

## تعريف بلوحة Nucleo و تنصيب الأدوات والبرامج

### محتويات الجلسة:

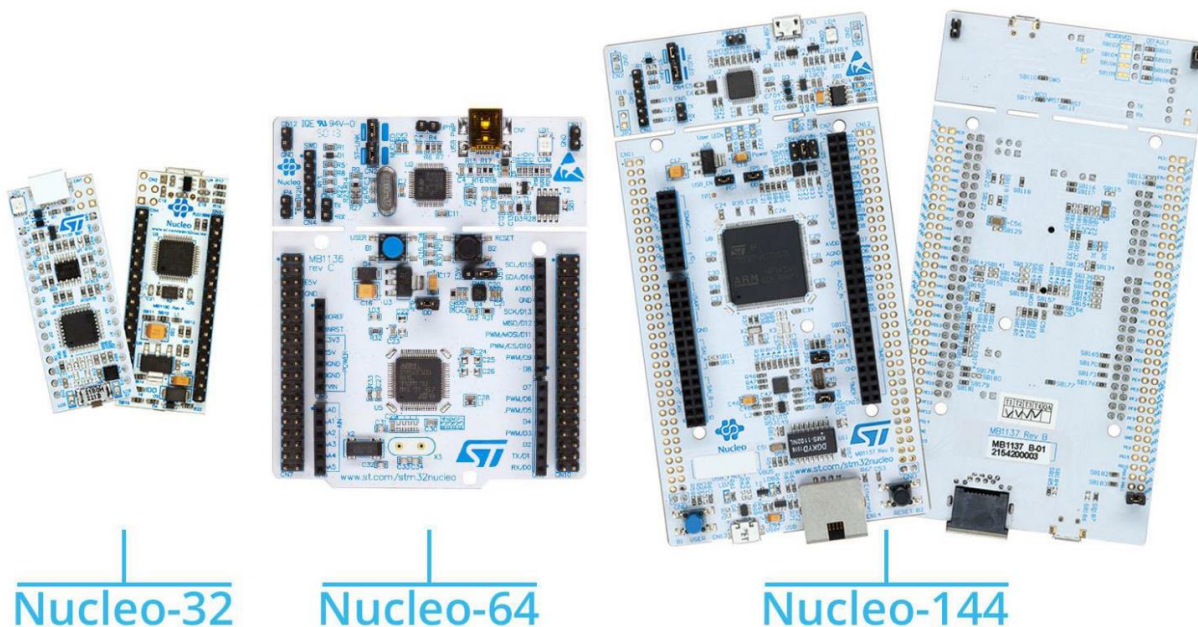
- 1- تعريف بلوحة Nucleo
- 2- تنصيب البرامج والأدوات
- 3- بناء أول تطبيق لإضاءة ليد باستخدام متحكمات stm32 ومكتبة hal

### الأدوات اللازمة للجلسة:

- لوحة Nucleo
- كبل Type-A to Mini-B

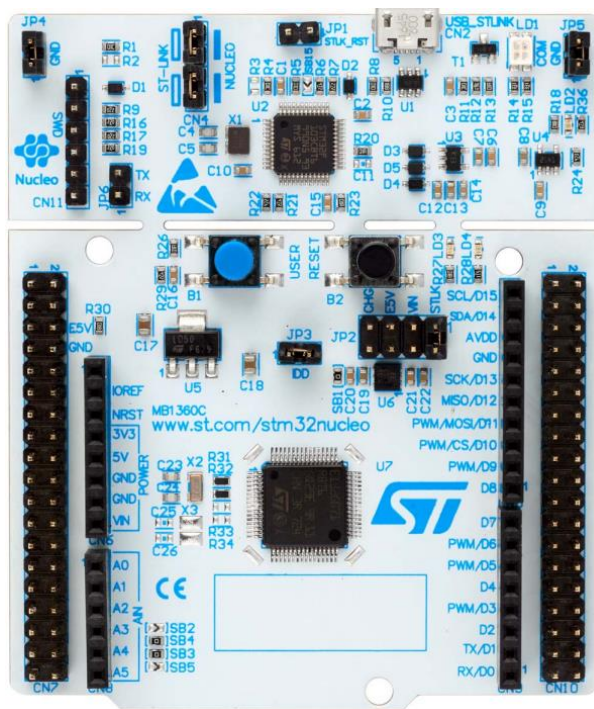
### 1. تعريف بلوحة Nucleo:

هناك ثلاث أشكال مختلفة من لوحة Nucleo كما هو موضح بالشكل التالي:



الشكل(1):الأشكال المختلفة من لوحة Nucleo

سنعتمد في هذا الكورس على لوحة Nucleo-64 (G071RB) وهي اللوحة الأقرب للوحة Arduino-UNO :

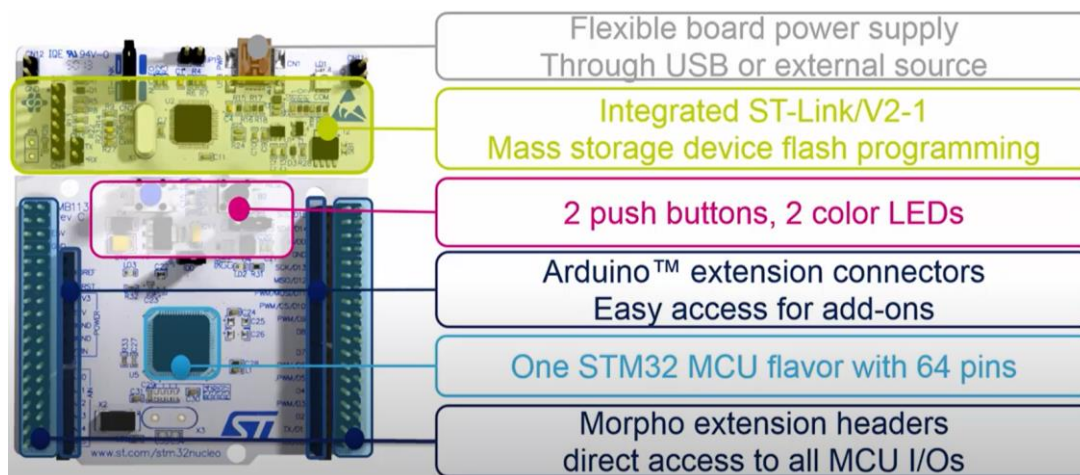


الشكل(2): لوحة Nucleo-64

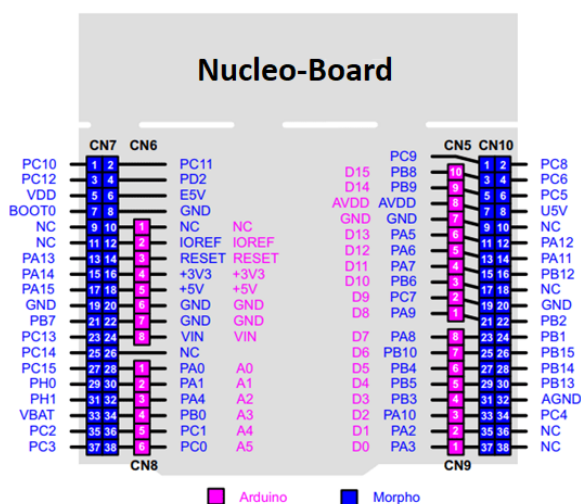
و لها المزايا التالي:

- تحتوي على متحكم stm32 ذو بنية ARM Cortex M4
- استهلاك التيار 2.4uA خلال وضع الاستعداد standby وبدون RTC
- مجال جهد العمل للمتحكم 1.7V to 3.6V
- سعة ذاكرة البرنامج flash memory هي 512KB
- سعة الذاكرة العشوائية 96KB
- تمتلك 50 قطب يمكن استخدامها كأقطاب إدخال/ إخراج عام GPIO
- تمتلك 16 قناة تشابهية بدقة 12bit
- تمتلك 6 مؤقتات بطول 16bit ومؤقتان بطول 32bit
- تمتلك مؤقتي مراقبة watchdog
- تمتلك 4 قنوات اتصال من نوع USART/UART Communication
- 3 قنوات اتصال من نوع I2C Communication
- 3 قنوات اتصال من نوع SPI Communication
- تردد العمل للمعالج 84MHZ
- مجال الهزاز الكريستالي الخارجي الممكن تركيبه 4 to 26 MHz
- تحتوي على هزاز كريستالي داخلي 16MHz
- تحتوي على هزاز كريستالي 32.768 kHz
- تحتوي على أطراف توصيل مشابهة تماماً لأطراف الـ ARDUINO-UNO تمكن المستخدم من استخدام كافة الـ shields المصممة للـ ARDUINO-UNO
- كما تحتوي على أطراف توصيل موصلة ST morpho تمكن المستخدم من الوصول إلى كافة أطراف المتحكم المصغر stm32

- يمكن تغذية اللوحة من خلال المبرمجة ST-LINK أو من خلال مصادر تغذية خارجية
  - تحتوي اللوحة على مبرمجة ST-LINK مدمجة تمكن المستخدم من رفع الكود للمتحكم المصغر وأيضاً فتح جلسة debug لتتبع الكود واكتشاف الأخطاء
  - تدعم العديد من البيئات البرمجية منها IAR Embedded Workbench®, MDK-ARM, Kiel and STM32CubeIDE
- بالإضافة إلى قابلية برمجتها من خلال بيئة Arduino IDE



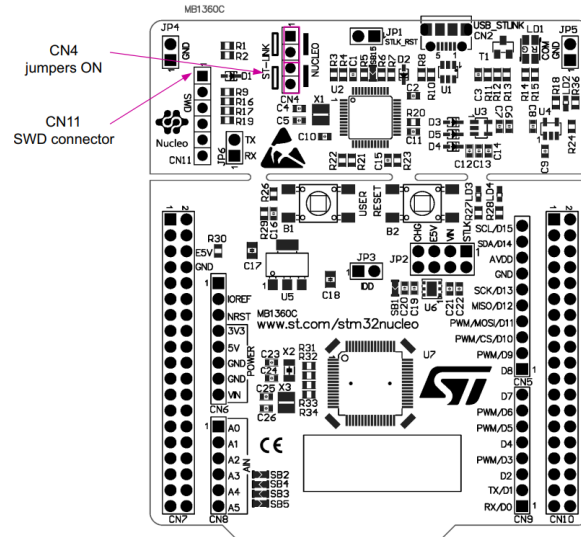
الشكل (3): الأجزاء الرئيسية من لوحة Nucleo-64



الشكل (4): أقطاب لوحة Nucleo-64

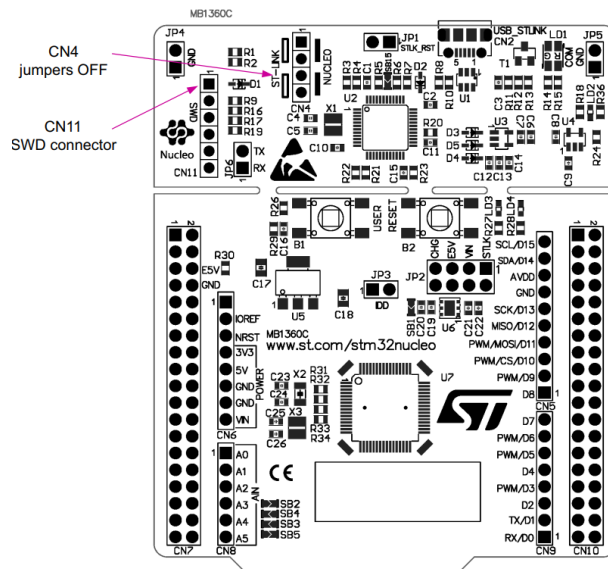
### 1.1 استخدام المبرمجة ST-LINK لبرمجة لوحة Nucleo:

لبرمجة المتحكم stm32 الموجود على لوحة Nucleo قم بوضع القصرات ضمن الـ CN4 jumper - ولكن في هذه الحالة لا يمكنك استخدام الـ ST-LINK لبرمجة متحكم خارج اللوحة أي لا يمكن استخدام الـ CN11 Jumper ، كما هو موضح بالشكل التالي:



الشكل (5): استخدام المبرمجة ST-LINK لبرمجة لوحة Nucleo

- 2.1. استخدام المبرمجة ST-LINK لبرمجة متحكم stm32 خارج لوحة Nucleo:
- لبرمجة متحكم stm32 خارج لوحة Nucleo قم بإزالة القصرات ضمن الـ CN4 jumper و استخدام الـ CN11 Jumper ، كما هو موضح بالشكل التالي:



الشكل (6): استخدام المبرمجة ST-LINK لبرمجة متحكم stm32 خارج لوحة Nucleo

قم بتوصيل (CN11) SWD connector مع المتحكم المصغر المراد برمجته وفق الجدول التالي:

Pin	CN11	Designation
1	VDD_TARGET	VDD from the application
2	SWCLK	SWD clock
3	GND	Ground
4	SWDIO	SWD data input/output
5	NRST	RESET of target MCU
6	SWO	Reserved



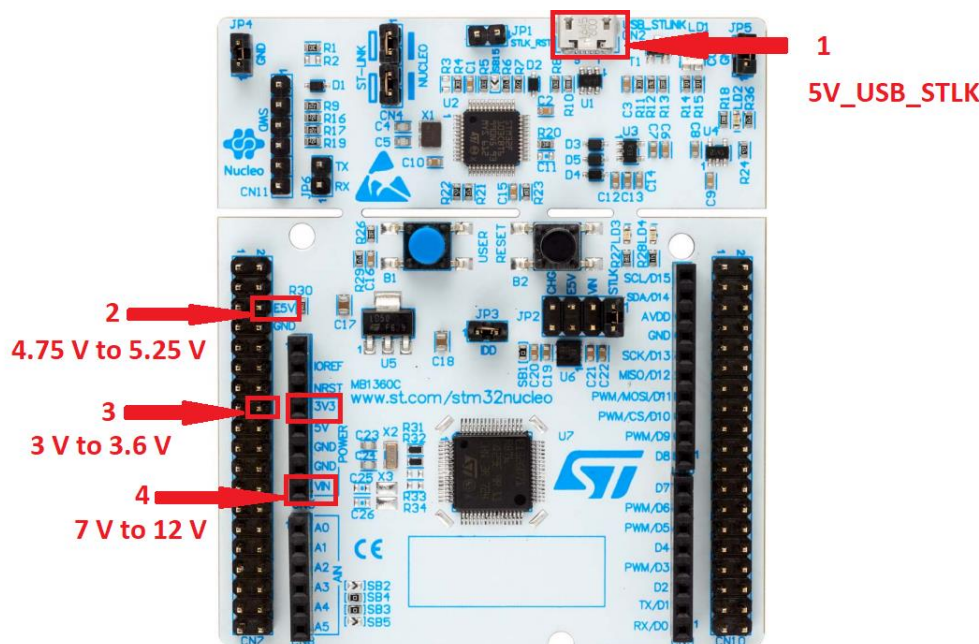
الجدول(1):كيفية توصيل متحكم خارجي مع المبرمجة ST-LINK

### 1.3 مصادر تغذية لوحة Nucleo:

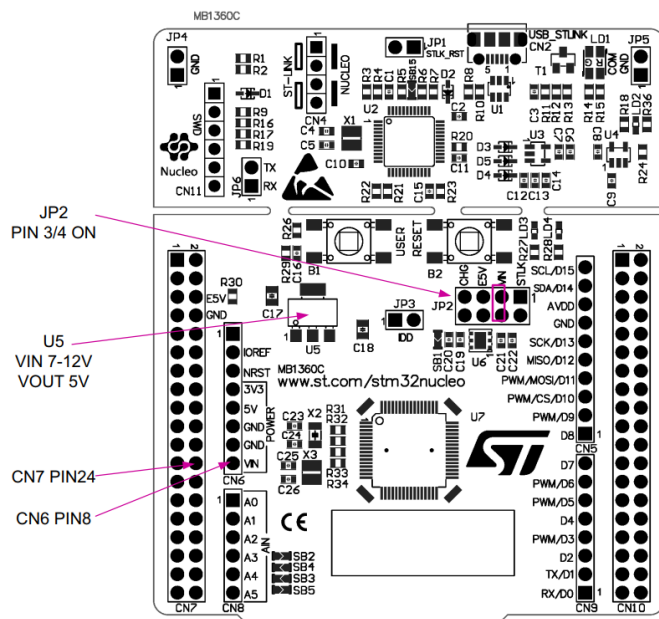
- هناك عدة طرق مختلفة كما هو موضح بالشكل(7) لإمداد لوحة Nucleo بالطاقة وهي:
- 1- من خلال منفذ usb للحاسب وعبر المبرمجة 5V\_USB\_STLK عن طريق كابل Type-A to Mini-B حيث يتم ادخال جهد 5V
  - 2- من خلال شاحن usb أي 5V\_USB\_CHG حيث يتم ادخال جهد 5V عن طريق كابل Type-A to Mini-B
  - 3- من خلال الطرف رقم pin6 ويمكن وصله بجهد بين 4.75v والـ 5.25v
  - 4- من خلال القطب 3.3v أو القطب رقم pin16 ويمكن وصله بجهد 3.3v
  - 5- من خلال القطب vin ويمكن وصله بجهد بين 7v والـ 12v

ملاحظة:

- يجب الانتباه إلى وضع القصرة المناسبة التي تدل على مصدر تغذية اللوحة ضمن الـ JP2 كما هو موضح بالشكل (8).
- القطب +5v هو قطب إخراج فقط أي يمكن استخدامه ليقيم بتغذية بعض العناصر كالدوائر وغيرها ، ولا يمكن تغذية اللوحة من خلاله.



الشكل(7): مصادر تغذية لوحة Nucleo

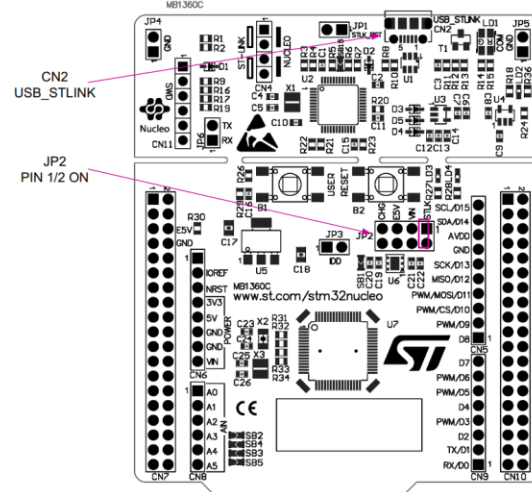


الشكل (8): مكان وضع القصرة ضمن الـ JP2 عند تغذية اللوحة من القطب vin  
يوضح الجدول التالي حدود التيار المسموح استجراره تبعاً لمصدر التغذية المستخدم لتغذية لوحة Nucleo:

Input Power	Connector pins	Voltage range	Max current	Limitation
5V_USB_STLK	CN2 PIN1	4.75 V to 5.25 V	500 mA	Max current depends on the USB enumeration: – 100 mA without enumeration – 500 mA with enumeration OK
VIN	CN6 pin 8 CN7 pin 24	7 V to 12 V	800 mA	From 7 V to 12 V only and input current capability is linked to input voltage: – 800 mA input current when VIN=7 V – 450 mA input current when 7 V<VIN<9 V – 300 mA input current when 10 V>VIN>9 V – less than 300 mA input current when VIN>10 V
E5V	CN7 pin 6	4.75 V to 5.25 V	500 mA	
5V_USB_CHG	CN2 pin 1	4.75 V to 5.25 V	500 mA	Max current depends on the USB wall charger used to power the Nucleo board
3V3	CN6 pin 4 CN7 pin 16 JP3 pin 1	3 V to 3.6 V	-	Used when ST-LINK part of PCB not used or remove SB1 and SB19

الجدول (2): حدود التيار المسموح استجراره تبعاً لمصدر التغذية المستخدم

فعلى سبيل المثال عند اختيار دائرة المبرمجة 5V\_ST\_LINK لتغذية لوحة Nucleo، ففي هذه الحالة تكون تغذية اللوحة قادمة من منفذ الـ usb لجهاز الحاسب الموصل معه اللوحة ويكون الحد الأعظمي لا ستجرار التيار هو 500mA فإذا كان جهاز الحاسب يستطيع تزويد اللوحة بهذا التيار عندها يضيء الليد LD3 باللون الأخضر ويدل على وصول التغذية للوحة، أما في حالة أن الحاسب لا يمكنه تزويد اللوحة بهذا التيار عندها سيفشل الاتصال ولن يضيء الليد LD3.



#### 1.4 الشكل (9): مكان وضع القصرة ضمن الـ JP2 عند تغذية اللوحة من منفذ usb للحاسب برمجة وتتبع أخطاء (debugging) لوحة Nucleo من خلال ST-LINK في حال استخدام تغذية خارجية لتغذية اللوحة:

عندما يكون التيار الذي سيتم استجراره من اللوحة يتجاوز الحدود المسموحة لمنفذ الـ usb للحاسب ، عندها يمكن استخدام منفذ الـ usb فقط للبرمجة وتتبع الأخطاء debug ويتم استخدام القطب Vin أو القطب E5V لتغذية لوحة الـ Nucleo. يتم ذلك من خلال الخطوات التالية:

1. قم بوضع القصرة بين pin3 و pin4 ضمن الـ JP2 لاختيار القطب Vin كمصدر تغذية أو بين pin5 و pin6 ضمن الـ JP2 لاختيار القطب EV5 كمصدر تغذية
2. قم بوصل مصدر التغذية إلى القطب Vin أو القطب EV5
3. قم بتشغيل مصدر التغذية على أن يكون مجاله بين  $7V < VIN < 12V$  من أجل Vin ، أو 5V من أجل EV5
4. يجب أن يضيء الليد LD3 باللون الأخضر
5. قم بوصل اللوحة مع منفذ الـ usb للحاسب من خلال الكابل
6. في حال لم تتبع الخطوات الـ سابقة فإن اللوحة ستأخذ تغذيتها من الحاسب وفي هذه الحالة إذا تجاوز استهلاك التيار الـ 500mA فقد يتضرر الحاسب أو لن يكفي التيار للوحة وفي هذه الحالة لن يضيء الليد LD3.

#### 1.5 مصادر الساعة في لوحة Nucleo :

1. LSE(X2): وهي عبارة عن هزاز كريستالي بتردد 32.768 kHz وهو من أجل الـ RTC (ساعة الزمن الحقيقي) المدمجة بداخل المتحكم المصغر
2. MCO: هي عبارة عن ساعة بتردد 8 MHz موجودة ضمن دائرة المبرمجة ST-LINK MCU
3. HSE(X3): هي هزاز كريستالي خارجي بتردد 8 MHz وهي غير موجودة على اللوحة.
4. HSI(high speed internal clock): هي عبارة عن ساعة المتحكم الداخلية

#### 1.6 طرق الـ Reset (التصفير) في لوحة Nucleo :

1. من خلال الضغط على المفتاح اللحظي B2
2. من خلال دائرة المبرمجة ST-LINK/V2-1
3. من خلال القطب pin3 الموجود ضمن الخط CN6 ARDUINO®

4. من خلال القطب pin14 الموجود ضمن الخط CN7 ST morpho connector

### 1.7. الليدات في لوحة Nucleo :

- يوجد أربع ليدات على لوحة Nucleo لكل منها وظيفة الخاصة بالشكل التالي:
1. LD1: هو ليد ثنائي اللون، الحالة الافتراضية له اللون الأحمر ، يومض باللونين الأحمر والأخضر أثناء الاتصال بين الحاسب والمبرمجة، ويدضيء باللون البرتقالي عند فشل الاتصال بين الحاسب والمبرمجة
  2. LD2: يضيء هذا الليد باللون الأحمر عند استرجار تيار زائد (أكبر من 500mA) على كابل الـ usb في هذه الحالة من الأفضل تغذية اللوحة تغذية خارجية من أحد القطبين Vin أو EV5 أو من خلال شاحن usb خارجي
  3. LD3: يضيء هذا الليد باللون الأخضر عند تغذية اللوحة من مصدر جهد 5V
  4. LD4: هو ليد باللون الأخضر وهو قابل للبرمجة من قبل المستخدم وهو متصل مع القطب D13 من خط ARDUINO® أي القطب PA5 من stm32 ويضيء عند كتابة واحد منطقي على هذا القطب.

### 1.8. المفاتيح اللحظية في لوحة Nucleo :

1. B1: مفتاح لحظي موصول مع القطب PC13 (Pin 3) من المتحكم stm32 وقابل للبرمجة من قبل المستخدم
2. B2: مفتاح لحظي موصول مع NRST ويستخدم لتصفير المتحكم

### 1.9. قياس استرجار التيار IDD في لوحة Nucleo :

يتم قياس استرجار التيار للوحة IDD من خلال الـ JP3 عند إزالة هذه القصرة يجب وضع مقياس التيار بين طرفي هذه القصرة لقياس التيار الذي تم استرجاره من المتحكم وفي هذه الحالة سيتم فصل التغذية عن المتحكم لحين وصل المقياس أما إذا كانت القصرة موجودة على الـ JP3 سيتم تغذية المتحكم بالطاقة بشكل طبيعي.

### 2. تنصيب الأدوات والبرامج اللازمة للعمل:

أهم البرامج الواجب تنصيبها علماً أن جميع هذه الأدوات والبرامج يتم تحميلها من موقع ST بـ شكل مجاني وهي:

- **STM32CubeIDE**: يستخدم لكتابة وتعديل الـ firmware بما يناسب المشروع كما يتيح لك فتح جلسة debug لمراقبة سير الكود في المتحكم واكتشاف الأخطاء البرمجية وتصحيحها.
- **STM32CubeProg**: وهي الأداة المستخدمة لحقن المتحكم بالـ firmware المناسب.
- **الحزمة البرمجية STM32CubeG0 Firmware package** (اختياري)

### المتطلبات الأساسية لتنصيب هذه الأدوات والبرامج:

- جهاز كمبيوتر يعمل بنظام Windows 7 (64bit) (أو أعلى)
- تثبيت حزمة Java
- الوصول إلى الويب

### الأدوات اللازمة:

- لوحة NUCLEO-G071RB board (64-pin)



• كابل Type-A to Mini-B

### a. تنصيب البرنامج STM32CubeIDE:

مهمة بيئة STM32CubeIDE هي السماح للمستخدم بكتابة الكود المناسب للتحكم المصغر، قم بتحميل الإصدار المناسب لنظام التشغيل عندك من بيئة STM32CubeIDE من خلال موقع ST

#### Get Software

Part Number	General Description	Software Version	Download	Previous versions
+ STM32CubeIDE-DEB	STM32CubeIDE Debian Linux Installer	1.3.0	<a href="#">Download</a>	Select version ▼
+ STM32CubeIDE-Lnx	STM32CubeIDE Generic Linux Installer	1.3.0	<a href="#">Download</a>	Select version ▼
+ STM32CubeIDE-Mac	STM32CubeIDE macOS Installer	1.3.0	<a href="#">Download</a>	Select version ▼
+ STM32CubeIDE-RPM	STM32CubeIDE RPM Linux Installer	1.3.0	<a href="#">Download</a>	Select version ▼
+ STM32CubeIDE-Win	STM32CubeIDE Windows Installer	1.3.0	<a href="#">Download</a>	Select version ▼

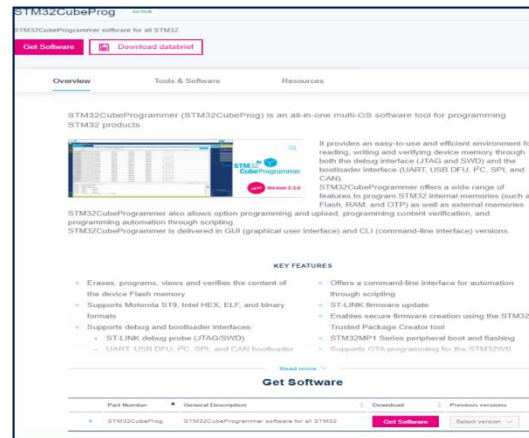
الشكل(10): تحميل الإصدار المناسب للبيئة STM32CubeIDE

تعتمد البيئة البرمجية STM32cubeIDE على لغة C/C++ ، وتستخدم مع معالجات ومتحكمات STM32 حيث تتميز بمايلي:

- I. دمج مزايا البيئة البرمجية STM32CubeMX : حيث أصبح بإمكانك ضبط إعدادات أقطاب المتحكم/المعالج (selection Pinout) والطرفيات (peripheral) ، والساعة (clock) وتوليد الكود المناسب اعتماداً على هذه الإعدادات
- II. تعتمد على البيئة البرمجية CDT / Eclipse® : مع دعم الوظائف الإضافية Eclipse® و GNU C / C++ لسلسلة أدوات Arm® ومصحح أخطاء GDB (GDB debugger) وميزات تصحيح الأخطاء المتقدمة الإضافية debug : بما في ذلك CPU ، مسجلات الطرفيات (ADC, UART, TIMERS,...etc) ، مراقبة تغيرات المتحولات المخزنة في ذاكرة المتحكم ، تتبع سير عمل الكود في الزمن الحقيقي (SWV) ، وأداة تحليل أخطاء وحدة المعالجة المركزية (CPU) ، ودعم تصحيح أخطاء وحدة المعالجة المركزية (RTOS)
- IV. تدعم المبرمجيات : من نوع (ST-Microelectronics) ST-LINK و J-Link and (SEGGER)
- V. دعم أنظمة التشغيل المتعددة: Windows® و Linux® و macOS® ، إصدارات 64 بت فقط.
- VI. عبارة عن بيئة برمجية مجانية ، ستحتاج فقط إلى إنشاء حساب على موقع ST (مجاني) ، لكنه يتطلب عنوان بريد إلكتروني).

### b. تنصيب الأداة STM32CubeProgrammer:

مهمة هذه الأداة هي تأمين الاتصال بين الحاسب والمتحكم المصغر عن طريق إما الاتصال التسلسلي (UART) عبر كابل الـ FTDI أو من خلال الـ (SWD) عبر دائرة المبرمجة ST-LINK ، قم بتحميل الإصدار المناسب لنظام التشغيل عندك من الأداة STM32CubeProgrammer من خلال موقع ST كما هو موضح بالشكل التالي:



الشكل (11): تحميل الإصدار المناسب للأداة STM32CubeProgrammer

c. **تنصيب الحزمة البرمجية STM32CubeG0 Firmware package (اختياري)**  
قم بتحميل الإصدار المناسب للحزمة البرمجية STM32CubeG0 والتي تتضمن العديد من الأمثلة الجاهزة لبوردة Nucleo، وذلك من خلال موقع شركة ST.

#### Get Software

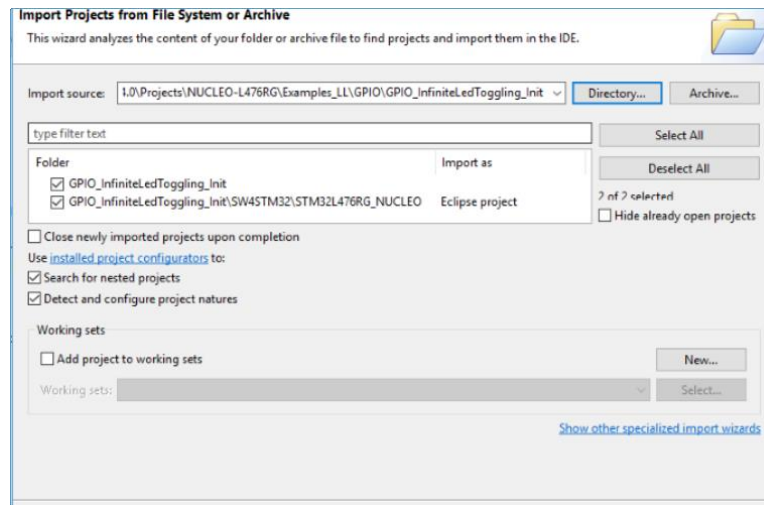
Part Number	General Description	Latest version	Download	All versions
+ Patch_CubeG0	Patch for STM32CubeG0	1.5.1	<a href="#">Get latest</a>	<a href="#">Select version</a>
+ STM32CubeG0	STM32Cube MCU Package for STM32G0 series	1.5.0	<a href="#">Get latest</a> <a href="#">Get from GitHub</a>	<a href="#">Select version</a>

الشكل (12): تحميل الإصدار المناسب للحزمة البرمجية STM32CubeG0

ملاحظة: تم دمج الأداة STM32cubeMx مع البيئة STM32CubeIDE لذا لا داعي لتنصيبها، وهي المسؤولة عن ضبط الإعدادات الأساسية للمتحكم كضبط تردد الساعة، ضبط المداخل والمخارج والمؤقتات وغيرها.

d. استخدام بيئة STM32CubeIDE وفتح أحد الأمثلة الجاهزة والبدء بجلسة Debug:

- قم بفتح بيئة STM32CubeIDE
- اختر مساحة العمل الافتراضية default workspace
- من القائمة ملف اختر Open Projects from File System

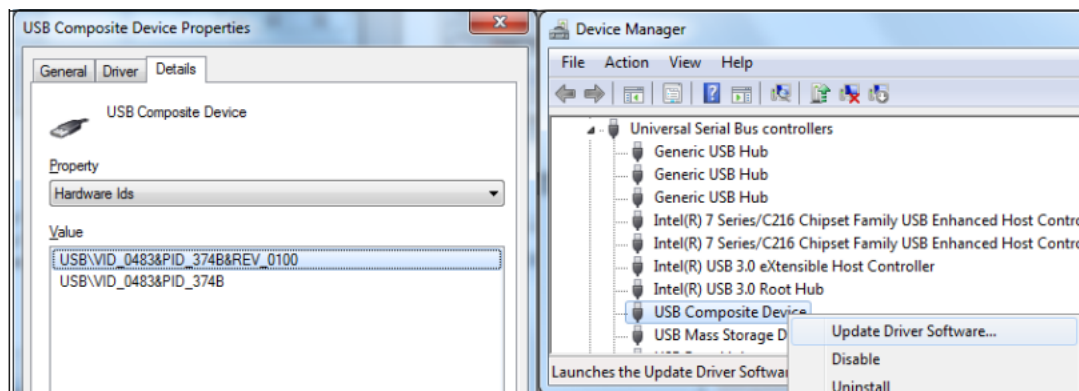


الشكل(13):فتح مشروع في بيئة STM32cubeIDE

- اختر أحد الأمثلة التي تم تنزيلها وقم بفتحه من خلال بيئة STM32cubeIDE

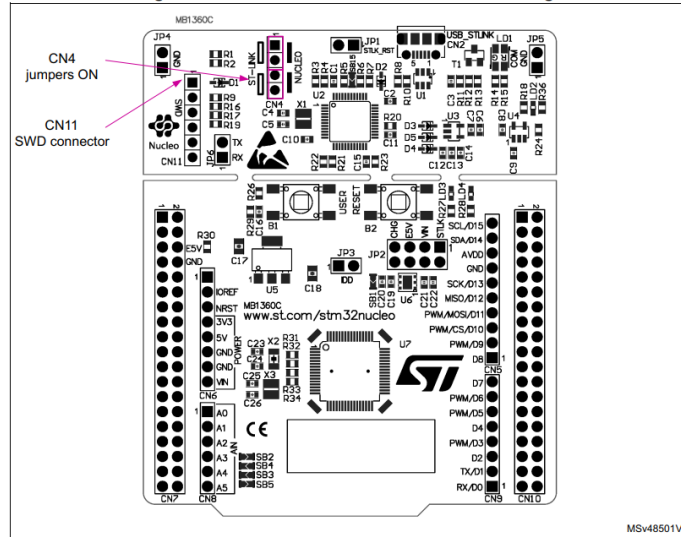
ملاحظة:

- تأكد من تنصيب تعريف ST-LINK/V2-1 على جهاز الحاسب من خلال تحميله من موقع ST، حيث عند و صلك ال - ST-Link في البداية مع الحاسب من خلال كبل Type-A to Mini-B تظهر كجهاز غير معروف ثم من خلال نافذة إدارة الأجهزة قم باختيار ال - ST-Link وتحديث التعريف باستخدام التعريف الذي تم تنزيله من موقع ST.



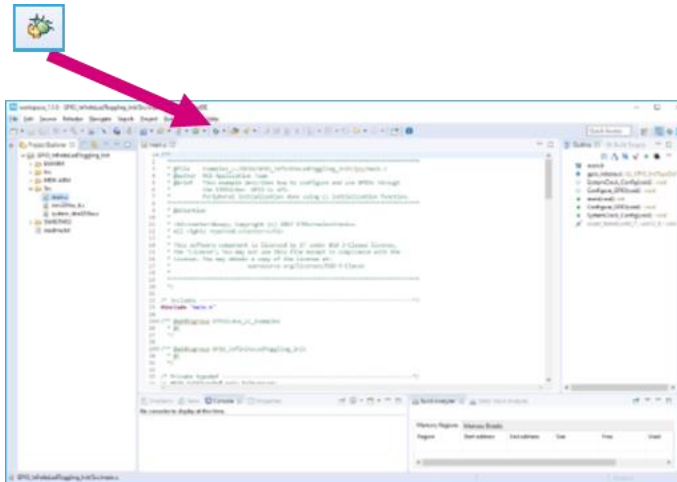
الشكل(15):تنصيب تعريف ST-LINK/V2-1

- تأكد من وضع ال - CN4 jumper المسؤول عن وصل ال - ST-Link مع بوردة Nucleo وبالتالي يكون بإمكانك حقن الكود بمتحكم stm32 الموجودة ضمن اللوحة.



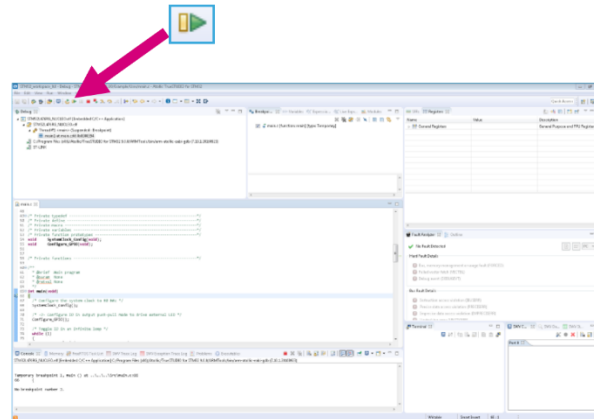
الشكل (16): لوحة Nucleo

- قم بالضغط على أيقونة الـ debug لتبدأ دائرة ST-Link برفع الكود إلى المتحكم ثم البدء بجلسة debug لمراقبة الكود خطوة بخطوة وتصحيح الأخطاء البرمجية.



الشكل (17): فتح جلسة debug

- ثم قم بالضغط على أيقونة Resume



الشكل(18):البدا بجلسة debug

أصبح بإمكانك:

- إيجاد وتنصيب المكتبات المختلفة لمتحكمات Stm32
- فتح مشروع جديد (أحد الأمثلة المتوفرة) ضمن بيئة STM32CubeIDE
- رفع الكود للمتحكم وتشغيل المشروع بنمط Debug

و أصبح لديك على جهاز الحاسب:

- مكتبات المتحكم STM32G0
- أمثلة جاهزة للمتحكم STM32G0
- أداة STM32CubeProg للاتصال بالمتحكم ورفع الكود له
- بيئة STM32CubeIDE لضبط إعدادات المتحكم وكتابة الكود المناسب للتطبيق