Tìm hiểu và hướng dẫn lập trình chip Renesas

Nội dung

[1. Mục đích tài liệu 2](#_Toc84065485)

[2. Giới thiệu về Renesas 2](#_Toc84065486)

[3. Hướng dẫn lập trình cho dòng chip RA Arm Cortex-M MCUs 3](#_Toc84065487)

[3.1. Giới thiệu về dòng chip Renesas 3](#_Toc84065488)

[3.2. Công cụ 7](#_Toc84065489)

[3.3. Hướng dẫn viết chương trình bằng e2 studio 8](#_Toc84065490)

[a. Hướng dẫn tạo project 8](#_Toc84065491)

[b. Hướng dẫn cấu hình và thêm driver cho MCUs 10](#_Toc84065492)

[c. Cấu trúc các file và thư mục của 1 project 12](#_Toc84065493)

[d. Hướng dẫn viết chương trình với các Example 13](#_Toc84065494)

[e. Một số lưu ý khi dùng phần mềm: 20](#_Toc84065495)

# Mục đích tài liệu

* Giới thiệu về các dòng chip của hãng Renesas
* Hướng dẫn lập trình trên chip RA Renesas, cụ thể là KIT RA-6M3 với các tính năng cơ bản:
* Điều khiển đèn LED qua GPIO và Timer
* Đọc trạng thái phím bấm qua ngắt ngoài
* Giao tiếp với PC qua UART
* Điều khiển text LCD 16x2 qua I2C
* Lập lịch thời gian thực cho bộ RTC
* Cài đặt Watchdog
* Sử dụng hệ điều hành thời gian thực FreeRTOS

# Giới thiệu về Renesas

Là công ty đến từ Nhật Bản thành lập từ năm 2002, với các sản phẩm về vi điều khiển, IC tương tự, IC nguồn, SoC đáp ứng các giải pháp trong lĩnh vực tự ô tô, công nghiệp, hệ thống điện toà nhà, văn phòng, …

website: [Renesas Electronics Corporation](https://www.renesas.com/us/en)

Youtube: [RenesasPresents - YouTube](https://www.youtube.com/user/RenesasPresents)

Các sản phẩm và dòng chip chính của Renesas (hình ảnh)

Graphical user interface, text, application, email

Description automatically generated

Trong hướng dẫn này chúng ta sẽ tập trung vào dòng RA Arm Coterx-M MCUs

So sánh với hãng ST:

* Các dòng MCUs của Renesas chủ yếu phục vụ cho thị trường trong nước (Nhật Bản) và Ấn Độ.
* Tài liệu hướng dẫn và các nhóm từ cộng đồng quốc tế không nhiều.
* Với thị trường Việt Nam tương đối ít người dùng, các trao đổi trên các diễn đàn, nhóm còn hạn chế.
* Các công cụ hỗ trợ cho việc lập trình, example chưa được phong phú và thân thiện bằng hãng ST

# Hướng dẫn lập trình cho dòng chip RA Arm Cortex-M MCUs

## Giới thiệu về dòng chip Renesas

Cũng như các hãng sản xuất chip khác, Renesas có nhiều dòng chip lõi Arm để chúng ta có thể lựa chọn tùy vào mục đích sản phẩm.

* **RA Arm Cortex-M**:

Các chip vi điều khiển lõi Arm cortex-M dùng cho các ứng dụng IOT, điều khiển… Đây là dòng nâng cao với nhiều Advanced.

* **Rz Arm-base High-end 32&64-bit MPUs**:

Các chip vi xử lí Arm Cortex-A và R thường dùng cho các ứng dụng high–resolution như máy tính nhúng, các ứng dụng AI…

* **RE Cortex-M ultra-low Power SOTB MCUs**:

Các chip Arm cortex-M tuy nhiên có các ưu điểm vượt trội về các chế độ tiết kiệm năng lượng, phù hợp cho các ứng dụng liên quan IOT như các thiết bị đeo tay, giám sát, theo dõi trong lĩnh vực sức khỏe và nông nghiệp…

* **RX 32-bit Performance/Efficiency MCUs**:

Với đa dạng các series từ thấp đến cao thiên về hiệu năng xử lí phục vụ nhiều tính năng như điều khiển động cơ, cảm ứng, mạng công nghiệp, giao diện đồ họa (GUI), đo lường…

(Lưu ý dòng này không phải lõi Arm và đa dạng hướng tới nhiều ứng dụng cụ thể).

* **Renesas Sysnergy Platform MCUs**:

Đây là dòng cơ bản với các chip lõi Arm cortex-M0, M4 phù hợp với các ứng dụng không yêu cầu cao về hiệu năng xử lí.

Ngoài ra còn có thêm nhiều dòng khác có thể tìm hiểu thêm tại web site của hãng.

Tài liệu sẽ tập trung vào các họ của dòng chip RA (Renesas Advanced)

[RA Family of Arm® Cortex®-M based MCUs | Renesas](https://www.renesas.com/us/en/products/microcontrollers-microprocessors/ra-cortex-m-mcus)

“The flexible **Renesas Advanced** (RA) 32-bit microcontrollers (MCUs) are industry-leading 32-bit MCUs with the **Arm® Cortex®-M33, -M23 and -M4** processor cores and PSA certification. RA delivers key advantages compared to competitive Arm Cortex-M MCUs by providing stronger embedded security, superior CoreMark® performance and ultra-low power operation. PSA certification provides customers the confidence and assurance to **quickly deploy secure IoT endpoint and edge devices, and smart factory equipment** for Industry 4.0.”

A picture containing diagram

Description automatically generated

Chúng ta sẽ tập trung vào RA6 dựa trên phần cứng là KIT EK-RA6M3

[EK-RA6M3 - Evaluation Kit for RA6M3 MCU Group | Renesas](https://www.renesas.com/br/en/products/microcontrollers-microprocessors/ra-cortex-m-mcus/ek-ra6m3-evaluation-kit-ra6m3-mcu-group)

A close-up of a computer chip

Description automatically generated with medium confidence

Tìm hiểu về chip RA6M3 với datasheet : [RA6M3 Microcontroller Group Datasheet (renesas.com)](https://www.renesas.com/us/en/document/dst/ra6m3-microcontroller-group-datasheet)

Một số tính năng nổi bật:

* Lõi Arm cortex-M4 (Revision: r0p 1-01 rel0, ARMv7E-M)
* Tần số hoạt động tối đa: 120 Mhz
* Bộ nhớ lên tới 2-MB flash và 640-KB SRAM
* Hỗ trợ đầy đủ các ngoại vi Ethernet, USB, , I2C, SPI, UART, ADC, LCD, Timer, RTC … ngoài ra còn có các tính năng khác như 2D Drawing Engine, Security and Safety.
* Hỗ trợ tối đa 133 chân chức năng Input/Output

Một số tính năng đi kèm với Module (**Table 1.1** tài liệu [RA6M3 Microcontroller Group Datasheet (renesas.com)](https://www.renesas.com/us/en/document/dst/ra6m3-microcontroller-group-datasheet))

Lưu ý đây là tên gọi và mô tả các tính năng chính của MCUs đồng thời được sử dụng trong phần mềm e2studio để thiết lập các tính năng này.

Một số tính năng chính:

|  |  |
| --- | --- |
| Interrupt Control Unit (ICU) | Liên quan đến lựa chọn các sự kiện sẽ liên kết với ngắt nào ví dụ như ngắt ngoài EXT |
| Asynchronous General-Purpose Timer (AGT) | Là bộ timer 16 bit thiết lập các tính năng cơ bản |
| Realtime Clock (RTC) | Thời gian thực |
| Serial Communications Interface (SCI) | Dùng thiết lập cho chuẩn truyền thông nối tiếp như UART, SPI, I2C |
| Watchdog Timer (WDT) | Dùng thiết lập watchdog |
| Clocks | Các nguồn dao động cho MCUs |

Các tên viết tắt này được dùng trong tab Stacks của phần mềm e2studio để thêm tính năng cho MCUs bằng thư viện HAL (Cụ thể sẽ được trình bày ở mục 3 và trong các ví dụ cho các Module cụ thể) như hình dưới.

Graphical user interface, application, Word

Description automatically generated

## Công cụ

**e2 studio** (editor, compiler, debuger)

Được phát triển lại từ Eclipse nên hoàn toàn miễn phí với nhiều tính năng hỗ trợ (khá tương đồng với Stm32CUBEIDE dùng cho ST) bao gồm editor, compiler and linker control, debug (GNU extended).

Bên cạnh đó e2 studio cung cấp gói phần mềm linh hoạt - Flexible Software Packet (FSP), giúp lập trình viên dễ dàng sử dụng, mở rộng và nâng cao hiệu quả chương trình.

Tích hợp giao diện cấu hình, lựa chọn module, tự động tạo code thuận tiện cho lập trình viên

Có thể tích hợp export sourcecode sang các phần mềm khác như KeilC hoặc IAR.

Tài liệu:

[e² studio | Renesas](https://www.renesas.com/us/en/software-tool/e-studio)

[Getting Started Guide for e2 studio for RA (renesas.com)](https://www.renesas.com/br/en/document/mat/getting-started-guide-e2-studio-ra?language=en)

FSP v3.3.0 : [Release v3.3.0 · renesas/fsp · GitHub](https://github.com/renesas/fsp/releases/tag/v3.3.0)

[Flexible Software Package (FSP) | Renesas](https://www.renesas.com/us/en/software-tool/flexible-software-package-fsp)

Diagram, timeline

Description automatically generated

Hướng dẫn cài đặt e2studio

Tất cả có tại đây Mục 2: [Getting Started Guide for e2 studio for RA (renesas.com)](https://www.renesas.com/br/en/document/mat/getting-started-guide-e2-studio-ra?language=en)

## Hướng dẫn viết chương trình bằng e2 studio

### Hướng dẫn tạo project

Tham khảo mục 3.Project Generation [Getting Started Guide for e2 studio for RA (renesas.com)](https://www.renesas.com/br/en/document/mat/getting-started-guide-e2-studio-ra?language=en)

Lưu ý: Chọn MCUs/Board (Board hiện tại là **EK-RA6M3** hoặc chọn Device **R7FA6M3AH3CFC**)

Graphical user interface, application

Description automatically generated

Sau khi kết thúc tạo Project, sẽ xuất hiện cửa sổ FSP Configuration cho phép cấu hình các chân, ngắt, clock và thêm các driver cần thiết (hình dưới)

Graphical user interface, text, application

Description automatically generated

### Hướng dẫn cấu hình và thêm driver cho MCUs

Chi tiết xem Mục 3.5 RA Project Configuration Editor

Một số các tab chính:

* **Summary**:

Thông tin tổng quan về Project (board, chip, software packet …). Đây là tab tổng hợp thông tin các tab còn lại.

Graphical user interface, text, application

Description automatically generated

* **BSP (Board Support Packet):**

Tại đây là các thông số chung liên quan đến MUCs và một số cài đặt. Khi click vào tab BSP chọn tiếp tab Properities (Nếu không có tab này vào Window -> Show View -> Properites) để xem chi tiết. Tương tự cho các tab còn lại.

Ở tab này lưu ý các tham số như hình dưới để điều chỉnh phù hợp

Heap size (bytes) có thể khai báo 0x200 hoặc 0x400

Các tham số Parameter Checking, Soft Reset nên để Enable

Graphical user interface, text, application

Description automatically generated

**Clocks**: Cài đặt nguồn clock và tần số cho các bus ngoại vi

**Pins**:

* + Vào Ports để thiết lập tính năng cho các các chân nếu muốn chúng như là GPIO thông thường.
  + Vào Peripherals nếu muốn thiết lập tính năng cho một chân nào đó liên quan đến các tính năng như ngắt, uart, spi …
  + Other Pins: Đây là các chân đặc biệt như đầu vào clock, chân nguồn …

Graphical user interface, application, table

Description automatically generated

* **Interrupts và Event Links**: Chưa cần quan tâm, để mặc định
* **Stacks**: Tab này sẽ thực hiện tự động tạo thư viện cho các module bao gồm driver, thread, semaphore… (nếu dùng RTOS)

Ví dụ đây là việc thêm Uart driver cho project: Chọn New Stack -> Driver -> …

Chi tiết về lựa chọn các driver và cấu hình tham số tham khảo phần Example phía sau.

Graphical user interface, application, Word

Description automatically generated

### Cấu trúc các file và thư mục của 1 project

Khi 1 chương trình mới tạo xong sẽ tương tự như hình.

Graphical user interface, text, application

Description automatically generated

Project sẽ bao gồm các thư mục bên trái hình:

* “ra” chứa các driver liên quan đến MCUs, các driver thêm mới sẽ nằm trong ra/fsp
* “ra\_gen”: chứa các file thông tin dữ liệu cho MCUs. Những file này sẽ bị thay đổi khi Genarate Project Content. Do đó không edit các file ở đây. Ngay cả file main.c
* “src”: đây là nơi cho phép thêm các file ứng dụng và edit code.

Như trên hình ảnh có thể thấy trong hàm main sẽ gọi hàm hal\_entry và chương trình chúng ta sẽ bắt đầu trong hàm này. (nếu không thêm RTOS vào project)

Graphical user interface, text, application

Description automatically generated

Các thư mục còn lại không cần phải quan tâm.

### Hướng dẫn viết chương trình với các Example

* Non-RTOS

Khai báo chung:

Sau khi tạo project mới theo các bước hướng dẫn ở trên. Trước khi bắt đầu chúng ta cài đặt một số thông số cơ bản chung cho tất cả các project như sau:

* + Khai báo heap và enable một số tính năng

Graphical user interface, application

Description automatically generated

* + Chọn nguồn dao động cho MCUs

Graphical user interface, text, application

Description automatically generated

* + Chọn kiểu chân nạp code và debug

Graphical user interface, text, application

Description automatically generated

Các module cơ bản:

* **GPIO**

Trước hết mở mạch nguyên lí của KIT-RA6M3 (download từ website hoặc file “ek-ra6m3-v1-schematics“ đính kèm)

Xác định các chân LED. Ví dụ LED\_USER\_R nối với chân P100

Chart

Description automatically generated

Đây là chức năng GPIO do đó vào tab Pins -> Port thiết lập

Graphical user interface, text, application, email

Description automatically generated

Sau khi Generate quay lại chế độ C/C++ để xem source codeGraphical user interface, text, application, Word

Description automatically generated

Lúc này chương trình sẽ thêm các file code được tự động tạo ra vào các folder “ra” và “ra\_gen”.

Graphical user interface, text, application, chat or text message

Description automatically generatedThêm thư viện mới của chúng ta nếu cần. Ví dụ ở đây là myLib với 2 file myGPIO.h và myGPIO.c (lưu ý thêm include đường dẫn cho project tới thư mục myGPIO trước khi dùng)

Tiếp theo vào file hal\_entry.c và tìm hàm **void** **hal\_entry**(**void**),

Chương trình sẽ bắt đầu từ đây.

* Graphical user interface, application

  Description automatically generatedVới External Interrupt:

Kích chuột phải vào chân và chọn tính năng IRQ12

Graphical user interface

Description automatically generated with medium confidence

Graphical user interface, application, table

Description automatically generated

Thêm driver (New Stack -> Driver -> Input -> Extenal IRQ)

Đặt tên và chọn channel tương ứng, khai báo

hàm callback

Graphical user interface, text, application, email

Description automatically generated

* Các thiết lập cho các dirver khác như TIMER, RTC, UART, I2C tương tự tham khảo code mẫu
* FreeRTOS

Để sử dụng RTOS trong quá trình tạo project thêm RTOS

Graphical user interface, application, Word

Description automatically generated

Với các bước thêm driver tương tự, để thêm Thread tab Stack chọn New Thread. Ngoài ra ở mục Object có thể thêm Semaphore, Mutex, Queue …

Graphical user interface, text, application

Description automatically generated

Đặt tên cho Tread và lưu ý ở các tham số chung cho Thread nên để enable Use Time Slicing để cho phép chuyển đổi ngữ cảnh giữa các Thread khi hết time slice.

Graphical user interface, application

Description automatically generated

Sau khi click vào Generate Project Content, e2 studio sẽ tạo ra ở folder “src” các file riêng cho từng Thread mới tạo. Và chương trình của chúng ra sẽ viết vào các file này (file hal\_entry.c sẽ không được sử dụng).

Text

Description automatically generated

### Một số lưu ý khi dùng phần mềm:

* + Điều chỉnh lại việc tối ưu code, để mức thấp nhất cho việc debug dễ dàng hơn

Graphical user interface, text, application, email

Description automatically generated

* + Theo dõi các biến một cách realtime

(Kích chuột phải vào biến, chọn Enable Real-Time Refresh,)

Graphical user interface, application, table, Excel

Description automatically generated

* Chuyển đổi giữa các giao diện

**C/C++:** Giao diện viết chương trình

**FSP Configuration**: Giao diện cấu hình

**Debug**: Giao diện debug

Graphical user interface, text, application

Description automatically generated