Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 22 (2014) 132 – 137

www.elsevier.com/locate/procedia

3rd International Conference on Through-life Engineering Services

Self-healing fuel pump controller mapped into memory based finite state machine

Philipp Schiefer*, Richard McWilliam, Alan Purvis

Durham University, South Road, Durham, DH1 3LE, United Kingdom

* Corresponding Philipp Schiefer. Tel.: +44 191 334 2418; fax: +44 191 334 2408. *E-mail address:* philipp.schiefer@durham.ac.uk

Abstract

This paper describes our on-going research into the design of finite state machines (FSMs) that exhibit self-healing characteristics. The approach adopted here is based on conversion of the traditionally adopted logic hardware design into generic look-up table (LUT) format. Instead of relying upon bespoke hardware mitigation strategies such as triple modular redundancy, our approach relies upon well-established data error detection and correction (EDC) codes that are ideally suited to protecting LUTs. This 'memory-mapping' of logic brings self-healing capabilities that can be applied to a wide variety of FSMs. We illustrate our method by mapping a generic automotive used fuel pump controller (FPC) design to LUT format. Built-in repair is and fault monitoring are both considered to be extremely important embedded control applications and we therefore discuss significant benefits that can be brought by incorporating self-healing capability to the underlying hardware. We demonstrate the design principles of our approach verify the state-based behavior of the resulting FSM. We further discuss the how content addressable memory (CAM) can be used to achieve efficient address mapping. In order to protect against address errors occurring at the input, a two-stage LUT implementation is used that removes errors occurring in the input data stream as well as protection of the state mapping itself.

© 2014 Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/3.0/).
Peer-review under responsibility of the Programme Chair of the 3rd InternationalThrough-life Engineering Conference

Keywords: Finite State machine; Memory Look-Up Tables; Fuel Pump Controller; Content Access Memory

Technical glossary

Abbr	Definition
CAM	Content-addressable memory
ECU	Engine control unit
EDC	Error detection and correction
FPC	Fuel pump controller
FPGA	Field-programmable gate array
FPS	Fuel pressure system
FSM	Finite-state machine
LUT	Look-up table
MUX	Multiplexer
PLD	Programmable logic device
SEU	Single upset event
STB	State transition block

1. Introduction

In our world today every fossil fuel burning engine in different applications requires a type of fuel storage and a means of getting this fuel to the engine at the right pressure and amount required. This is done in all the cases with the help of an electric pump which can be controlled through a simple relay or a fuel pump controller (FPC) based on a PLD. For controlling the pressure for the simple relay solution a mechanical pressure stabilizing system is part of this fuel supply system. Mechanical system can cause problems in unstable pressure , which is why most of fossil fuel driven engines are equipped with pressure controlled FPC. FPCs are based on PLDs programed in standard programming logic

2212-8271 © 2014 Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license
(http://creativecommons.org/licenses/by-nc-nd/3.0/).
Peer-review under responsibility of the Programme Chair of the 3rd InternationalThrough-life Engineering Conference
doi:10.1016/j.procir.2014.07.151

according to a given specification that describes the system behavior in regards of performance and total car system interfacing. The core of a PLD system contain in most cases a type of microcontroller based system and in some cases a field programming gate array (FPGA) surrounded by sensor and output interfacing hardware. Today`s FPC are a part of complex engine systems controlled by several electrical control units (ECU) that require inter-system communication for maintaining optimal system behavior for performance in the event of sensor degradation and to handle fault conditions. This research work focuses on the main task of the FPC providing fuel at the right amount and pressure to the engine. The adaptation is achieved by converting and coding the state transition diagram into a memory based system which only uses the memory data to perform the FSM task. For accurate pressure control feedback or feed-forward control can be utilized in the application. What particular controller hardware gets chosen depends on the design and the preference of the designer. The research work described in this paper is based on a P (proportional) controller type which can be implemented by switching on the power to fuel pump at a given voltage (V) level to produce the required minimal fuel pressure to start the engine. The required fuel pressure is achieved by incrementing or decrementing the voltage level in a stepwise fashion. A LUT based FPC simulation is presented in Figure 1 which is based on a MatLab adaptation of our approach. Other than the P type controller, a PI (proportional integer) or PID (proportional integer derivative) type controller behavior can been used.

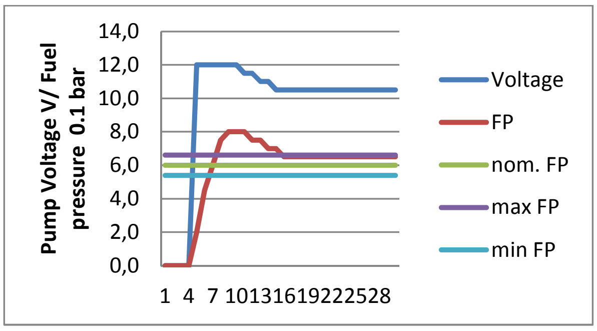

Fig. 1 Simulation data of memory mapped FPC

The FPC is then mapped into a self-protecting FSM in the form of a Moore state machine, wherein the memory is CAM based in order to produce a space-efficient mapping. CAM based memory offers the advantage over linear addressable memory, that is uses pattern matching for data access. Through this feature the memory structure is compactor and memory gaps can be avoided. The task of the self-protecting of the FSM is been done through dynamic input filtering with the goal of limiting the possibility of FSM upsets. The task of self-controlling and healing are part of the goal of maintain a functional LUT based FSM in case of memory data corruption. This three features are seen as self-* qualities for creating a fault tolerant system. For the FSM coding the approach of self-configuring FSM for LUT out of [1] gets used. The complete fuel pressure system (FPS), comprising FPC and fuel pump will be limited in this simulation to the FPC only and simulated. A basic FPS as a block diagram gets presented in Figure 2.

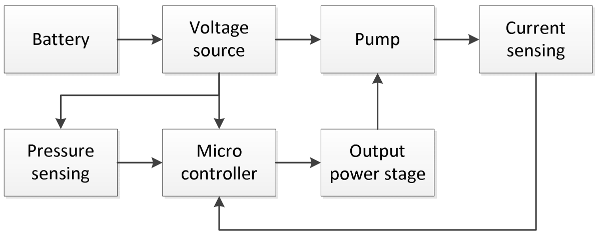

Fig. 2: Basic block diagram of a FPS

2. LUT-based architectures for implementing FSMs

FSMs are used in many applications where a fixed mapping between input and output logic patterns is needed. In its simplest form, the FSM stores a direct one-to-one mapping between input logic data patterns and the corresponding output data patterns. This is normally achieved by storing all of the possible mappings in the form a LUT and using the current input pattern to look-up the relevant output pattern. For simple input/output mappings, the contents of the LUT can be determined directly by analyzing the system's state transition diagram.

More complex input/output mappings are possible, in which case a formal approach can be used. The logic design of an FSM with a given number of inputs I, outputs O and states S can be defined as a 5-tuple (I,O,S,δ,ω). The state transition function is δ : I x S → S and the output function is ω : I x S → O [2-4]. An FSM alters its current state following the defined state transition defined by the input specified at this state and is defined as either a Moore or a Mealy state machine [5-7] as illustrated in Figure 3.

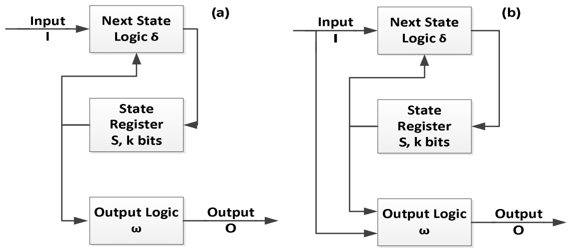

Fig. 3 Moore (a) and Mealy (b) FSMs [7]

The difference between Moore and Mealy is defined by the output response in regards to the input stimulus. In a Moore based FSM the output functionality is controlled by the state only (see Figure 3a) and defined as ω : S → O. The output function of a Mealy state machine depends on the state and input and is defined as ω : I x S → O (see Figure 3b) [2-4]. Between the two designs there is an important dissimilarity: - Mealy require less state transition than a Moore. In this regards a Mealy FSM has more compact state transition

Download English Version:

https://daneshyari.com/en/article/1699857

Download Persian Version:

https://daneshyari.com/article/1699857

Daneshyari.com