

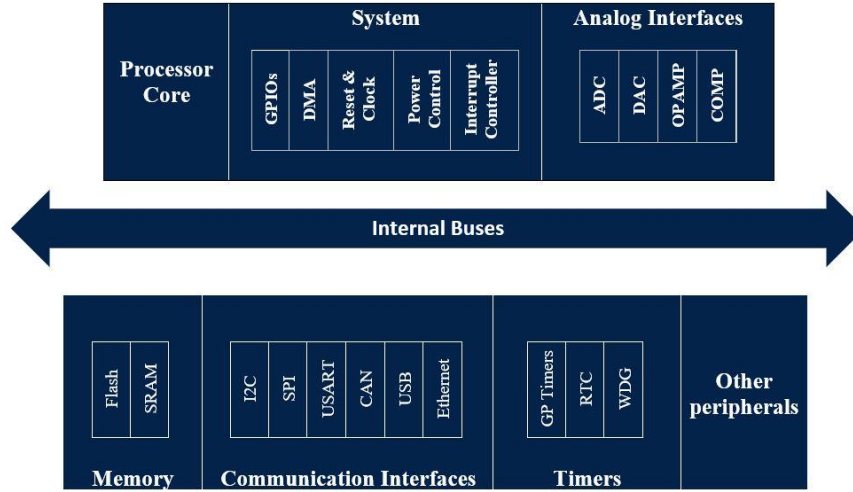
# مدخل إلى متحكمات STM32

## محتويات الجلسة:

1. ماهو المتحكم المصغر (MCU)
  - 1.1. تعريف بالمتحكمات المصغرة
  - 1.2. بنية المتحكم المصغر
  - 1.3. نظرة عامة عن متحكمات stm32
  - 1.4. نظرة عامة عن بنية Cortex-M
  - 1.5. مزايا متحكمات stm32
  - 1.6. الأنواع المختلفة من متحكمات stm32
2. الحاجة لاستخدام المتحكمات المصغرة
  - 2.1. الفرق بين المتحكم المصغر والمعالج المصغر
  - 2.2. تطبيقات المتحكمات المصغرة
3. كيفية اختيار المتحكم المصغر المناسب لمشروعك
4. كيفية استخدام متحكمات stm32
  - 4.1. أهم اللوحات التطويرية الخاصة بمتحكمات stm32
  - 4.2. كيفية برمجة متحكمات stm32 وأهم الأدوات والبرامج اللازمة

## 1. ماهو المتحكم المصغر (MCU)

- 1.1. تعريف المتحكم المصغر:  
المتحكم المصغر (يسمى أيضاً  $\mu C$  أو MCU) عبارة عن شريحة كمبيوتر مدمجة تتحكم في معظم الأدوات والأجهزة الإلكترونية التي نستخدمها بشكل يومي، فهو عبارة عن دائرة متكاملة مدمجة قابلة للبرمجة مصممة للتحكم في عملية محددة في نظام مدمج معين Embedded system.
- 1.2. بنية المتحكم المصغر:  
يحتوي المتحكم المصغر على المعالج والذاكرة والأجهزة الطرفية للإدخال / الإخراج (I / O) على شريحة واحدة. يمكن توسيع مكوناته لتشمل: الإدخال / الإخراج الرقمي ، الإدخال / الإخراج التشابهي ، الموقتات ، واجهات الاتصال ، مؤقت المراقبة watchdog (وهو مؤقت مسؤول عن اكتشاف انتهاء الزمن المحدد لتنفيذ التعليمات) وغيرها من الطرفيات.  
المعالج عبارة عن شريحة صغيرة موجودة في المتحكم يقوم بمسح المداخل الرقمية والتشابهية وإصدار الأوامر المناسبة للمخارج  
يبين الشكل التالي بنية المتحكم المصغر:



الشكل (1) بنية المتحكم المصغر

### 1.3.

#### نظرة عامة عن متحكمات stm32

تقوم شركة STMicroelectronics بتصنيع متحكمات STM32 MCUs التي تحتوي على معالج Arm® Cortex®-M. وهو عبارة عن نوى معالج 32 RISC Arm بت ، والتي تعتبر مثالية من ناحيتي التكلفة المادية و الاستهلاك المنخفض للطاقة.

تقوم شركة STMicroelectronics بتصنيع متحكمات STM32 MCUs التي تعتمد على معالج Arm® Cortex®-M. وهي عبارة عن نوى معالج 32 RISC Arm بت ذات الكلفة واستهلاك الطاقة المنخفض.

حيث RISC Arm (Reduced Instruction Set Computer) عبارة عن مجموعة تعليمات تسمح بتقسيم كل أمر من الأوامر إلى عدة تعليمات بسيطة غير معقدة تقوم كل منها بأداء مهمة صغيرة.

تقدم عائلة STM32 مجموعة كبيرة من المتحكمات التي تجمع بين:

- سهولة التطوير بفضل توفر عدة أدوات للتطوير بما في ذلك الأدوات المجانية منها STM32cubeIDE ، STM32cubeProgrammer ، STM32cubeMonitor وغيرها من الأدوات.

- أداء عالٍ جداً بفضل Arm® Cortex®-M core و ST ART Accelerator™ لبعض العائلات

- إمكانية العمل بالزمن الحقيقي

- القدرة على معالجة الاشارات الرقمية

- الاستهلاك منخفض الطاقة والجهد المنخفض

- ميزات الرسم

تعتمد العديد من التطبيقات على متحكمات STM32 (سيتم تفصيلها لاحقاً) ومنها:

- أنظمة التحكم الصناعية

- أجهزة واجهات المستخدم

- التحكم في المحركات

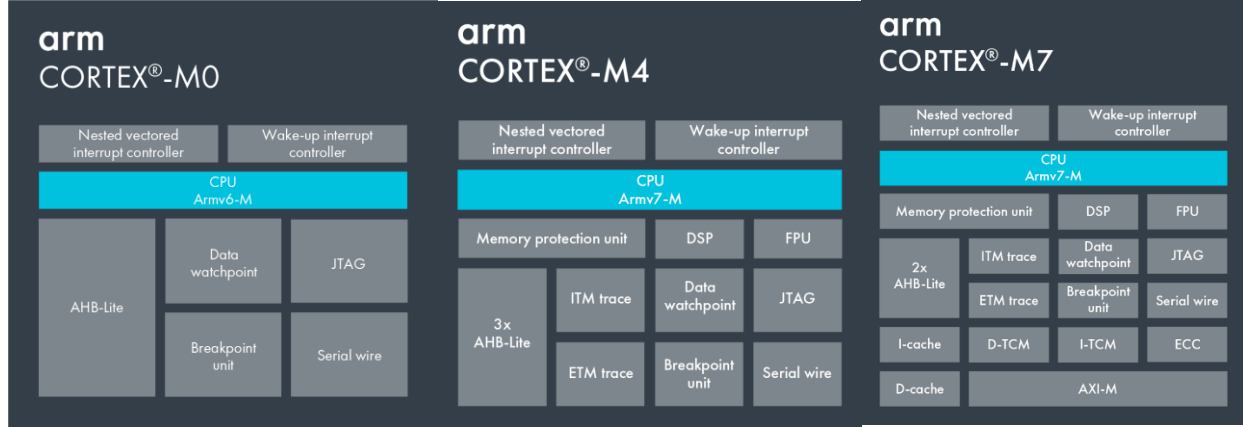
- الأجهزة والأدوات الطبية

- المباني والأمن (أجهزة الإنذار ، عدادات الطاقة ...)

- انترنت الأشياء
- وغيرها....

## 1.4 الهيكلية Cortex-M :

وتتضمن عدة هيكليات، أهم هذه الهيكليات التي تم استخدامها في بناء متحكمات STM32 هي:



الشكل (2) هيكليات Cortex-M المستخدمة في متحكمات stm32

حيث:

Feature	Cortex-M0	Cortex-M0+	Cortex-M3	Cortex-M4	Cortex-M33	Cortex-M7
Instruction set architecture	Armv6-M	Armv6-M	Armv7-M	Armv7-M	Armv8-M Mainline	Armv7-M
	Thumb, Thumb-2					
Pipeline stages	3	2	3	3	3	6
Memory Protection Unit	No	Yes	Yes	Yes	Yes	Yes
Maximum MPU regions	0	8	8	8	16	16
Trace (ETM or MTB)	No	MTB	ETMv3	ETMv3	MTB and/or ETMv4	ETMv4
DSP	No	No	No	Yes	Yes	Yes
Floating point hardware	No	No	No	Yes	Yes	Yes
Bus protocol	AHB Lite	AHB Lite	AHB Lite, APB	AHB Lite, APB	AHB5	AXI4, AHB Lite, APB, TCM
Maximum # external interrupts	32	32	240	240	480	240
CMSIS Support	Yes					

الشكل (3) الفروقات بين هيكليات Cortex-M المختلفة

## 1.5 مزايا متحكمات stm32

### 1.5.1 الطرفيات الموجودة في متحكمات STM32 peripherals :

يتم تطوير الطرفيات الموجودة في متحكمات stm32 باستمرار لتواكب احتياجات السوق كما يتم إصلاح المشاكل التقنية التي قد تتواجد في الطرفيات الموجودة مسبقاً. مثال على بعض الطرفيات التي تمت إضافتها كاشف انخفاض الجهد ونظام الـ (clock security system (css أيضاً مؤقت المراقبة watchdog وغيرها لضمان عمل التطبيقات التي تحتاج إلى وثوقية عالية بعملها. كما تم إضافة ميزة قفل وحماية الكود من خلال إضافة خاصية الحماية لذاكرة المتحكم Flash memory.

### 1.5.2. العناصر التي ستحتاجها كي تبدأ باستخدام متحكمات stm32 :STM32 hardware setup

1. مصدر تغذية مستمرة لتغذية المتحكم.
2. هزاز كريستالي: تحتوي متحكمات stm32 على هزاز RC داخلي ولكن يبقى الهزاز الكريستالي الخارجي أكثر استقراراً من الهزاز الداخلي.
3. مراقبة الكود وتتبع الأخطاء أثناء عمل المتحكم وذلك من خلال أداة تدعى الـ debugger والتي يتم تنصيبها بشكل تلقائي عند تنصيب برنامج stm32cubeIDE ، ولكي تتمكن من عمل debug للكود لابد من وصل المتحكم stm32 بدارة وسيطية (على سبيل المثال ST-LINK) بينه وبين الحاسب وهناك طريقتين لوصل المتحكم مع الحاسب:
  - a. SWD (Serial Wire Debug): ويتم ذلك من خلال ثلاثة أقطاب من المتحكم (Reset, clock, data) وتستخدم لتتبع أخطاء المتحكم نفسه.
  - b. JTAG (Joint Test Action Group): ويتم ذلك من خلال أربع أو خمس أقطاب من أقطاب المتحكم وتستخدم لتتبع واكتشاف أخطاء البوردرات أثناء التصنيع.

### 1.5.3. مصادر المعلومات عن متحكمات stm32:

- يضم موقع ST website جميع المعلومات المهمة عن مختلف عائلات متحكمات stm32 ويشمل مايلي:
1. Datasheet (DS): وتتضمن الخصائص الكهربائية بالإضافة إلى شرح لجميع أقطاب كل متحكم من متحكمات stm32.
  2. Errata sheet (ES): وتتضمن شرح كامل لقيود العمل (limitation) الخاصة بكل متحكم من متحكمات stm32 كي يعمل بشكل أمثل وماهي الحدود المسموح تجاوزها.
  3. Reference manual (RM): ويتضمن شرح لمواصفات المتحكم مع تفصيل لكامل مسجلاته.

### 1.6. أصناف متحكمات stm32

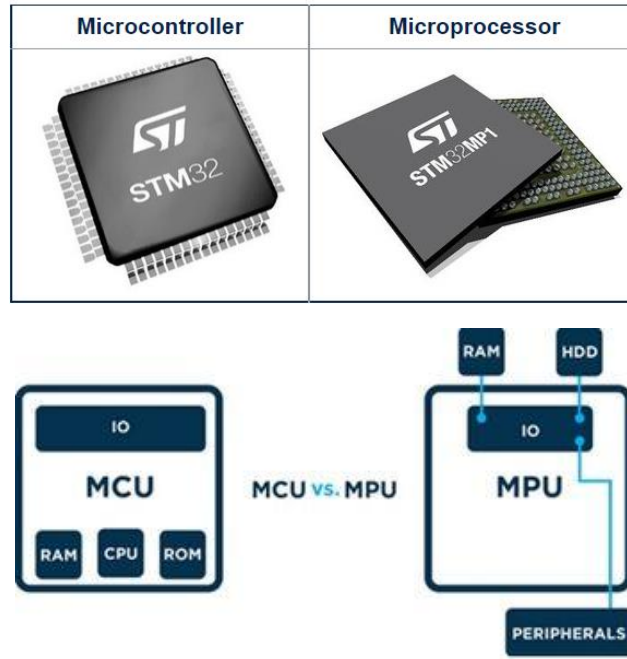
تضم متحكمات stm32 لحد الآن 16 عائلة من المتحكمات المبنية على Cortex-M 16 مقسمة إلى أربع مجموعات أساسية كما هو موضح بالجدول التالي:

	Family	Core	Max Frequency	Flash
High Performance	STM32H7	Cortex-M7 - Cortex-M4	480 MHz - 240 MHz	1 to 2 Mbytes
	STM32F7	Cortex-M7	216 MHz	256 Kbytes to 2 Mbytes
	STM32F4	Cortex-M4	180 MHz	64 Kbytes to 2 Mbytes
	STM32F2	Cortex-M3	120 MHz	128 Kbytes to 1 Mbyte
Mainstream	STM32G4	Cortex-M4	170 MHz	32 to 512 Kbytes
	STM32F3	Cortex-M4	72 MHz	16 to 512 Kbytes
	STM32F1	Cortex-M3	72 MHz	16 Kbytes to 1 Mbyte
	STM32G0	Cortex-M0+	64 MHz	16 to 512 Kbytes
	STM32F0	Cortex-M0	48 MHz	16 to 256 Kbytes
Ultra-low-power	STM32L5	Cortex-M33	110 MHz	256 to 512 Kbytes
	STM32L4+	Cortex-M4	120 MHz	512 Kbytes to 2 Mbytes
	STM32L4	Cortex-M4	80 MHz	64 Kbytes to 1 Mbyte
	STM32L1	Cortex-M3	32 MHz	32 to 512 Kbytes
	STM32L0	Cortex-M0+	32 MHz	8 to 192 Kbytes
Wireless	STM32WB	Cortex-M4 - Cortex-M0+	64 MHz - 32 MHz	256 Kbytes to 1 Mbyte
	STM32WL	Cortex-M4	48 MHz	64 Kbytes to 256 Kbytes

## 2. الحاجة لاستخدام المتحكمات المصغرة:

### 2.1. الفرق بين المتحكم المصغر والمعالج المصغر :

على الرغم من التشابه الكبير بين المتحكم المصغر والمعالج المصغر ولكن هناك عدة اختلافات بينهما، حيث يمكن اعتبار المتحكم المصغر (MCUs) حاسب مصغر يحتوي على كافة العناصر الموجودة بداخل الحاسب (معالج، ذواكر، طرفيات، منافذ إدخال ومنافذ إخراج وغيرها) وعلى الرغم من ذلك فهو أرخص وأصغر حجماً من المعالج المصغر والذي يحتاج عند استخدامه إلى وصل ذاكرة معه بالإضافة إلى وصل طرفيات للإدخال والإخراج وغيرها، ومع كل هذه الاختلافات يبقى المتحكم المصغر هو الأنسب لبناء الأنظمة المدمجة الصغيرة والمتوسطة.



الشكل (4): الفرق بين المتحكم المصغر والمعالج المصغر

### 2.2. تطبيقات المتحكم المصغر MCUs Application:

يتكون أي نظام مدمج من مكونين أساسيين هما:

1. **Hardware**: ويتضمن المتحكم المصغر MCU بالإضافة إلى مجموعة عناصر أخرى كالحساسات والمقاومات والمكثفات والمفاتيح وغيرها..

2. **Software** : البرنامج أو الكود أو كما نسميه firmware والذي يلعب الدور الأساسي في التحكم بكامل النظام المدمج فمثلاً يقوم بقراءة القيم القادمة من الحساسات التشابيهية والرقمية وإصدار الأوامر المناسبة وفقاً لهذه القيم (على سبيل المثال: عند تصميم نظام لمراقبة درجة الحرارة ، تكون مهمة البرنامج قراءة القيم القادمة من حساس الحرارة وإصدار الأوامر بتشغيل أو إطفاء المروحة أو تشغيل أو إطفاء السخان، وتشغيل إنذار معين عند تجاوز درجة الحرارة المسموحة).

ونظراً لسهولة التعامل مع المتحكمات المصغرة بالإضافة إلى سعرها المنخفض مقارنةً بالمعالجات المصغرة، فهي تعتبر الأنسب للتحكم بالأنظمة المدمجة التي لا تحتاج إلى عمليات

معالجة معقدة، وهناك العديد من الأجهزة الالكترونية حولنا التي تستخدم المتحكمات المصغرة على سبيل المثال:

- في حياتنا اليومية (الغسالة ، التلفزيون ، ساعات التنبيه المختلفة ، أجهزة التحكم عن بعد ، إنترنت الأشياء ، جهاز تحكم بشاشة تعمل باللمس ...).
- في المجال الطبي (قياس نبضات القلب ، أجهزة قياس السكر في الدم ، أجهزة قياس الطاقة ، ...).
- أنظمة الحماية (التحكم في درجة الحرارة ، المراقبة ، الكاميرات ، أجهزة الإنذار ...).

### 2.3 أسباب استخدام المتحكم المصغر :Reasons to use an MCU

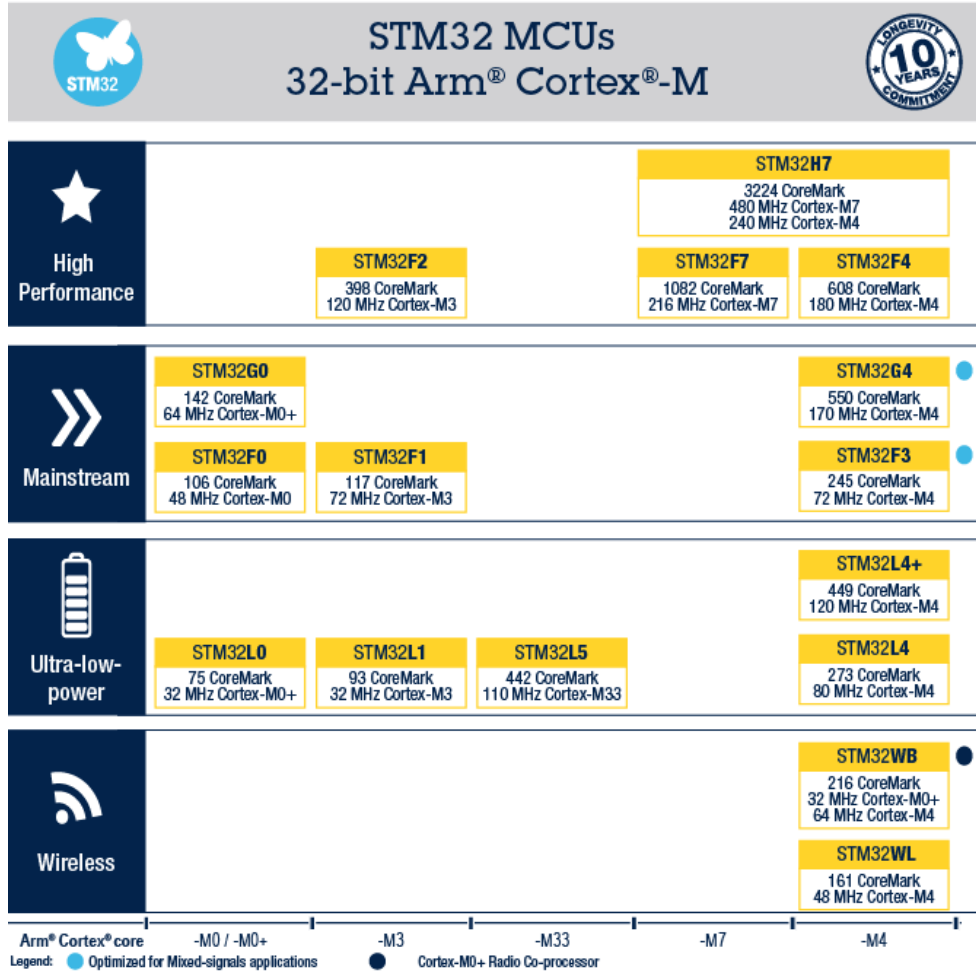
هناك عدة أسباب لاستخدام المتحكمات المصغرة منها:

- **استهلاك الطاقة Power consumption:** إحدى أهم النقاط التي يجب مراعاتها عند تصميم أي جهاز إلكتروني مدمج هو مدى كفاءته في استهلاك الطاقة وخاصةً الأجهزة المحمولة التي تعمل على مصدر طاقة محدود كالبطاريات فقد يكون المطلوب أن يعمل الجهاز لساعات طويلة أو حتى لشهور أو سنوات دون أن يكون هناك حاجة لتبديل البطاريات. قد تستهلك MCU بضع من الميكرووات في وضع التشغيل. يمكن أن ينخفض استهلاك الطاقة هذا بمقدار بضع نانوات في بعض أوضاع التوقف أو السكون (stop or sleep mode). هذا يجعلها مناسبة جداً للتطبيقات منخفضة الطاقة.
- أحد الأهداف الأساسية لشركات تصنيع المتحكمات المصغرة هو بناء متحكمات ذات كفاءة عالية ، مع قدرتها على القيام بالعمليات التي تحتاج إلى معالجة أعقد، وأجهزة طرفية متطورة ، وذات قدرة على التعامل مع الرسومات graphics ، والاتصال RF وحتى قدرتها على القيام بتطبيقات الذكاء الاصطناعي، مما يجعلها منافساً للمعالجات المصغرة MPU خصوصاً للأجهزة التي لا تتطلب التطبيقات فيها الكثير فيما يتعلق بالأداء وحجم الذاكرة.
- **التكلفة Cost:** نظراً لتصميمها وحجمها الصغير ، فإن تكلفة إنتاج وصيانة المتحكمات المصغرة منخفضة نسبياً، حيث يمكن أن تكون تكلفة التصنيع أقل من 0.10 دولار لكل متحكم مصغر ، ومتوسط سعر متحكم 32 بت أقل من 1 دولار. نقطة أخرى هي أن MCU فعال ، إذا لم يعمل بشكل صحيح فيمكن بسهولة إعادة برمجته تبعاً للتطبيق المصمم.
- **القابلية لإعادة الاستخدام Reusable :** كما ذكرنا في القسم السابق ، هناك ميزة أخرى للميكروكونترولر وهي أنها قابلة للبرمجة لذا يمكن إعادة استخدام نفس MCU في حالات وخيارات وتطبيقات مختلفة فقط عن طريق إعادة برمجتها. لذلك ، لتغيير بعض البارامترات أو بعض البيانات ، ليست هناك حاجة لشراء MCU أخرى ، فالأمر يتعلق فقط بإعادة برمجة نفس MCU.

### 3. كيفية اختيار المتحكم المصغر المناسب لمشروعك:

تقدم STMicroelectronics مجموعة كبيرة ومتنوعة من STM32 MCUs ، بناءً على نوى Arm® Cortex®-M0 و M0+ و M3 و M4 و M33 و M7 ، اعتماداً على العديد من الميزات والبارامترات. لاتخاذ أفضل قرار بشأن اختيار متحكم مصغر STM32 ، هناك العديد من العوامل التي يجب مراعاتها ، ولا سيما البنية الداخلية للمتحكم، ومتطلبات الذاكرة والطاقة ، والأجهزة الطرفية التي يجب دعمها مثل الأجهزة الطرفية للاتصال: USB و I2C و SPI و UART والتكلفة المادية.

يعتمد الاختيار على المتطلبات الوظيفية للتطبيق: أداء منخفض للغاية للطاقة ، أو أداء عالي ، أو اتصال لاسلكي.



الشكل (5): عائلات المتحكمات STM32

### 1.3. المتحكمات المناسبة للتطبيقات الرسومية:

يمكن أخذ العديد من العوامل في الاعتبار لضمان الاختيار الصحيح لمتحكم STM32 وفقاً لمتطلبات التطبيقات الرسومية. أول شيء تحتاج إلى تحديده هو نواة المتحكم والعائلة التي ينتمي إليها، مع الانتباه إلى حجم ذاكرة الفلاش وذاكرة الوصول العشوائي Ram المناسبة للتطبيق بناءً على حجم التطبيق الرسومي المراد تصميمه. أيضاً يجب الأخذ بعين الاعتبار:

- دقة العرض المطلوبة من التطبيق الرسومي المراد تصميمه، عمق اللون، الأداء المتوقع، وجميع الأجهزة الطرفية المطلوبة مثل:

- مسرع (DMA2D) Chrom-ART™
- برنامج ترميز JPEG
- Chrom-GRC™
- شاشة LCD-TFT
- FMC / FSMC

## More HMI and IoT applications with STM32F769 Discovery kit



الشكل (6): المتحكمات المناسبة للتطبيقات الرسومية

### 2.3. المتحكمات المناسبة للتطبيقات ذات الاستهلاك المنخفض للطاقة:

تقدم ST مجموعة كبيرة من المتحكمات منخفضة الطاقة: (+ Arm® Cortex® M0 STM32L0 و STM32L1 (Cortex M3) و STM32L4 و STM32L4 + (Cortex M4) و STM32L5 و STM32L5 (Cortex-M33)) مما يتيح لك بناء نظام مدمج منخفض الطاقة باستخدام هذه المتحكمات.

لتحديد أفضل متحكم لتطبيق منخفض الاستهلاك للطاقة ، تحتاج أولاً إلى تحديد نواة المعالج ، ثم ضع في اعتبارك أداء المتحكم وكفاءته باستهلاك الطاقة ، حيث يعتمد اختيار المتحكم المناسب على العديد من العوامل الرئيسية التي يجب مراعاتها مثل:

- استهلاك الطاقة، حجم الذاكرة، ترددات العمل، أوضاع الطاقة المنخفضة المختلفة : وضع التشغيل منخفض الطاقة ، ووضع السكون ، ووضع السكون منخفض الطاقة ، ووضع التوقف ، ووضع الاستعداد، الزمن اللازم للاستيقاظ، مصادر التنبيه ، التي تسمح بتقليل متوسط الاستهلاك الحالي للتطبيق.
- وجود (Switched-Mode Power Supply (SMPS لتقليل استجرار التيار للمتحكم.
- وجود متحكم الوصول المباشر للذاكرة (DMA) والذي يفيد في تحسين الأداء وتقليل الطاقة المستهلكة.
- معرفة مجال جهد العمل للمتحكم ، والأجهزة الطرفية التي يمكن أن تعمل في وضع الطاقة المنخفضة لتقليل استهلاك الطاقة .



## New ultra-low-power MCU series based on Arm® Cortex®-M33



- A full set of security
- Extended battery lifetime
- High integration & innovation

الشكل(7): المتحكمات المناسبة لتطبيقات الطاقة المنخفضة

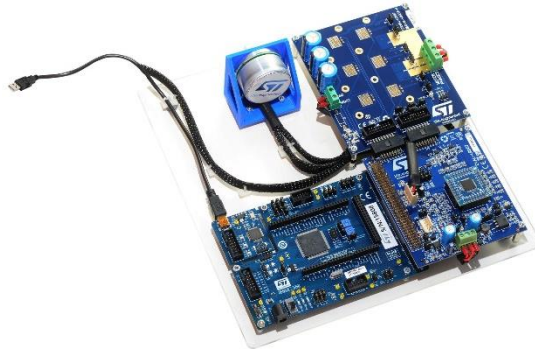
### 3.3. المتحكمات المناسبة لتطبيقات التحكم بالمحركات:

لتحديد MCU المناسب لتطبيق التحكم في المحرك تحتاج إلى التركيز على وجود الطرفيات التالية في الـ MCU:

- المبدلات التناظرية إلى رقمية (ADC)
- عدد المقارنات (COMP)
- المبدلات الرقمية إلى تناظرية (DAC)
- HRTIM
- المؤقتات

حيث يعتمد ذلك بشكل كلي على متطلبات التطبيق الذي تقوم بتصميمه مثل:

الأداء المطلوب، متطلبات الذاكرة، عدد المحركات التي سيتم التحكم بها ، OPAMP، سرعة المحرك المطلوبة، هل يتطلب مؤقت 16 بت، هل سيحتاج التطبيق لوجود شاشة رسومية ، لذا فأنت بحاجة إلى جهاز طرفي لشاشات lcd.



الشكل(8): المتحكمات المناسبة لتطبيقات التحكم بالمحرك

### 4.3. الأداة ST-MCU-FINDER :

بفضل أداة ST-MCU-FINDER التي تتيح البحث السهل عن STM32 MCU بمعايير متعددة بما في ذلك نواة المعالج وتردد عمل الـ CPU، وسعة الذاكرة Eeprom والذاكرة flash والذاكرة Ram والسعر والحزمة ودرجة الحرارة والأجهزة الطرفية مثل المؤقتات والمقارنات والمبدلات التشابعية رقمية ADC والمبدلات الرقمية إلى تشابعية DAC.

توفر أداة ST-MCU-FINDER أداة رسومية لها محاكاة simulation تسمح لك باختيار الأداء المطلوب لتحديد MCU أو MPU الذي يناسب احتياجات التطبيق الخاص بك.

بالإضافة إلى ذلك، توفر هذه الأداة إمكانية الوصول إلى البيانات والوثائق الفنية للتحكم المختار ( datasheets, reference manuals, application notes, user manuals, programming manuals, errata sheets).

تم دمج أداة ST-MCU-FINDER في أدوات STM32CubeIDE و STM32CubeMX لتحديد MCU أو لوحة التطوير المناسبة.

يمكنك تحديد المتحكم الذي تحتاجه بناءً على العديد من الخيارات مثل MCU Selector أو Board Selector أو Cross selector وفقاً لمتطلبات التطبيق الخاص بك. على سبيل المثال، يتطلب تحديد MCU لتطبيق TouchGFX باستخدام ST-MCU-FINDER الأجهزة الطرفية التالية:

LTDC ، FMC SDRAM ، CRC،DMA2D

بإمكانك تنزيل الأداة ST-MCU-FINDER-PC من موقع ST على الإنترنت.

#### 4. كيفية استخدام متحكمات STM32:

هناك طريقتين أساسيتين لاستخدام متحكمات STM32:

- 1- من خلال بناء وتصميم لوحتك الخاصة
- 2- من خلال استخدام إحدى اللوحات التطويرية المتاحة

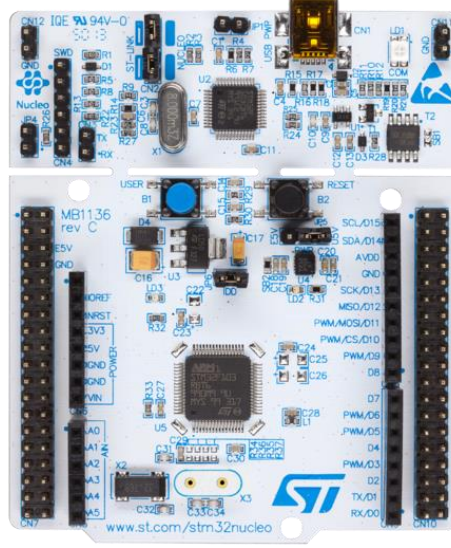
##### 1.4 أنواع اللوحات المتوفرة Boards type

هناك عدة لوحات متوفرة لمتحكمات stm32 نذكر منها:

##### 1- Nucleo board:

تتميز لوحات STM32 Nucleo بالسعر المقبول و سهولة التعامل مع المتحكم stm32 ، حيث بإمكانك تجربة أي فكرة برمجية عليها بسهولة كبيرة لعدة أسباب:

- إمكانية الوصول المباشر إلى معظم مداخل/ مخارج GPIO المتحكم و توصيل العناصر الإلكترونية المختلفة معها .
- وجود مبرمجة ST-LINK مدمجة مع اللوحة يمكنك من حقن الكود بالمتحكم بسهولة كما يمكنك من اختبار وتفحص أخطاء المتحكم من خلال الـ debug.
- وجود عدد كبير من الأمثلة البرمجية
- أيضاً هي متوافقة مع بيئة أروينو ARDUINO® connector compliant



الشكل(9): لوحة التطوير Nucleo

## -2 STM32 Discovery kits:

تتيح مجموعات STM32 Discovery للمستخدمين تطوير التطبيقات بسهولة باستخدام متحكمات مصغرة عالية الأداء. فهي تتيح للمبتدئين و للمبرمجين ذوي الخبرة على حدٍ سواء، تجربة التطبيقات المختلفة على متحكمات stm32.

أيضاً تعتبر حل رخيص وكامل لتقييم مزايا متحكمات ومعالجات STM32، حيث تتميز بـ:

- إمكانية الوصول المباشر إلى معظم مداخل/ مخرج GPIO المتحكم و توصيل العناصر الإلكترونية المختلفة معها .
- وجود مبرمجة ST-LINK مدمجة مع اللوحة يمكنك من حقن الكود بالمتحكم بسهولة كما يمكنك من اختبار وتفحص أخطاء المتحكم من خلال الـ debug.
- وجود عدد كبير من الأمثلة البرمجية
- أيضاً هي متوافقة مع بيئة أردوينو ARDUINO® connector compliant
- تحتوي بعض اللوحات على شاشة عرض



الشكل(10): STM32 Discovery kits

### 3- STM32 Eval boards:

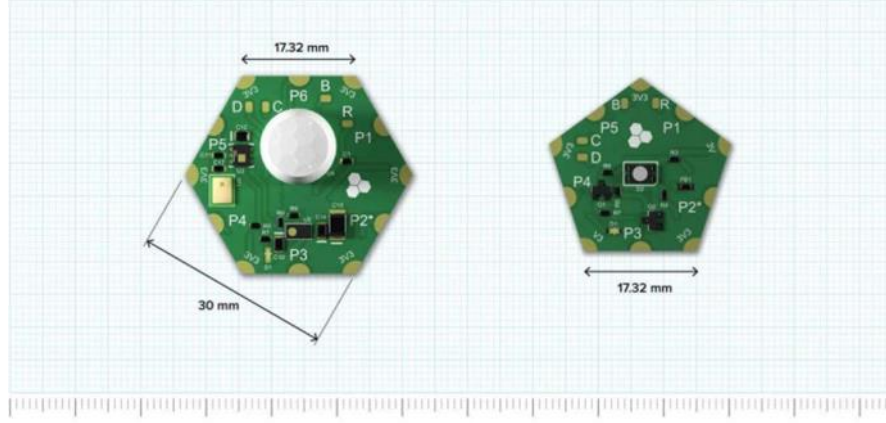
تم تصميم لوحات STM32 Eval لتكون بمثابة منصة تطوير متكاملة لمتحكمات ومعالجات STM32، حيث تختلف عن لوحتي Nucleo و Discovery بأنها تحتوي على العديد من العناصر الالكترونية كالحساسات و أجهزة الإرسال والاستقبال ، وواجهات الذاكرة ، وشاشات العرض وغيرها الكثير فتمكنك من تنفيذ التطبيقات المختلفة وتجربتها على متحكمات stm32 دون الحاجة لوصل عناصر خارجية مع اللوحة بالإضافة لكونها تحتوي على مبرمجة ST-Link مدمجة معها تمكنك من إرسال الكود للمتحكم وتفحص الأخطاء من خلال الـ debug.



الشكل(11): STM32 Eval boards

### 4- موديولات Hexabitz:

- موديولات Hexabitz هي منصة نماذج الكترونية أولية (prototype platform) تضم عدد كبير من الموديولات (modules) والتي تأخذ عدة أشكال سداسي hexagon (من هنا جاءت تسمية الشركة)، خماسي pentagon، مستطيل، مربع ومثلث، وكل من هذه الموديولات يحتوي على متحكم مصغر من نوع (Cortex-M0 ARM) منخفض الطاقة للاتصال ومعالجة المهام المطلوبة من الموديول، أي يمكن استخدام أي موديول دون الحاجة إلى استخدام متحكم مصغر خارجي (microcontroller). كما يمكن تجميعها مع بعضها ضمن مصفوفة لتكون أقرب ما يمكن للشكل النهائي للدائرة المطبوعة pcb للنظام المراد تصميمه من ناحية الشكل والوزن والحجم.
- تقوم المتحكمات المصغرة microcontrollers الموجودة في الموديولات بربط هذه الموديولات ببعضها عن طريق شبكة سلكية لامركزية wired-mesh network وليتم تنفيذ المهام على التوازي بدلاً من تنفيذها من قبل متحكم (مايكروكونترولر) واحد مركزي.



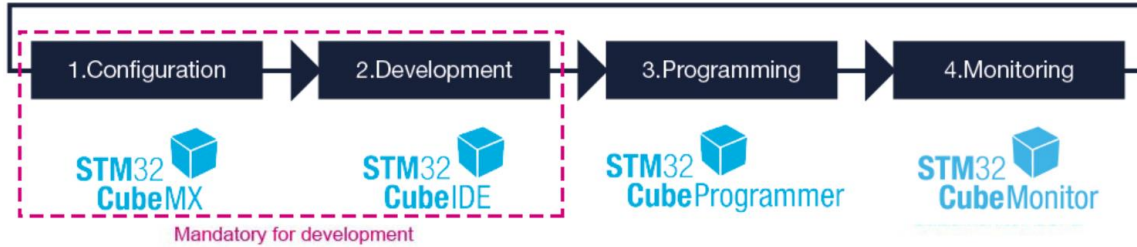
الشكل (12): موديلات HexabitZ

## 2.4. كيفية برمجة متحكمات STM32:

### 1- Software Tools:

لبرمجة متحكمات STM32 تحتاج إلى حزمة برامج تقدمها شركة STM مجاناً لتقوم بتنصيبها على جهاز الحاسب أهمها:

- STM32CubeIDE
- STM32CubeMX
- STM32CubeMonitor
- X-CUBE-MCSDK...



الشكل (13): Software Tools

أيضاً متحكمات stm32 متوافقة مع كلٍ من :

- ARM Mbed Ecosystem
- ARDUINO
- MicroPython

### 2- STM32Cube firmware:

تعتبر بيئة STM32Cube عبارة عن مجموعة من الأدوات والبرمجيات المدمجة والمقدمة مجاناً من شركة stm32 لتتيح للمبرمج سهولة وسرعة العمل حيث تتضمن على:

- HAL(High Abstraction Layer)

- LL(Low Layer)
- Examples
- Readme files

### -3 Hardware tools:

يتم ذلك من خلال طريقتين أساسيتين:

- 1 من خلال كبل الاتصال التسلسلي FTDI
- 2 من خلال المبرمجة الخاصة بمتحكمات STM32 (ST-LINK) التي تقوم بمهمة نقل الكود للمتحكم وأيضاً تسمح بمراقبة وتتبع الأخطاء البرمجية من خلال نافذة الـ debug.



الشكل (14): ST-Link