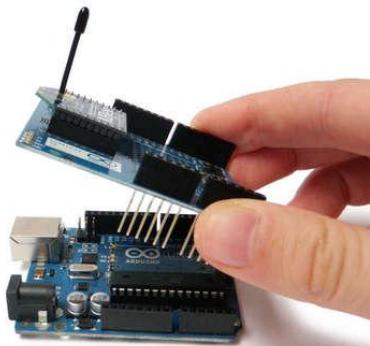


بعض المشاريع التي تحتاج تسجیل قیم بصوره دوریه مثل مشروع لتسجیل درجات الحرارة كل فتره زمنیه معینه و تخزين القيم على کرت ذاکه ثم ارسالها عن طريق الانترنوت الى صاحب المشروع

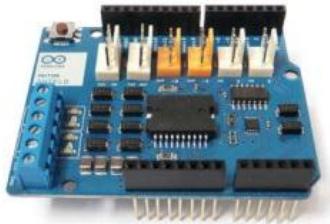
## • غطاء الشبكات اللاسلكية Xbee Shield •



يوفّر هذا الغطاء نفس الوظائف الموجودة في Wireless Ethernet Shield لكن بصورة لا سليكة حيث يمكنك من توصيل الأردوينو بـ شبـكة لـاسـلـكـيـه في نطاق يقارب 100 متر (يختلف النطاق بناء على قوه الاشاره اللاسلكيـه)

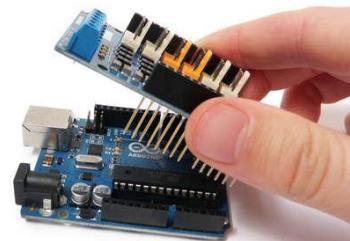


## • غطاء المحركات (الموتور) •

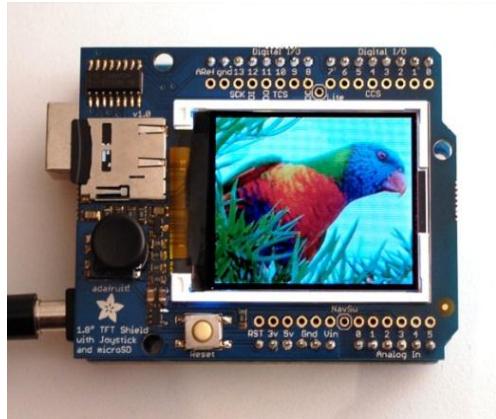


يستخدم هذا الغطاء في توصيل المحركات من نوع DC Motor, Servo Motors, 3 Stepper Motors حيث يمكنك توصيل 3 محركات في نفس الوقت (ملحوظة: بعض

إصدارات هذا الغطاء تسمح بتوصيل محركيين فقط) ويمكنك استغلال هذا الغطاء في المشاريع التي تحتاج محركات مثل مشاريع الإنسان الآلي Robots و ماكينات CNC



## • Arduino colored Touch screen



يستخدم ذلك الغطاء في توفير شاشه ملونه تفاعليه تعمل بالاستجابة للمسات الأصبع و يمكن استخدامها في المشاريع التي تحتاج إلى وسيليء لعرض صوره او بيانات معقده و التي لا تكفي الشاشات lcd ذات اللون الواحد لعرضها، و يتوفّر ذلك الغطاء بأحجام و مقاسات مختلفة تبدأ من مساحة 2 انش مربع إلى 4 انش مربع (الإنش هو وحده قياس تساوى 2.5 سنتيمتر)



هناك قائمه كبيره بأسماء و مواصفات أغطيه اردوينو المختلفة تجدها على الموقع التالي:

<http://shieldlist.org>

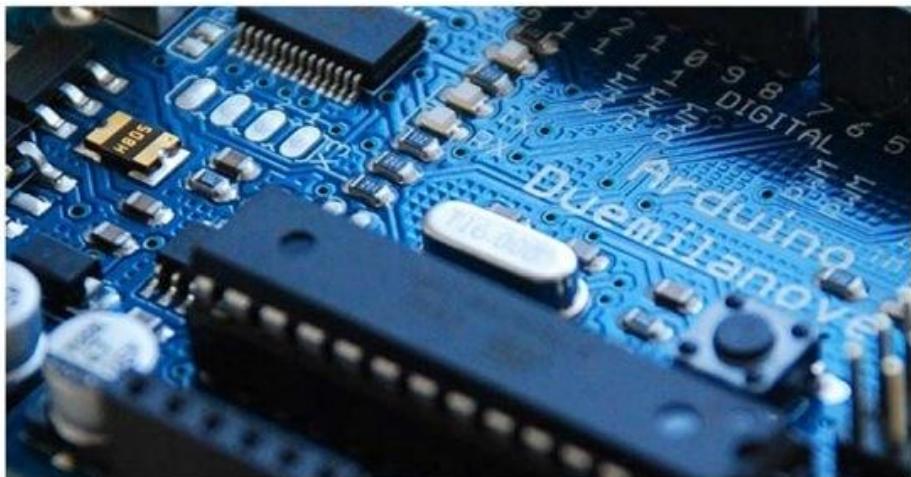
## ملاحظات شخصيه :

هذه الصفحة مخصصة لكتابه ملاحظاتك الشخصيه عن الفصل الثالث :

اكتب ملاحظاتك  
 هنا

# الفصل التاسع

اردوينو و تطور، ثورة اطتقحمات الدقيقة مفتوحة اطصد، من  
عائلة AVR الى ARM Cortex بـ FPGA



**تبدا** القصه فى عام 2005 فى مدينه ايفريا Ivrea الايطاليه عندما أجمتع فريق من مهندسي الألكترونيات ليناقشوا سُبل تعليم طُلاب الجامعه و المعاهد التقنيه علوم المتحكمات الدقيقه او ما يعرف باسم **Microcontrollers** ، كان الهدف من اللقاء ايجاد أسلوب سهل و غير مكلف لأدخال هذا النوع من العلوم فى الدراسه الجامعيه و يكون متاح للهواه فى ذات الوقت دون ان تكون هناك قيود أو تراخيص مكلفة يحتاجون لشرائها لاستخدام البرمجيات التى كانت تستخدم فى برمجه المتحكمات الدقيقه فى هذا الوقت .. من هنا تبدأ رحله نشاء اردوينو Arduino



قام كل من "ماسيمو بانزي" Massimo Banzi " بالتعاون مع " David كوارتيليس Cuartielles و جيانلوكا مارتينو Gianluca Martino " بأطلاق مشروع أردوين ايفريا Arduin of Ivrea و تمت تسميه المشروع باسم أشهر شخصيه تاريخيه فى المدينه و كان الهدف الأساسى للمشروع هو

عمل بيئه تطوير للمتحكمات دقيقه بصوره مفتوحه المصدر 100 فى المئه و تضمن هذا المشروع عمل بيئه تطوير برمجيه للمتحكمات الدقيقه Integrated Development Environment و تكون مجانيه فى ذات الوقت كما تضمن عمل لوحات تطوير Development Boards صغيره الحجم بتكلفه بسيطه يمكن للطلاب و الهواه التقنيين تحمل سعرها.

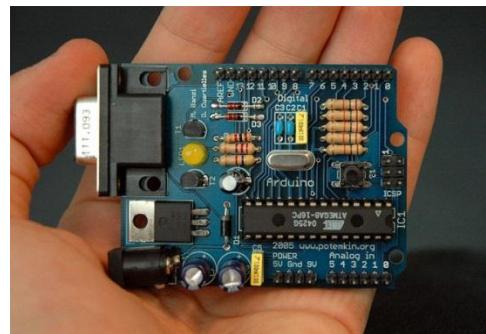
**بدأ تطوير المشروع اعتماداً على مشروع سابق اسمه Wiring Platform** والذى قد نجح فى تحقيق بعض أهداف مشروع اردوينو الأساسية مثل المصدر المفتوح

# Wiring

لكن لم يصل الى المستوى الذى تمناه فريق التطوير من الناحيه البرمجيه والتصميم العتادى Hardware Design لذلك تم البدأ فى العمل على تطوير اللغة البرمجيه المستخدمه فى مشروع Wiring وجعلها أسهل وأفضل كما تم دمج بعض التقنيات المستخدمه فى [مشروع Processing](#) وسميت هذه اللغة الجديده باسم [Arduino C](#)

هي مشابهه جداً للغه البرمجيه C++ من ناحيه طريقة كتابه الأوامر وتنسيق الأكواد.

تم إطلاق أول لوحة تطويريه لأردوينو في اواخر عام 2005 وأعتمدت على شريحة ATmega168 من العائله المشهوره AVR والتي تنتجه شركه Atmel الألكترونويه ، وسميت هذه اللوحة بأسم Arduino Serial V.1



بعد إطلاق اللوحة قام مؤسسى المشروع بنشر كل ما يتعلق بأردوينو بربخه مفتوحه المصدر على موقع [arduino.cc](http://arduino.cc) والذى كان يعني أن اي فرد في العالم يمكنه الاطلاع والتعديل على التصميمات الهندسيه والشفرات المصدرية Source Codes لكل من بوردات اردوينو المختلفة Arduino Boards والبرمجيات Arduino IDE وتم نشر كل هذا مجاناً.

و ما ان تم نشر المشروع حتى توالى الكثير من التطويرات على التصميمات الهندسية للوحات الألكترونية و بيئه التطوير من مهندسين و هواه مختلفين حول العالم و تم اصدار العديد من اللوحات الأحدث مثل:

### Arduino UNO , Arduino Mega, Arduino lilyPad, Arduino Duemilanove, Arduino Fio

و كان القاسم المشترك في كل الاصدارات السابقة هي انها تعتمد على شرائح المتحكمات الدقيقة من عائله AVR التي تنتجهما شركة Atmel و حقق المشروع نجاح باهر حتى وصل عدد اللوحات الألكترونية المباعة الى 300,000 قطعة في مختلف دول العالم.

**فى عام 2009** نعرض اردوينو لنقله نوعيه جديدده

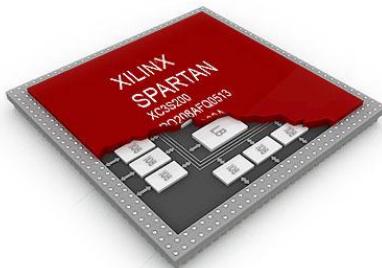
جائت على يد فريق LeafLabs والذى قام بتطوير لوحة اردوينو جديدده تسمى The Maple board تعمل بشرائح ألكترونية من عائله ARM Cortex والتى نقلت اردوينو الى افق جديد من السرعه والقوه حيث تتميز جميع شرائح ARM باحتواها على معالجات عاليه السرعه تمتلك قوه

معالجه تبدأ من 72 ميجا هرتز و تصل فى بعض الاصدارات الى 1.5 جيجا هرتز .



وبذلك أستطيع فريق LeafLab الخروج من حدود امكانيات شرائح AVR واستخدام الجيل الجديد من ARM Cortex مع الاحفاظ بلغه اردوينو البرمجيه التى تتميز بالسهوله و خلوها من التعقيدات و فى ذات الوقت الاحفاظ بعامل السعر حيث بلغ سعر Maple Board حوالي 34 دولار مع تكلفه الشحن.

**و بالرغم من كل التطورات السريعة في لوحات أردوينو المختلفة إلا ان التطوير لم يتوقف عن حدود ARM Cortex بل وصل إلى شرائح "مصفوفات البوابات المنطقية القابلة للبرمجة" Field-Programmable Gate Arrays (FPGA) في عام 2010 تم نشر مشروع مفتوح المصدر اسمه Papilio Arduino والهدف منه هو صناعة لوحة أردوينو تعمل بشرائح من نوع FPGA ويمكن برمجتها بلغة Arduino C**



**تتميز شرائح FPGA** بالسرعة الفائقة في اداء عمليات المعالجة بصورة متوازية وفك تشفير البيانات Decoding لذلك تستخدم في الكثير من التطبيقات الصناعية التي يكون فيها عامل الوقت امرا حاسماً في تصميم المنتج النهائي لكن يعيي هذه الشرائح عده أمور تجعلها مقتصرة على المحترفين والشركات فقط، الأول: تكلفة لوحات التطوير الخاصه بها والتى يبدأ سرعها غالبا من 100 دولار و اكثرا، ثانيا: صعوبه برمجه الشرائح حيث تستخدم لغات برمجيه معقده في تشغيل هذه الشرائح مثل Verilog ولغه VHDL.

جاء مشروع Papilio Arduino ليحل هذه المشكلات ويقدم لوحة تطوير جديدة الى عائله لوحات اردوينو و تم اصدار Papilio Arduino IDE وهي نسخه معدله من برنامج Arduino IDE لجعل لغه اردوينو



تناسب مع شرائح FPGA كما تم اصدار لوحات papilio boards للبيع من منتصف عام 2010 بأسعار تبدأ من 40 دولار فقط و يمكنك الاطلاع على المزيد من المعلومات عن لوحات papilio من الموقع التالي

<http://papilio.cc>

بعد مشروع Papilio Arduino أصبحت عائله اردوينو تحتوى على لوحات تطويرات تعمل تقريبا بكافة التقنيات التي تم ابتكارها مثل AVR و شرائح ARM Cortex حتى شرائح FPGA

وفي النهايه يمكننا القول أن مشروع اردوينو صنع أكبر مجتمع تقنى من المهندسين والفنين والهواه يعملون على تطوير الافكار والمشاريع المتعلقة بالتحكم الآلى بصورة تشاركيه و تفاعليه حول العالم اعتماداً على استخدام لوحات الكترونيه بتقنيات مختلفه ومع ذلك يتم برمجتها جمیعا بلغه برمجيه واحده و مجانيه و متاحه للجميع.

انه مجتمع جديد و مختلف .. حقاً أنها ثوره الكترونيه ....



## فاب لاب مصر او ما يعرف باسم ورشه التصنيع الشخصي

هو مكان مُصمم لتحويل الافكار من مجرد خيال الى واقع ملموس وبناء مجتمع من مصنعي الافكار والمنتجات من الهواه والمحترفين في مجالات مختلفه مثل: (الالكترونيات - الهندسه الميكانيكية - هندسه الحاسب - التصنيع الرقمي - الفنون)

ويوفر المكان العديد من الادوات التقنية التي تساعدك على تصميم افكارك مثل:

- ماكينة القطع بالليزر LASER cutter machine
- طابعه ثلاثيه الابعاد 3D printer
- ماكينة حفر بالتحكم الرقمي CNC machine
- بوردات اردوينو المختلفه و مكونات الكترونيه منوعه

والعديد من الادوات الاخرى ....

يهدف الفاب لاب الى توفير بيئه تشاركيه و تعليميه فى ذات الوقت تساعد الافراد على الابداع و اطلاق العنان لافكارهم و خيالهم و مشاركه هذه الافكار مع المجتمع.

كما يوفر وورش و دورات تدريبيه للافراد مثل:

- ورش تدريب عمليه لاستخدام اردوينو Arduino workshops
- ورش تدريبيه للاطفال لتعلم الالكترونيات Young Fab.Academy

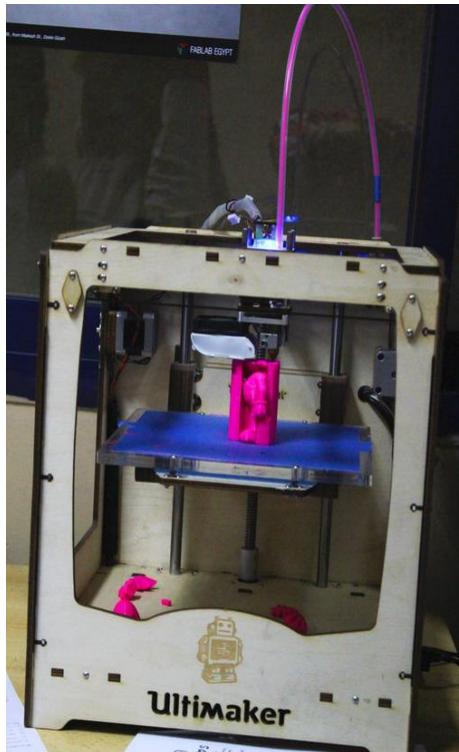
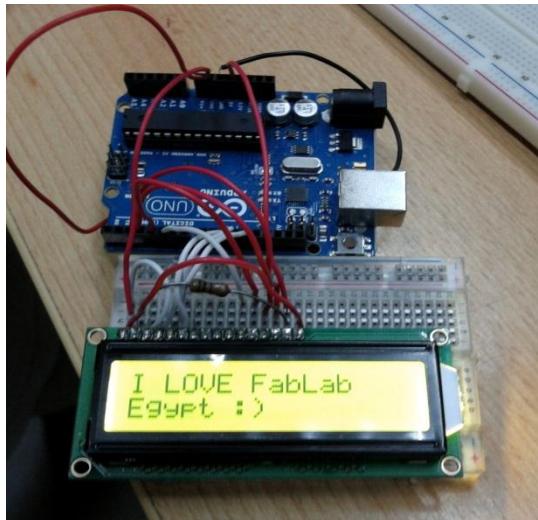


اذا كانت لديك فكره و تريد ان تنفذها و تحتاج الى بعض المساعده او تعاون من اشخاص اخرين او تحتاج الى دورات تدريبيه فى اردوينو فيمكنك زيارة فاب لاب مصر يوميا من الساعه 12 ظهرا حتى 8 مساءً جميع الايام

<http://fablab-egypt.com>

<https://www.facebook.com/fablab.egypt>





# مُلْحُقُ الْعِرَاجِعِ

**المصادر التعليمية التي اعتمدت عليها في كتابه  
محتوى الكتاب:**

- <http://www.instructables.com/id/Arduino-Projects>
- <http://www.oomlout.com>
- <http://ladyada.net>
- <http://bildr.org>
- <http://hlt.media.mit.edu/?cat=5>
- <http://circuit-projects.com/microcontroller>
- <http://www.eeweb.com>
- <http://www.lvl1.org>
- <http://www.coolcircuit.com>
- <http://www.instructables.com/id/Breadboard-How-To>
- <http://blog.makezine.com/2009/12/11/arduino-shields-open-source-hardware/>

## لمن يريد استخدام اردوينو مع الماتلاب MATLAB

- <http://www.mathworks.com/academia/arduino-software/arduino-matlab.html>
- <http://arduino.cc/playground/Interfacing/Matlab>

## مشاريع رائعة باستخدام اردوينو يمكنك ان تصنعها بنفسك :

- <http://fritzing.org/projects/>
- <http://www.instructables.com/id/Arduino-Projects>
- <http://www.ladyada.net/make/boarduino/index.html>
- <http://www.ladyada.net/make/mshield/index.html>
- <http://www.shapeoko.com/wiki/index.php/About>
- <http://www.ladyada.net/make/monochron/index.html>
- <http://www.ladyada.net/make/bedazzler/index.html>
- <http://www.ladyada.net/make/mintyboost/index.html>
- <http://dangerousprototypes.com/forum/viewtopic.php?f=56&t=2892#p28410>
- <http://www.ladyada.net/make/logshield/>
- <http://www.wayneandlayne.com/projects/video-game-shield/>
- <http://excamera.com/sphinx/gameduino/>
- <http://dangerousprototypes.com/2012/03/24/arduino-voice-control-with-easyvr-shield/>
- <http://blog.minibloq.org/p/documentation.html>
- <http://www.doctormonk.com/2011/09/arduino-solar-radio.html>
- <http://j4mie.org/blog/how-to-make-a-physical-gmail-notifier/>
- <http://blog.tinyenormous.com/2008/11/25/gmail-notifier-project-for-dummies/>
- [http://reprap.org/wiki/Arduino\\_Mega\\_Pololu\\_Shield](http://reprap.org/wiki/Arduino_Mega_Pololu_Shield)

# مراجع أخرى: كتب

- **30 Arduino Evil Genius projects**
- **Arduino.Internals**
- **Arduino Projects to Save the World**
- **Arduino robotics**
- **Beginning Android ADK with Arduino**
- **Environmental Monitoring Arduino**
- **Getting Started with Arduino (Second Edition)**
- **Shrinkify Your Arduino Projects -burn arduino to ATiny (Video)**
- **Make a mind-controlled arduino robot**
- **Make\_Magazine\_Volume\_25**
- **Making Things Talk Using Sensors (Second Edition)**
- **Making Things see hear and feel your world (Second Edition)**
- **O'Reilly - Arduino Cookbook**
- **O'Reilly-Make - Arduino Bots and Gadgets**
- **Oreilly Verlag - Arduino Physical Computing**
- **Practical Arduino Cool Projects For Open Source Hardware**
- **Programming Interactivity**
- **Programming Your Home Automate with Arduino, Android, and Your Computer**
- **The Art of Electronics 2nd edition Complete - Horowitz and Hill**