BÁO CÁO THỰC HÀNH BÀI 2

Môn học: **CHUYÊN ĐỀ THIẾT KẾ HỆ THỐNG NHÚNG 1**- Mã lớp: **CE437.N11**

Giảng viên hướng dẫn thực hành: Phạm Minh Quân

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Thông tin sinh viên** | |  |  | | --- | --- | | MSSV | Họ và tên | | 20520211 | Trương Hữu Khang | | 20520219 | Nguyễn Linh Anh Khoa | | 20520597 | Phan Duy Thông | |
| **Link các tài liệu tham khảo** *(nếu có)* |  |
| **Đánh giá của giảng viên**:  *+ Nhận xét*  *+ Các lỗi trong chương trình*  *+ Gợi ý* |  |

*[Báo cáo chi tiết các thao tác, quy trình sinh viên đã thực hiện trong quá trình làm bài thực hành. Chụp lại hình ảnh màn hình hoặc hình ảnh kết quả chạy trên sản phẩm. Mô tả và giải thích chương trình tương ứng để cho ra kết quả như hình ảnh đã trình bày. Sinh viên xuất ra file .pdf và đặt tên theo cấu trúc: MSSV\_HoTen\_Labx\_Report.pdf (Trong đó: MSSV là mã số sinh viên, HoTen là họ và tên, x trong Labx là chỉ số của bài thực hành tương ứng]*

**MỤC LỤC**

[Câu 1. Tìm hiểu UART 3](#_Toc119867763)

[Câu 2. Viết chương trình để thực hiện đọc dữ liệu từ UART: 4](#_Toc119867764)

[Câu 3. Bài tập thực hành 6](#_Toc119867765)

1. Tìm hiểu UART

Giới thiệu UART:

* UART là Universal Asynchronous Reciver/Transmister một chuẩn giao tiếp không đồng bộ cho MCU và các thiết bị ngoại vi.
* Chuẩn UART là chuẩn giao tiếp điểm và điểm, nghĩa là trong mạng chỉ có hai thiết bị đóng vai trò là transmister hoặc reciver.

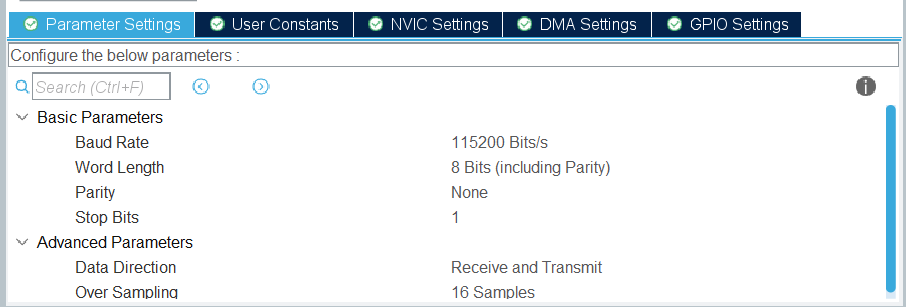
Cách hoạt động của giao thức UART

* UART là giao thức truyền thông không đồng bộ, nghĩa là không có xung Clock, các thiết bị có thể hiểu được nhau nếu các Setting giống nhau
* UART là truyền thông song công(Full duplex) nghĩa là tại một thời điểm có thể truyền và nhận đồng thời.
* Trong đó quan trọng nhất là Baund rate (tốc độ Baund) là khoảng thời gian dành cho 1bit được truyền. Phải được cài đặt giống nhau ở gửi và nhận.
* Sau đó là định dạng gói tin.

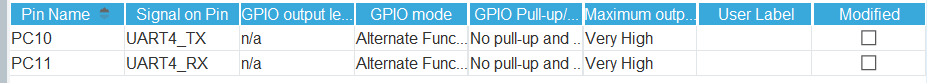
Định dạng gói tin như sau:

* Start – Bit
  + Start-bit còn được gọi là bit đồng bộ hóa được đặt trước dữ liệu thực tế. Nói chung, một đường truyền dữ liệu không hoạt động được điều khiển ở mức điện áp cao. Để bắt đầu truyền dữ liệu, truyền UART kéo đường dữ liệu từ mức điện áp cao (1) xuống mức điện áp thấp (0). UART thu được thông báo sự chuyển đổi này từ mức cao sang mức thấp qua đường dữ liệu cũng như bắt đầu hiểu dữ liệu thực. Nói chung, chỉ có một start-bit.
* Stop – Bit
  + Bit dừng được đặt ở phần cuối của gói dữ liệu. Thông thường, bit này dài 2 bit nhưng thường chỉ sử dụng 1 bit. Để dừng sóng, UART giữ đường dữ liệu ở mức điện áp cao.
* Partity Bit
  + Bit chẵn lẻ cho phép người nhận đảm bảo liệu dữ liệu được thu thập có đúng hay không. Đây là một hệ thống kiểm tra lỗi cấp thấp & bit chẵn lẻ có sẵn trong hai phạm vi như Chẵn lẻ – chẵn lẻ cũng như Chẵn lẻ – lẻ. Trên thực tế, bit này không được sử dụng rộng rãi nên không bắt buộc.
* Data frame
  + Các bit dữ liệu bao gồm dữ liệu thực được truyền từ người gửi đến người nhận. Độ dài khung dữ liệu có thể nằm trong khoảng 5 & 8. Nếu bit chẵn lẻ không được sử dụng thì chiều dài khung dữ liệu có thể dài 9 bit. Nói chung, LSB của dữ liệu được truyền trước tiên sau đó nó rất hữu ích cho việc truyền.

1. Viết chương trình để thực hiện đọc dữ liệu từ UART:

Cấu hình thông số cho UART

Cấu hình pin cho UART dựa trên Schematic của mạch



Graphical user interface

Description automatically generated

Hàm đọc dữ liệu từ UART

HAL\_UART\_Receive(UART\_HandleTypeDef \*huart, uint8\_t \*pData, uint16\_t Size, uint32\_t Timeout)

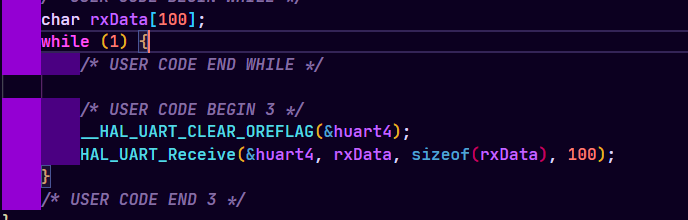
huart: UART sử dụng để đọc dữ liệu

pData: pointer trỏ vào vùng nhớ để lưu dữ liệu nhận được từ UART

Size: kích thước của dữ liệu (bytes)

Timeout: thời gian chờ

Chương trình đọc dữ liệu từ UART



Giải thích:

Hàm \_UART\_HAL\_CLEAR\_OREFLAG dùng để xóa các giá trị bị thừa ở lần input trước.

Hàm HAL\_UART\_Receive dùng để đọc dữ liệu từ UART

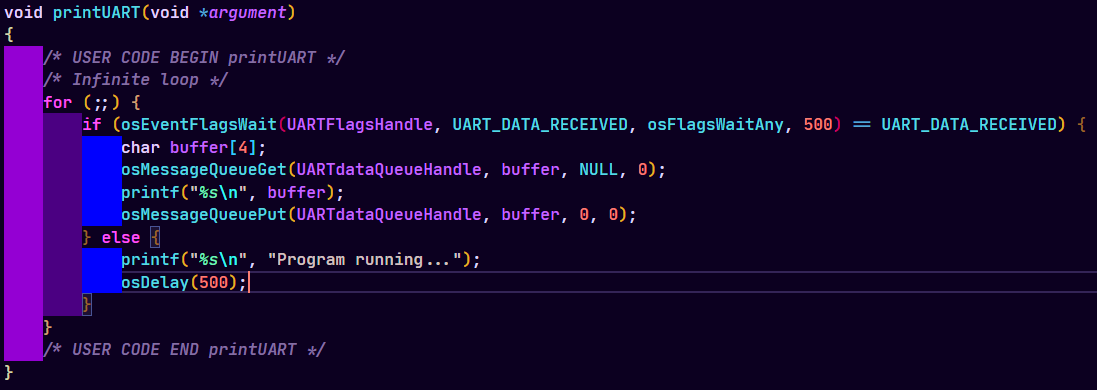
1. Bài tập thực hành

Tác vụ 1: đọc các dữ liệu nhận được từ UART

A screenshot of a computer

Description automatically generated

Tác vụ 2: Xuất ra dữ liệu đã nhận từ UART, xuất ra theo chu kỳ 500ms bản tin: “Program Runninng…”



**Tác vụ 3: Xử lý dữ liệu nhận được từ UART và điều khiển các LED theo dữ liệu đã được nhận từ UART**

Text

Description automatically generated

Cấu trúc chương trình:

Diagram

Description automatically generated

Cách thức hoạt động: chương trình sử dụng biến cờ để thông báo trạng thái hiện tại của chương trình.

Sau khi UART nhận được dữ liệu sẽ thêm dữ liệu vào Queue và bật cờ UART\_DATA\_RECEIVED để thông báo cho tác vụ printUART để đọc dữ liệu từ Queue. Sau khi việc xuất dữ liệu ra UART hoàn tất, trả dữ liệu lại vào Queue và bật cờ UART\_PRINTED để tác vụ controlLEDs xử lý.