Politecnico Di Torino

Department of CONTROL AND COMPUTER ENGINEERING (DAUIN)

Master degree in Mechatronics Engineering   2020/2021

Operating System for Embedded Systems

<u>Assignment Report</u>

Dec 14, 2020

Submitted By:

- Sufian Saeed 277428

Submitted To:

- Massimo Violante

Contents

1   Objective

The goal of this Assignment is to implement a Morse-based Twitter using OSEK approach.

2   Software and Hardware

The Software used to develop the code for Morse-based Twitter is Eclipse with Erika Enterprise. The code is written to run on Arduino Uno Micro-controller.

3   Algorithm logic Explanation

3.1   .oil file

The code is implemented using one basic task. In order to activate this task every 100 milliseconds the Alarm is configured with hardware counter. The duration of each tick is set to 1ms and the increment of counter is set to 10 ticks. Means after each 10 ticks counter is incremented by 1. So Alarm is wake up after 10 counter increments in order to full filled the 100ms delay for each bit.

3.2   .cpp file

As described early that basic task is activated every 100ms, So in order to change the tweet message every 180 seconds, I increment the count variable every 100ms inside the basic task to reach it to 1800. So in this way only one Alarm is sufficient to switch message after each 180 seconds. When the count variable reached to 1800 the condition becomes true and the tweet change function is called, in which new message is copied to array.

Inside function Switch Case logic is implemented in order to copy next message every time the function is called.

When the tweet message is changed after 180s, a pause of 0.5s is implemented for this reason inside the 1800 count condition when its become true the msg-delay variable is set to 5, So the message display is interrupted for 500ms. So it is also done without adding another Alarm for it. After 500ms the message again start reading from zero index of the array. As every index is set to zero again.

While reading the array message the switch case logic is implemented in which if Dot '.' is read then single one is displayed in the form of LED. After each symbol single zero is displayed by switch off the LED. If dash '-' is read then three one's are displayed and after three one's single zero is displayed.

In order to give three zeros pause after each inter code word I inserted comma after each inter code word. So in logic I inserted two zeros for comma, because one zero is already displayed after each symbol automatically. So in this way three zeros constraint is full filled. Same logic is implemented for space, when the word is completed seven zeros is displayed. So in logic six zeros are implemented and one zero came from symbol pause.

Two extra functions are made in order to switch on and off the light. So in this way the whole logic for this assignment is implemented in a simple way.

4   Memory Consumption

5038 bytes of flash verified.

text:4116
data:922
bss:236
dec:5274