

## **BÁO CÁO TIẾN ĐỘ LẦN I – 10/07/2024**

**Họ và tên:** Lê Trọng An

**MSSV:** 21010389.

**Nhóm:** 01;

**Lớp:** Kỹ thuật vi xử lý vi điều khiển N06.

**Phần việc được phân công:** Lập trình giao tiếp UART, lập trình điều khiển dịch và phát mã Morse.

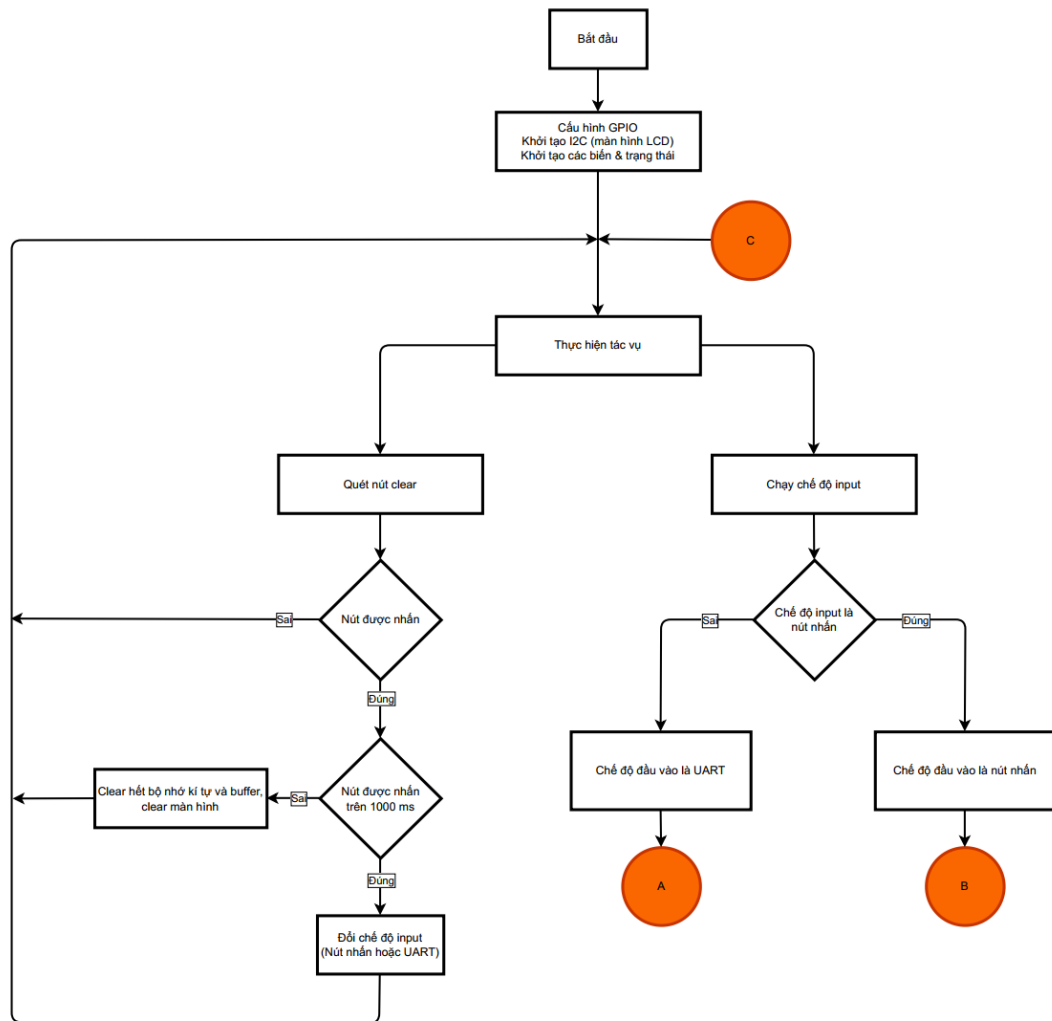
**Tiến độ công việc:** Đã hoàn thành.

### **1. Mô tả cách thức hoạt động của hệ thống**

Nhóm xây dựng yêu cầu hệ thống với các mục tiêu như sau:

- Mạch có thể dịch và phát mã Morse theo 2 chế độ input: Chế độ nhập bằng nút nhấn và chế độ nhập qua UART.
- Có thể chuyển chế độ bằng một nút nhấn trong 6 nút có trên PCB.
- Ở chế độ nút nhấn, hệ thống cần những nút sau:
  - Nút nhập liệu mã “.” hoặc “-” bằng cách nhấn nút thời gian ngắn hoặc dài
  - Nút xác nhận mã đã nhập để dịch sang kí tự.
  - Nút phát âm mã đã dịch bằng buzzer.
  - Nút phát âm mã đã dịch bằng LED.
  - Nút xoá toàn bộ âm mã đã dịch (nút clear).
- Ở chế độ UART:
  - Cần phát hiện và đọc data ở UART1 và hiện lên màn hình.
  - Cần phát âm kí tự bằng buzzer khi đọc được lệnh “b”.
  - Cần phát âm kí tự đã đọc bằng LED khi đọc được lệnh “l”.
  - Cần xoá toàn bộ âm kí tự đã đọc khi đọc được lệnh “c”.

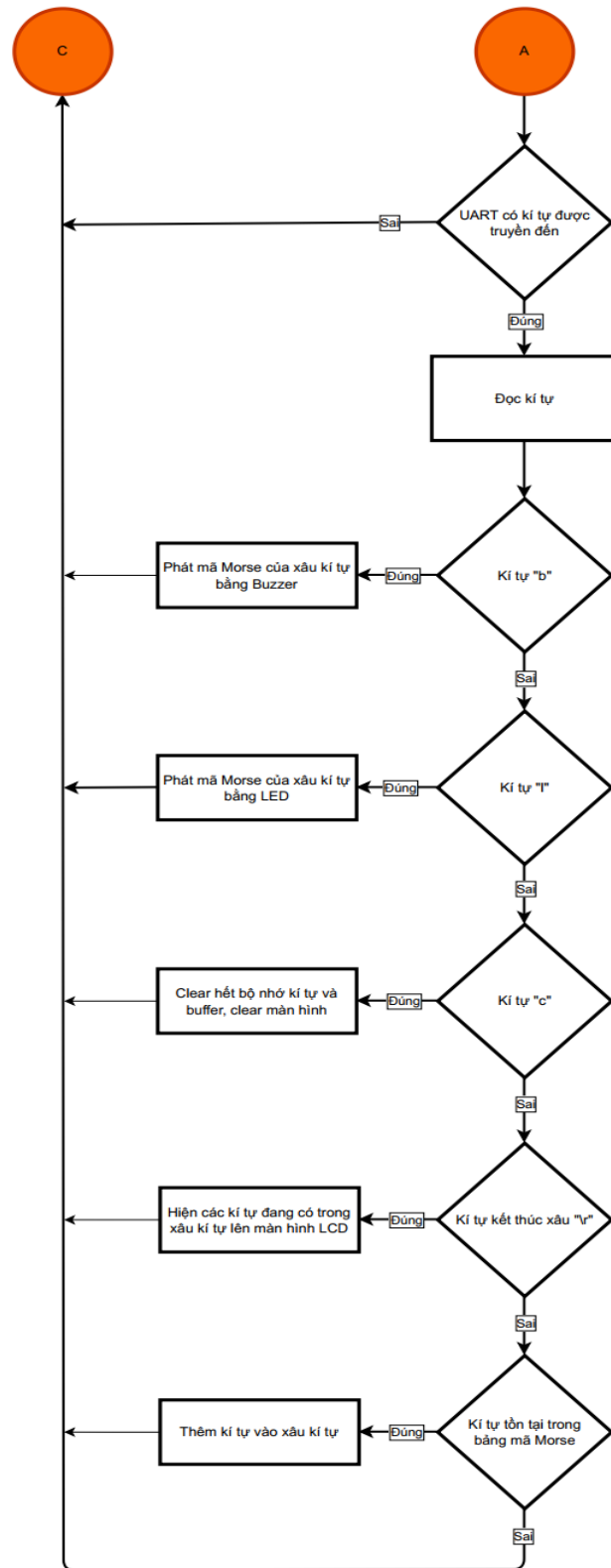
Từ các yêu cầu trên, em đã xây dựng flowchart cho hệ thống như sau:



Hình 1: Lưu đồ thuật toán cho toàn hệ thống

Thuật toán xử lý khi chế độ đầu vào là UART:

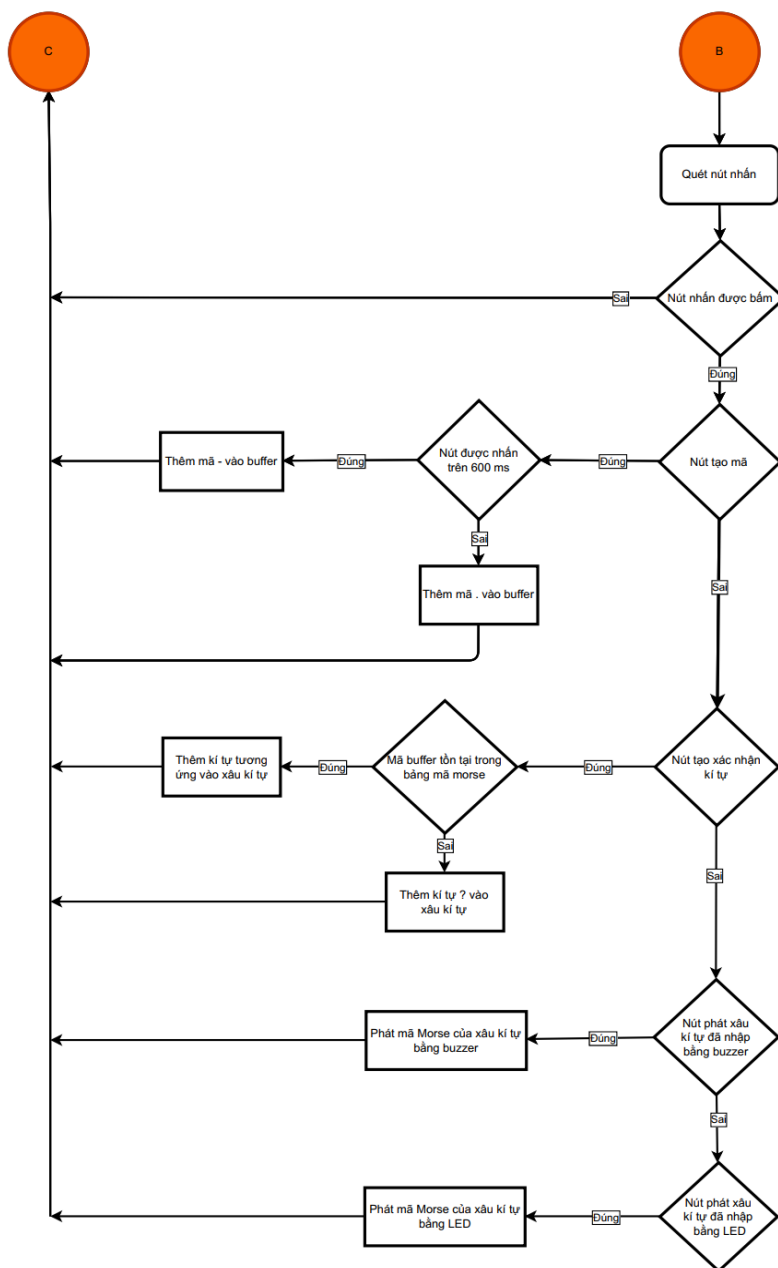
Khi nhận được dữ liệu từ máy tính, vi điều khiển sẽ tiến hành đọc và hiện thị dữ liệu nhận được lên màn hình LCD. Sau đó khi nhận được kí tự “b” sẽ thực hiện phát mã bằng buzzer, nếu kí tự nhận được là “l” thì sẽ phát mã bằng đèn LED. Nếu nhận được kí tự “c” thì thực hiện xóa bộ nhớ và xóa nội dung hiển thị trên LCD. Khi nhận được kí tự không nằm trong bảng mã đã nhập sẵn, LCD sẽ hiển thị dấu “?”.



Hình 2: Lưu đồ thuật toán xử lý chế độ input UART

Thuật toán xử lý khi chế độ đầu vào là nút nhấn:

Ở chế độ đầu vào là nút nhấn, các nút sẽ được quét liên tục. Khi nhấn nút tạo mã, nếu nhấn giữ trên 0,6 giây thì sẽ tạo 1 kí tự dài khi nhấn nhả hoặc nhấn giữ dưới 0,6 giây sẽ tạo 1 mã ngắn. Sau khi nhập mã, nhấn nút xác nhận để lưu vào bộ nhớ. Sau đó nếu nút chọn kiểu phát mã bằng buzzer được kích hoạt thì thực hiện phát mã đã được lưu bằng buzzer, nếu nút chọn kiểu phát bằng đèn được kích hoạt thì thực hiện phát mã qua đèn LED.



Hình 3: Lưu đồ thuật toán chế độ input nút nhấn

## 2. Mô tả cách thức hoạt động của các file code.

Mục tiêu khi em viết mã nguồn là chia nhỏ các phần xử lý thành các thư viện nhỏ theo chức năng. Việc module hoá như vậy sẽ giúp project dễ quản lý, nâng cấp, clean code.

Folder chứa mã nguồn của project gồm các file sau:

- main.c và main.h: Lần lượt là file source và file header chứa phần xử lý chính của project.
- lcd\_i2c: Thư viện để giao tiếp i2c với module PCF8574A của màn LCD 16x2.
- globalVariables.h: Là file header chứa toàn bộ các biến và hàm cần thiết trong cả project. Các thư viện con (module) sẽ đều khai báo thư viện này, từ đó sẽ không cần quan tâm đến việc truyền, nhận, trả biến khi sử dụng các hàm ở các thư viện khác nhau, giúp tiết kiệm ram, tối ưu tác vụ và thuận tiện cho việc lập trình.
- morseCodeReader.c: Thư viện em tự viết, chứa các phần mã xử lý cho phần đọc mã Morse. Vì thư viện này sử dụng các biến toàn cục ở file globalVariables.h và để thuận tiện khi sử dụng nên không cần thiết phải có file header kèm theo (morseCodeReader.c). Để sử dụng chỉ cần khai báo luôn file source này trong main.h.
- morseCodeSpeaker.c: Thư viện em tự viết, chứa các phần mã xử lý cho phần phát mã Morse bằng buzzer hoặc LED. Vì thư viện này sử dụng các biến toàn cục ở file globalVariables.h và để thuận tiện khi sử dụng nên không cần thiết phải có file header kèm theo (morseCodeSpeaker.c). Để sử dụng chỉ cần khai báo luôn file source này trong main.h.