

Control over CAN and Flexray
Embedded Control Systems

Sai Krishna Kalluri & Snorri Stefansson

March 25, 2017

Contents

| | |
|---|----------|
| Introduction | 2 |
| 1 Part 1 | 3 |
| 1.1 Introduction | 3 |
| 1.2 Response Time analysis | 3 |
| 1.2.1 Response time analysis per processing unit | 3 |
| 1.2.2 Response time analysis for the CAN bus messages | 3 |
| 1.3 System model | 3 |
| 1.4 Design decision | 3 |
| 1.5 Results | 3 |
| 2 Part 2 | 4 |
| 2.1 Introduction | 4 |
| 2.2 Response Time analysis | 4 |
| 2.2.1 Response time analysis per processing unit | 4 |
| 2.2.2 Response time analysis for the CAN bus messages | 5 |
| 2.3 Optimisation for sensor-to-actuator delay | 5 |
| 2.4 System model | 5 |
| 2.5 Design decision | 5 |
| 2.6 Results | 5 |
| 2.7 Conclusions | 5 |
| 3 Part 3 | 6 |
| 3.1 Introduction | 6 |
| 3.2 Answer all the questions | 6 |
| 3.2.1 Theoretical analysis versus actual implementation | 6 |
| 3.3 Design decision | 6 |
| 3.4 Results | 6 |
| 3.5 Conclusions | 6 |
| 3.6 Results | 6 |
| 3.7 Conclusion | 6 |

Introduction

Chapter 1

Part 1

1.1 Introduction

1.2 Response Time analysis

1.2.1 Response time analysis per processing unit

Table 1.1: By running the Matlab script `ResponsetimeAnylsis.FPP.m` with the different parameters given for PU1 and PU2 these response times are obtained for each of the tasks. These files are then delivered as PU1.m PU2.m

| PU1 | T_1 | T_2 | T_3 | $T_4 (T_s)$ |
|-------------|-------|-------|-------|-------------|
| Matlab (ms) | 0.1 | 2.1 | 4.1 | 7.2 |

| PU2 | T_5 | T_6 | T_7 | T_8 |
|-------------|-------|-------|-------|-------|
| Matlab (ms) | 6 | 3 | 9 | 5 |

1.2.2 Response time analysis for the CAN bus messages

Table 1.2: Response times for the CAN bus messages

| CAN | m_2 | m_1 | m_3 | m_8 |
|-------------|-------|-------|-------|-------|
| Matlab (ms) | 2 | 3 | 4 | 4 |

1.3 System model

1.4 Design decision

1.5 Results

Firstly: Response time analysis

Secondly: Control system input and output

Chapter 2

Part 2

2.1 Introduction

2.2 Response Time analysis

2.2.1 Response time analysis per processing unit

Table 2.1: By running the Matlab script ResponsetimeAnylsis.FPP.m with the different parameters given for PU1 and PU2 these response times are obtained. These files are then delivered as PU1.m PU2.m

| | PU1 | T_1 | T_2 | T_3 | T_4 (T_s) |
|--------------|-----|-------|-------|-------|-----------------|
| Matlab (ms) | | 0.1 | 2.1 | 4.1 | 7.2 |
| Inchron (ms) | | 0.1 | 2.1 | 4.1 | 7.2 |

| | PU2 | T_5 | T_6 | T_7 | T_8 |
|--------------|-----|-------|-------|-------|-------|
| Matlab (ms) | | 6 | 3 | 9 | 5 |
| Inchron (ms) | | 6 | 3 | 9 | 5 |

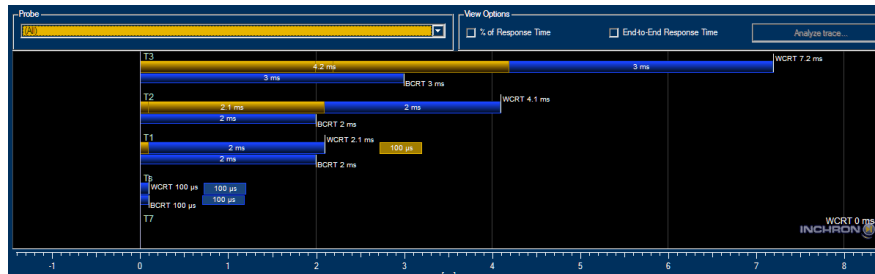


Figure 2.1:

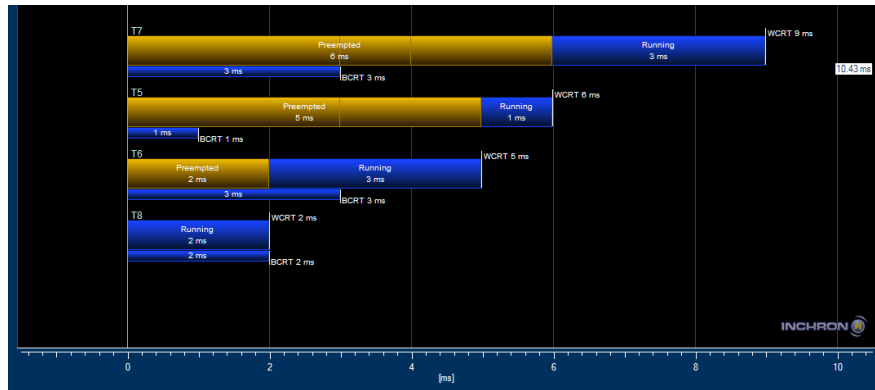


Figure 2.2:

2.2.2 Response time analysis for the CAN bus messages

2.3 Optimisation for sensor-to-actuator delay

2.4 System model

2.5 Design decision

2.6 Results

Firstly: Response time analysis

Secondly: Plots from chronVIEW (before and after optimization)

Last: Control system input and output

2.7 Conclusions

Chapter 3

Part 3

3.1 Introduction

3.2 Answer all the questions

3.2.1 Theoretical analysis versus actual implementation

3.3 Design decision

3.4 Results

Firstly: Solution to the design problem. (Include the parameters you have chosen)
Secondly: from chronVIEW for your design

3.5 Conclusions

3.6 Results

3.7 Conclusion