EECS 690
Project 1 Report

Program Profiling on a TI Tiva C TI_TM4C1294NCPDT using FreeRTOS

Written By
Ben Sokol
Kaiser Mittenburg

The University of Kansas
Electrical Engineering and Computer Science

September 18, 2018

1 Abstract

Many people write tasks for various computer systems every day without understanding how many resources that task will actually use. This report helps show how to calculate the amount of time each of those tasks will take, specifically on a TI_TM4C1294NCPDT board using FreeRTOS.

2 Revision History

The following table lists the revision history for this document.

Table 2-1 Revision History

Date	Revision	Description
September 18, 2018	1.0	Initial Release

3 Table of Contents

1 Abstract **2**
2 Revision History **2**
3 Table of Contents **3**
4 List of Figures **3**
5 List of Tables **3**
6 Principles of Operation (POP) **4**
7 Data Structure Descriptions **4**
8 Function Descriptions **4**
9 Parameters **4**
10 Testing **4**
11 Lessons Learned **4**
Appendix A – Program Source Code **6**

4 List of Figures

Table 2-1 Revision History **2**
Table 7-1 Data Structures **4**
Table 8-1 Functions **4**
Table 9-1 Parameters **4**

5 List of Tables

No table of figures entries found.

Principles of Operation (POP)

The POP shall describe the architecture and design of your project; it should start with an overall prose description of the project and goal and include a top-level block diagram; you should then describe your project in detail; you should list all modules you used or adapted from other sources

6 Data Structure Descriptions

Table 6-1 Data Structures

Data Structure Name	Type	Data Structure Description

7 Function Descriptions

Table 7-1 Functions

Function Name (Purpose)	Function Pseudo Code	Referenced External Functions
Task_ProgramTrace This task purpose.		

8 Parameters

Table 8-1 Parameters

Parameter Name	Parameter Type	Parameter Default Value	Parameter Description
Current_PC	uint32_t	0	Contains the current Program Counter.

9 Testing

10 Lessons Learned

Never write assembly, it is terrible…

Appendix A – Program Source Code