Contents

1 Introduction 1
- 1.1 Goals . . . 2
- 1.2 Dissertation Organization . . . 3

2 State of the Art 5
- 2.1 Background Concepts . . . 5
  - 2.1.1 What is an Embedded Linux System? . . . 5
  - 2.1.2 Boot Sequence . . . 7
    - 2.1.2.1 Hardware Initialization . . . 7
    - 2.1.2.2 Boot loader . . . 8
    - 2.1.2.3 Linux kernel . . . 9
    - 2.1.2.4 User Space . . . 11
- 2.2 Measurement Process . . . 12
  - 2.2.1 Boot time interval . . . 13
  - 2.2.2 Measurement Techniques . . . 13
    - 2.2.2.1 Serial Log Listening . . . 13
    - 2.2.2.2 GPIO toggling . . . 15
- 2.3 Optimization Techniques . . . 15
  - 2.3.1 Compilation Options . . . 15
    - 2.3.1.1 Static and Dynamic Compilation . . . 16
    - 2.3.1.2 Compilation Flags . . . 17
  - 2.3.2 User-Space Optimization . . . 18
    - 2.3.2.1 Tools to reorder systemd services hierarchy . . . 18
    - 2.3.2.2 Simplifying SysVInit scripts . . . 20
    - 2.3.2.3 Pre-Init . . . 20
  - 2.3.3 Filesystem Optimization . . . 21
  - 2.3.4 Kernel Optimization . . . 21
  - 2.3.5 Linux Suspension . . . 22
  - 2.3.6 Boot Loader optimization . . . 23
    - 2.3.6.1 Configuration options optimization . . . 23
    - 2.3.6.2 Reducing Boot loader script . . . 24
    - 2.3.6.3 Falcon Mode . . . 24
  - 2.3.7 Memory Selection . . . 25
    - 2.3.7.1 SD card . . . 25
    - 2.3.7.2 Flash memory . . . 26

2.4 Related Work . . . 27
2.4.1 Boot time approach methodology . . . 27
2.4.2 GUI application solution . . . 29
2.4.3 Extreme minimalistic solution . . . 30
2.4.4 Similar platform solution . . . 31

3 Case-study Specifications **33**
3.1 System Description . . . 33
3.1.1 P689 overall architecture . . . 33
3.1.2 HUD initialization sequence . . . 34
3.2 Development Environment . . . 35
3.2.1 Target Description . . . 36
3.2.2 Host Description . . . 36
3.2.2.1 Yocto Project . . . 37
3.2.2.2 Layers of Yocto Project . . . 38
3.2.2.3 Yocto Project's Licensing . . . 39

4 Implementation **41**
4.1 Measurement preparation . . . 41
4.1.1 Boot time measurement . . . 41
4.1.2 Changes on HUD application . . . 43
4.2 Optimization Techniques . . . 43
4.2.1 HUD Compilation . . . 43
4.2.2 File system optimization . . . 45
4.2.3 Pre-Init . . . 46
4.2.4 Kernel Optimization . . . 47
4.2.5 Falcon mode . . . 50
4.2.6 Memory Selection . . . 52
4.2.7 Final boot sequence . . . 52
4.2.8 Additional Work . . . 53
4.2.8.1 systemd's Units Analysis . . . 54
4.2.8.2 U-Boot's boot script reduction . . . 56
4.2.8.3 Automating Yocto's build process . . . 56

5 Results **59**
5.1 Compilation Comparison . . . 59
5.2 File system reduction . . . 61
5.3 Service and Pre-Init . . . 62
5.4 Kernel configuration . . . 63
5.5 Memory comparison . . . 64
5.6 Falcon Mode . . . 65

6 Conclusion **67**
6.1 Results Discussion . . . 67
6.2 Future Work . . . 70

I U-Boot's Boot Command Flowchart **73**

II Peripheral layout of the i.MX6Q Sabresd Board **75**

III Overall architecture of P689 **77**

IV Qt libraries configuration **79**

V List of used dynamic libraries **81**

VI Modified flowchart of HUD application **83**

References **85**