# Direct Memory Access (DMA)

**Project 3**

Prepared for

CSE 410 Advanced Topics in Embedded System Design

By

Steven Collins (sac9)

October 14, 2024

# Table of Contents

**Section 1: Division of Work..... 2**
**Section 2: Program Overview.....3**
➔ Program Overview..... 3
➔ Program Summary..... 3
➔ High Level Flowchart..... 4
**Section 3: Subroutine Descriptions..... 5**
➔ determine_keypad_button.....5
**Section 4: Subroutine Flowcharts..... 6**
➔ determine_keypad_button Flowchart.....6
**Section 5: Hypothesis.....7**

# Section 1: Division of Work

I, Steven Collins, wrote the entirety of the dma project. Some subroutines were used from CSE 379, on which my partner Frank Peretti and I worked together to implement.

# Section 2: Program Overview

## ➔ Program Overview

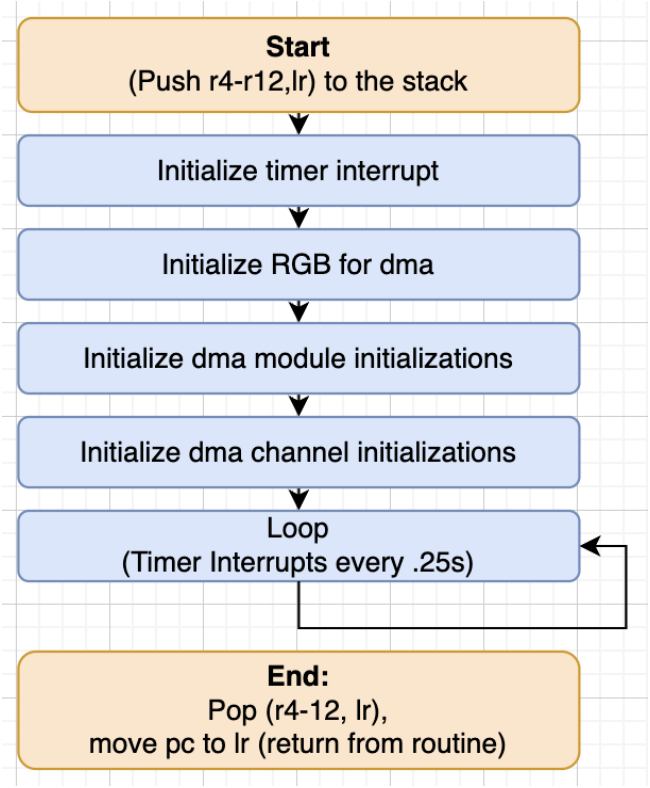
- Run Program
- Dma begins
- Pause program shortly after seeing white light, and watch dma continue to completion while the program is paused

## ➔ Program Summary

This direct memory access program causes the RGB LED to alternate red and blue. Once the dma completes (16 bytes), it is rerun by a timer interrupt. If the dma is in process and the program is paused, the dma continues to completion despite the processor being paused.

## ➔ High Level Flowchart

*An overview of the entire program*



# Section 3: Subroutine Descriptions

*Description of each subroutine's functionality, the arguments it passes through, and what the return values in each register are.*

## ➔ dma_module_init

- Functionality:
  - This subroutine configures module initializations: enables clock, controller, and sets the base address of channel control.
- Arguments:
  - There are no argument registers for the functionality of this subroutine.
- Return registers:
  - There are no return registers for the functionality of this subroutine.

## ➔ dma_configure_channel

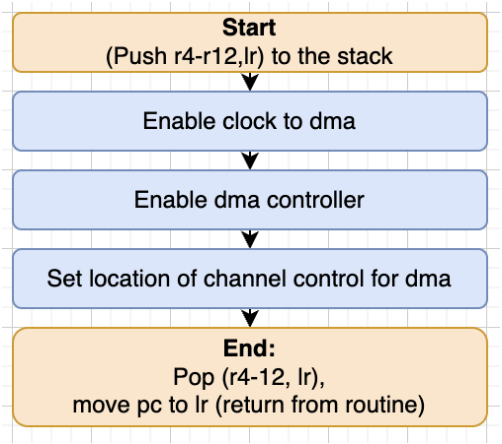
- Functionality:
  - This subroutine configures the channel structure: channel attributes and word structure
- Arguments:
  - There are no argument registers for the functionality of this subroutine.
- Return registers:
  - There are no return registers for the functionality of this subroutine.
  -

## ➔ Timer0A_Handler

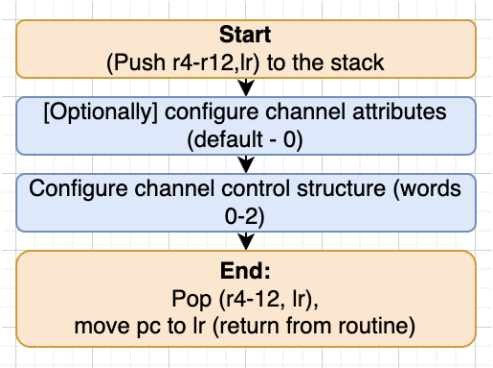
- Functionality:
  - This subroutine is called every .25 seconds. It sets the xfersize and xfermode of the dma channel structure, and then triggers the dma channel enable
- Arguments:
  - There are no argument registers for the functionality of this subroutine.
- Return registers:
  - There are no return registers for the functionality of this subroutine.

# Section 4: Subroutine Flowcharts

➔ dma_module_init Flowchart




➔ dma_configure_channel Flowchart




➔ Timer0A_Handler Flowchart