SE3910 – REAL TIME SYSTEMS

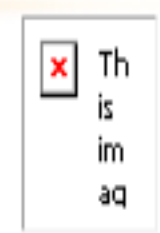
Ethics and Professional Responsibility for Embedded Systems

ROADMAP

- Today
  - Toyota systems failure
- Friday
  - Exam review and course wrapup

OBJECTIVES

- Explain the patriot missile failure
- Explain what is meant by expert testimony
- Understand the ethical responsibility of an engineer when giving expert testimony



MATERIAL BACKGROUND

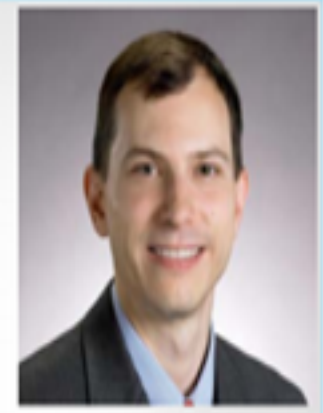
- 2007 A single vehicle crash occurs which injures the driver and kills the passenger in Oklahoma
- 2011 – NASA issues a report on unintended acceleration in Toyota vehicles
- January 2012 – Multiple engineers from the Barr group are able to analyze the Toyota software
- July 2012 billion dollar economic loss settlement
- October 2013 testimony from Michael Barr
- October 2013
  - Oklahoma jury found that Toyota owed each victim $1.5 million in compensatory damages and also found that Toyota acted with "reckless disregard"
- On December 13, 2013, Toyota settled another West Virginia case
- In March 2014,
  - the U.S. Department of Justice announced a $1.2 billion settlement in a criminal case against Toyota. As part of that settlement, Toyota admitted to past lying to NHTSA, Congress, and the public about unintended acceleration and also to putting its brand before public safety.
- April 1, 2014,
  - Michael Barr gave a keynote speech at the EE Live conference, which touched on the Toyota litigation
  - http://www.barrgroup.com/killer-apps/.
  - Material for this lecture comes from here.

WHO IS MICHAEL BARR

- Lets read from the court testimony (Page 23, lines 1 – 24)

MICHAEL BARR
Co-Founder/CTO, Barr Group

#KillerApps



Electrical Engineer (BSEE/MSEE)

Experienced Embedded Software Developer

Consultant & Trainer (1999-present)
- Embedded software process and architecture improvement
- Various industries (e.g., medical devices, industrial controls)

Former *Adjunct Professor*
- University of Maryland 2000-2003 (Design and Use of Operating Systems)
- Johns Hopkins University 2012 (Embedded Software Architecture)

Served as *Editor-in-Chief*, *Columnist*, *Conference Chair*

Expert witness (software patents/copyrights, product liability)

Author of 3 books and 70+ articles/papers

BARR group

THE PATRIOT MISSILE FAILURE

Brig General Neal, US Command

- "Looks like this [SCUD] broke apart in flight … [thus] wasn't in the parameters where it could be attacked."

Col. Garnett, Patriot Program Director

- "An anomaly that never showed up in thousands of hours of testing."

THE SOFTWARE BUG

THE PATRIOT SOFTWARE BUG

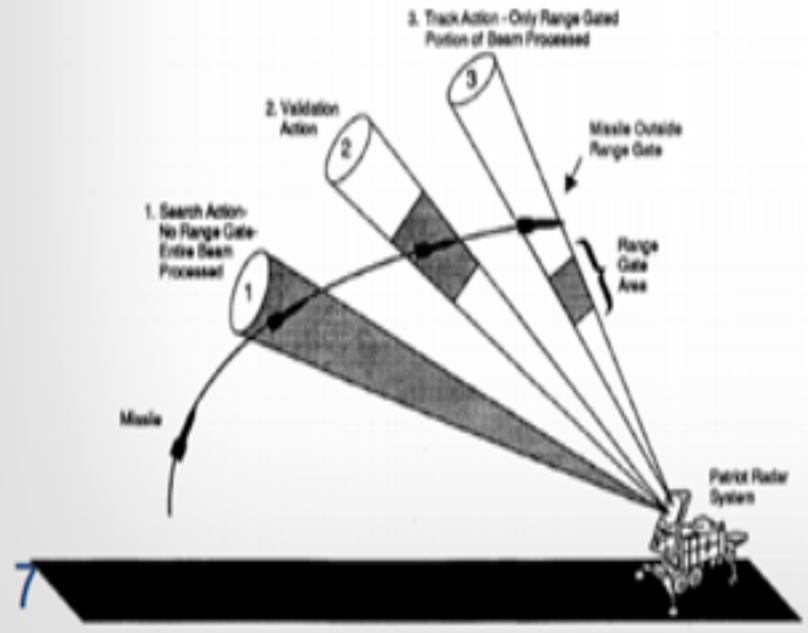
#KillerApps

Two versions of system time

- Timer chip integer representation
- Software fixed-point binary format

7.5s: $0000000000000000000111.1000000000000000000000_2$

Increasing inaccuracy...



uptime (h)	error (s)	shift (m)
1	.0034	7
8	.0275	55
20	.0687	137
100	.3433	687

BARR group

GAO Report: https://www.fas.org/spp/starwars/gao/im92026.htm

Th is im aq

MICHAEL BARR: WHAT IS EMBEDDED SOFTWARE PROCESS

- Page 26 line 16 of testimony to page 27 line 9

WHAT IS SOURCE CODE

- What is source code?
- Page 30 line 15 to page 31 line 7

UNINTENDED ACCELERATION

What is unintended acceleration?

- Acceleration the driver did not purposely cause

> [1] In this report, "unintended acceleration" refers to the occurrence of any degree of acceleration that the vehicle driver did not purposely cause to occur. Contrast this with the term "sudden acceleration incident," which refers to "unintended, unexpected, high-power accelerations from a stationary position or a very low initial speed accompanied by an apparent loss of braking effectiveness." *An Examination of Sudden Acceleration,* DOT-TSC-NHTSA-89-1 at v. As used here, unintended acceleration is a very broad term that encompasses sudden acceleration as well as incidents at higher speeds and incidents where brakes were partially or fully effective, including occurrences such as pedal entrapment by floor mats at full throttle and high speeds and incidents of lesser throttle openings at various speeds.

Loss of driver control of engine power

- A very dangerous vehicle malfunction!

BARR group

Source: http://www.nhtsa.gov/staticfiles/nvs/pdf/NHTSA-UA_report.pdf ("NHTSA"), p. vi

WHAT WAS MICHAEL BARR ASKED TO DO?

- Page 31 lines 8 – page 33 line 1

MY REVIEW OF TOYOTA'S SOURCE CODE

Access to Toyota's "electronic throttle" source code

- In a secure room in Maryland
- Subject to confidentiality agreements
- For vehicle models with ETCS spanning ~2002-2010 model years

  Camry, Lexus ES, Tacoma, and others

Approximately 18 months of calendar time with code

- By a very experienced team of embedded systems experts

  Including 3 other engineers from Barr Group
- Building upon NASA's earlier source code review; digging deeper

NASA must reach a clear-cut conclusion by the end of August.

So they are under a fair amount of pressure.

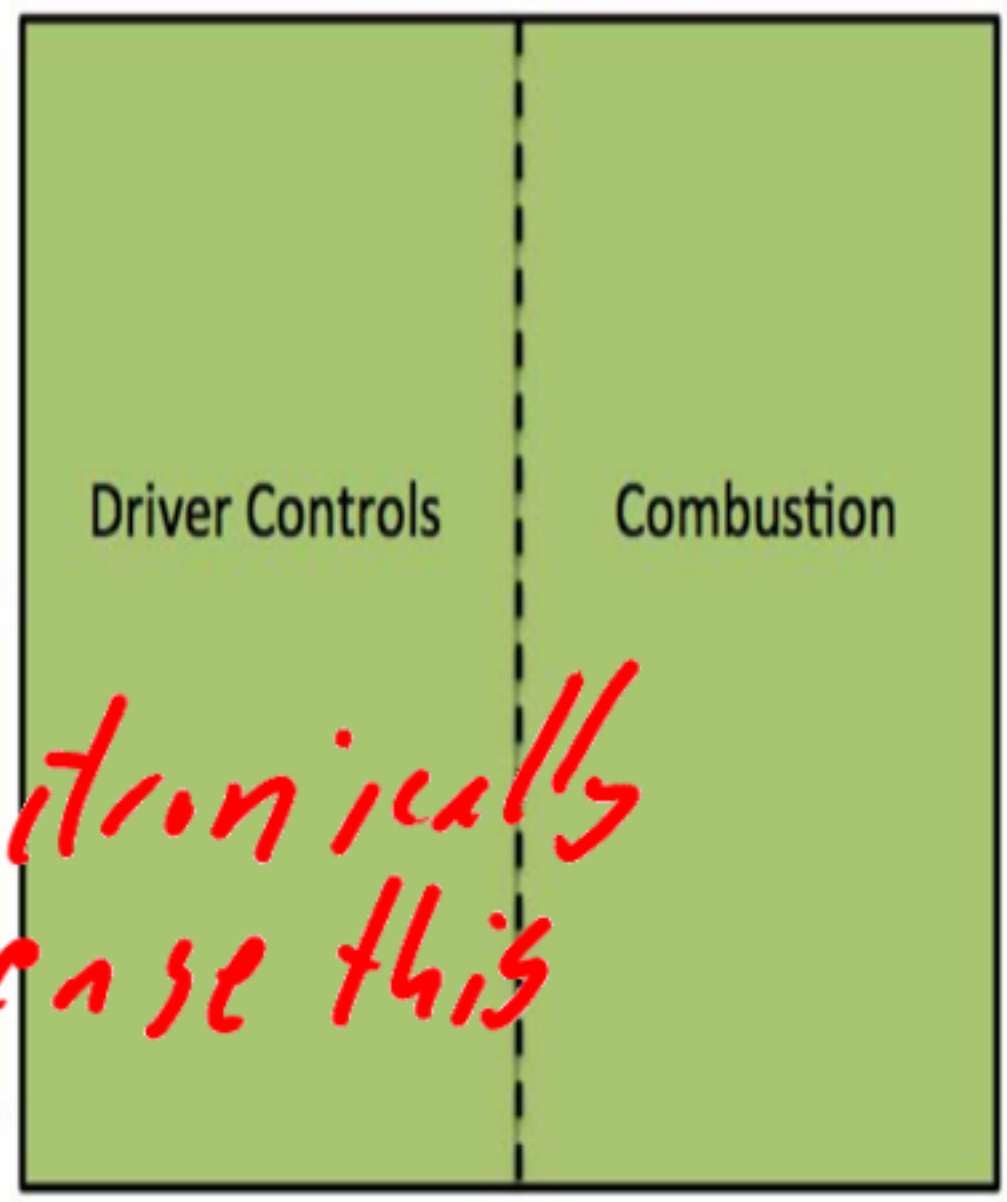
TOY-MDL0595137
TOY-MDL05951378P-00

BARR group

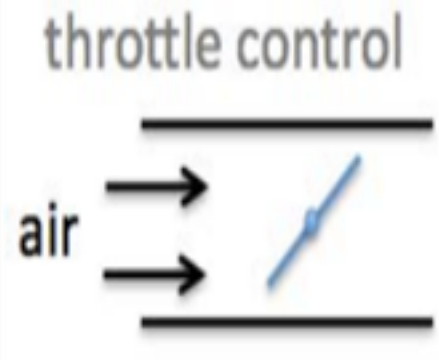
THE TOYOTA SYSTEM

#KillerApps

ELECTRONIC THROTTLE CONTROL



Electronically Sense this



WHAT TYPES OF COMPLAINTS DID TOYOTA RECEIVE?

#KillerApps

TOYOTA'S HIGH COMPLAINT RATE

Complaints jump after "electronic throttle"

- NHTSA data 2004 vs. 2000-2003
  - *All UA complaints ~2,000 (vs. 1,200-1,400)*
  - *Toyota's percentage ~20% (vs. 4-7%)*

Toyota +300%

Complaint Statistics: http://democrats.energycommerce.house.gov/Press_111/20100222/Detailed.Timeline.and.Background.of.NHTSA.Actions.Regarding.Toyota.Sudden.Acceleration.pdf

Could driver errors explain the jump?

- Expect driver errors ~even across makes
- Why such a big increase w/in Toyota?

BARR group

IS IT AN ELECTRONICS PROBLEM?

Statistical analysis and slide by R.A. Whitfield to NAS' Transportation Research Board (Oct 2010)

How likely is it that these factors...

Vehicle Factors:
- Floor mats
- Sticky pedals

Recalls of some cars.

- Pedal placement
- Gated gear shift pattern
- Ignition switch design

Driver Factors:
- Mass hysteria
- Fraud
- Old age
- Youth
- Inexperience
- Incompetent drivers

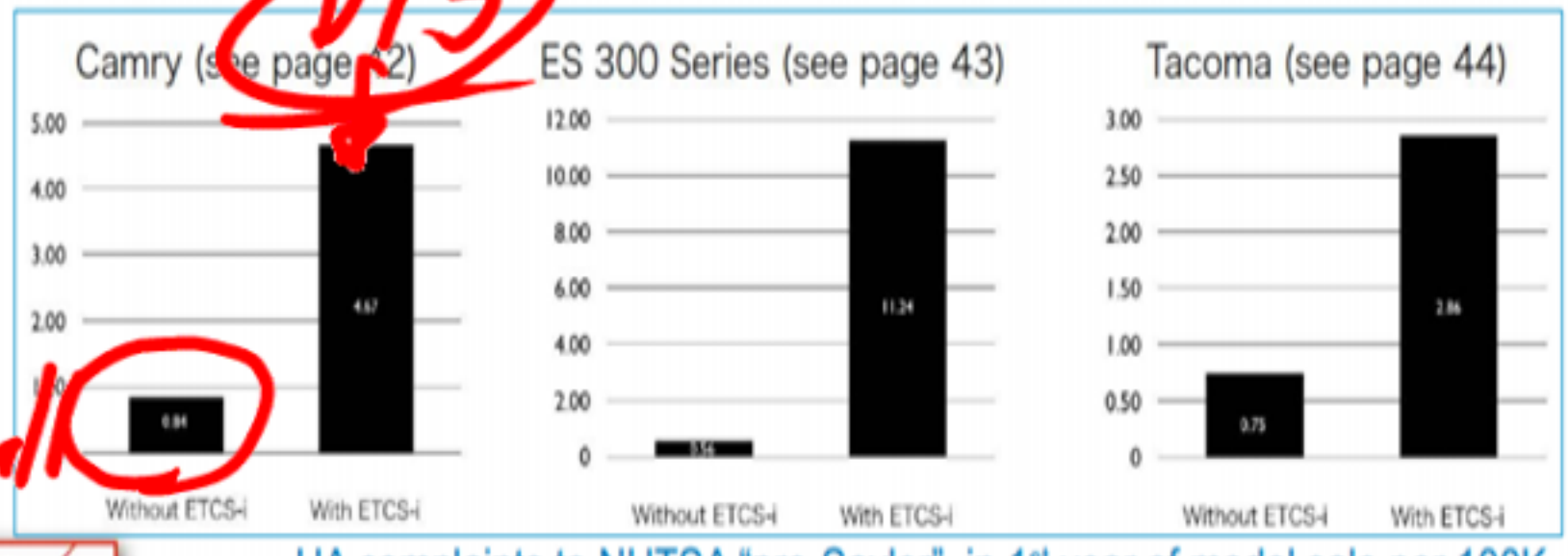
Environmental/Usage Factors

Factors held ~constant.

...explain these results when controlling for make/model and years in service?

Big

Small



UA complaints to NHTSA "pre-Saylor", in 1st year of model sale per 100K.

QCS CORP

Source: http://onlinepubs.trb.org/onlinepubs/UA/101011Whitfield.pdf

This im aq

NASA IS BROUGHT IN
PAGE 39 LINES 2 - 26

#KillerApps

KEY NASA STATEMENTS

Electronic Throttle

Because proof that the ETCS-i caused the reported UAs was not found does not mean it could not occur. However, the testing and analysis described in this report did not find that TMC

Due to system complexity which will be described and the many possible electronic hardware and software systems interactions, it is not realistic to attempt to "prove" that the ETCS-i cannot cause UAs. Today's vehicles are sufficiently complex that no reasonable amount of analysis or testing can prove electronics and software have no errors. Therefore, absence of proof that the ETCS-i has caused a UA does not vindicate the system. From calendar year 2005 to 2010 TMC

The NESC team identified two hypothetical ETCS-i failure mode scenarios (as opposed to non-electronic pedal problems caused by sticking accelerator pedal, floor mat entrapment, or operator misapplication) that could lead to a UA without generating a diagnostic trouble code (DTC): specific dual failures in the pedal position sensing system and a systematic software malfunction

The second postulated scenario is a systematic software malfunction in the Main CPU that opens the throttle without operator action and continues to properly control fuel injection and ignition.

BARR group

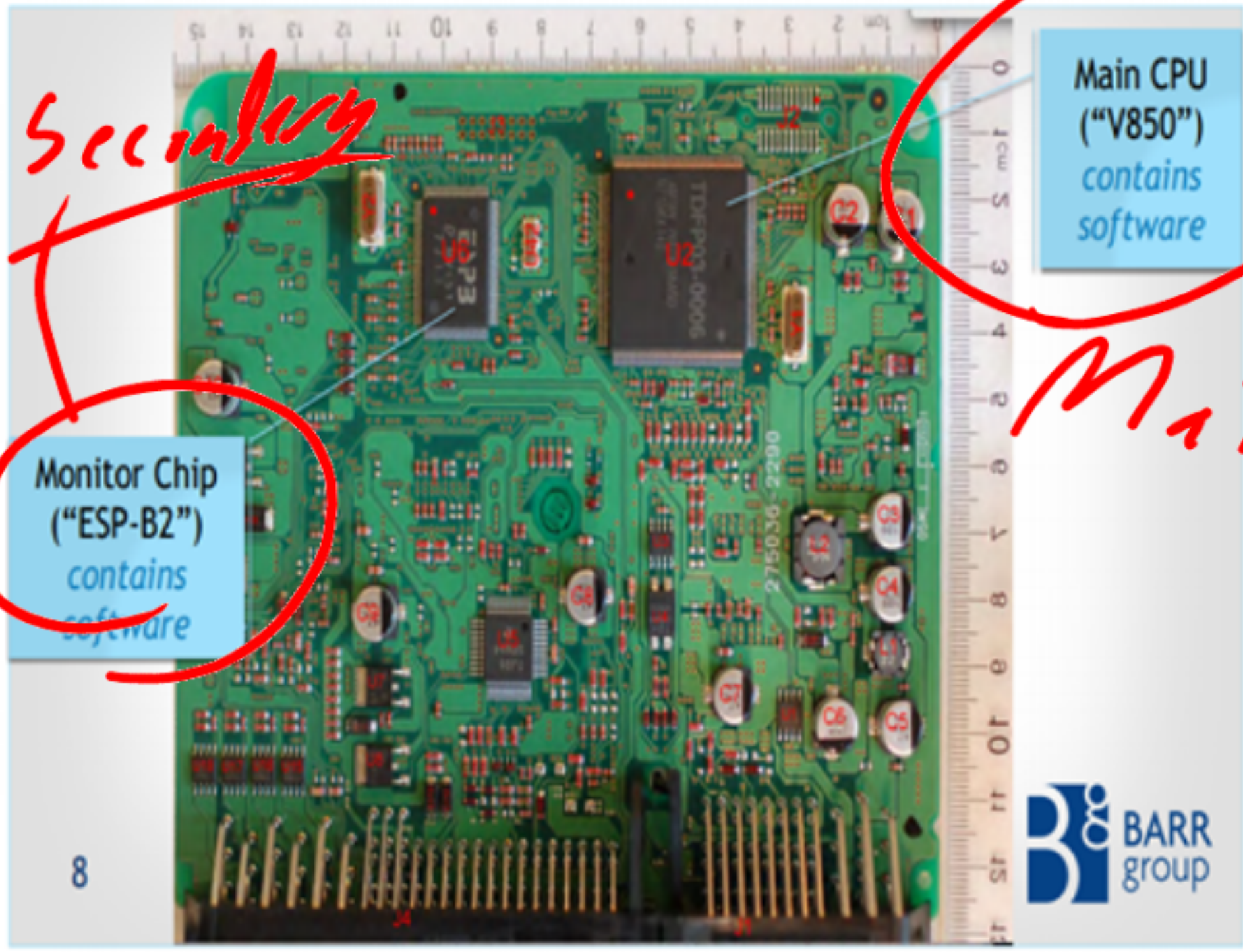
Source: http://www.nhtsa.gov/staticfiles/nvs/pdf/NASA-UA_report.pdf ("NASA"), pp. 15-20

WHAT WAS TOYOTA'S SOURCE CODE LIKE?

- Page 42, lines 8-22

TOYOTA'S ECM

TOYOTA'S ENGINE CONTROL MODULE (ECM)



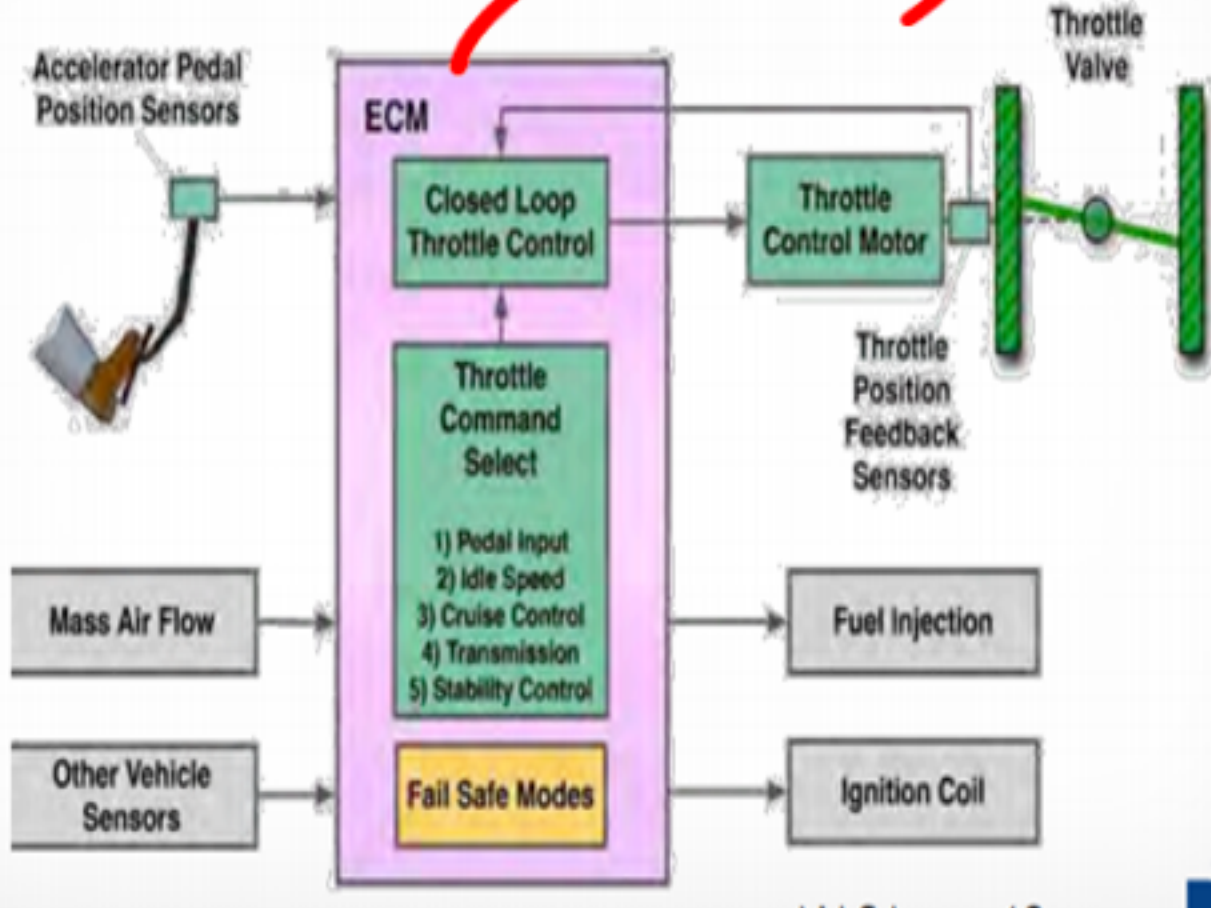
BARR group

TOYOTA'S ECM MODULE

ELECTRONIC THROTTLE CONTROL (ETCS)

"Toyota ETCS-i is an example of a safety-critical hard real-time system."

- *NASA, Appendix A, p. 118*



NASA, p. 13

BARR group

Engine control module

Th
is
im
ag

THE CRASH RECONSTRUCTED

#KillerApps

BOOKOUT RECONSTRUCTION

Speed estimates

- Skid start ~50mph
- At impact ~25mph

Agreed she braked

- Parking brake too?

150’ skid mark

- Way too long!

Th
is
im
ag

- Page 47, line 17 – page 48 line 9

HOW DO WE DETECT SOFTWARE BUGS

THE TOYOTA SOFTWARE

SUMMARY OF 2005 CAMRY L4 CONCLUSIONS

Toyota's ETCS source code is of unreasonable quality

- Toyota's source code is defective and contains bugs
  - Including bugs that can cause unintended acceleration
- Code quality metrics predict presence of additional bugs

Toyota's fail safes are defective and inadequate

- "House of cards" safety architecture
  - Random hardware and software faults are a fact of life

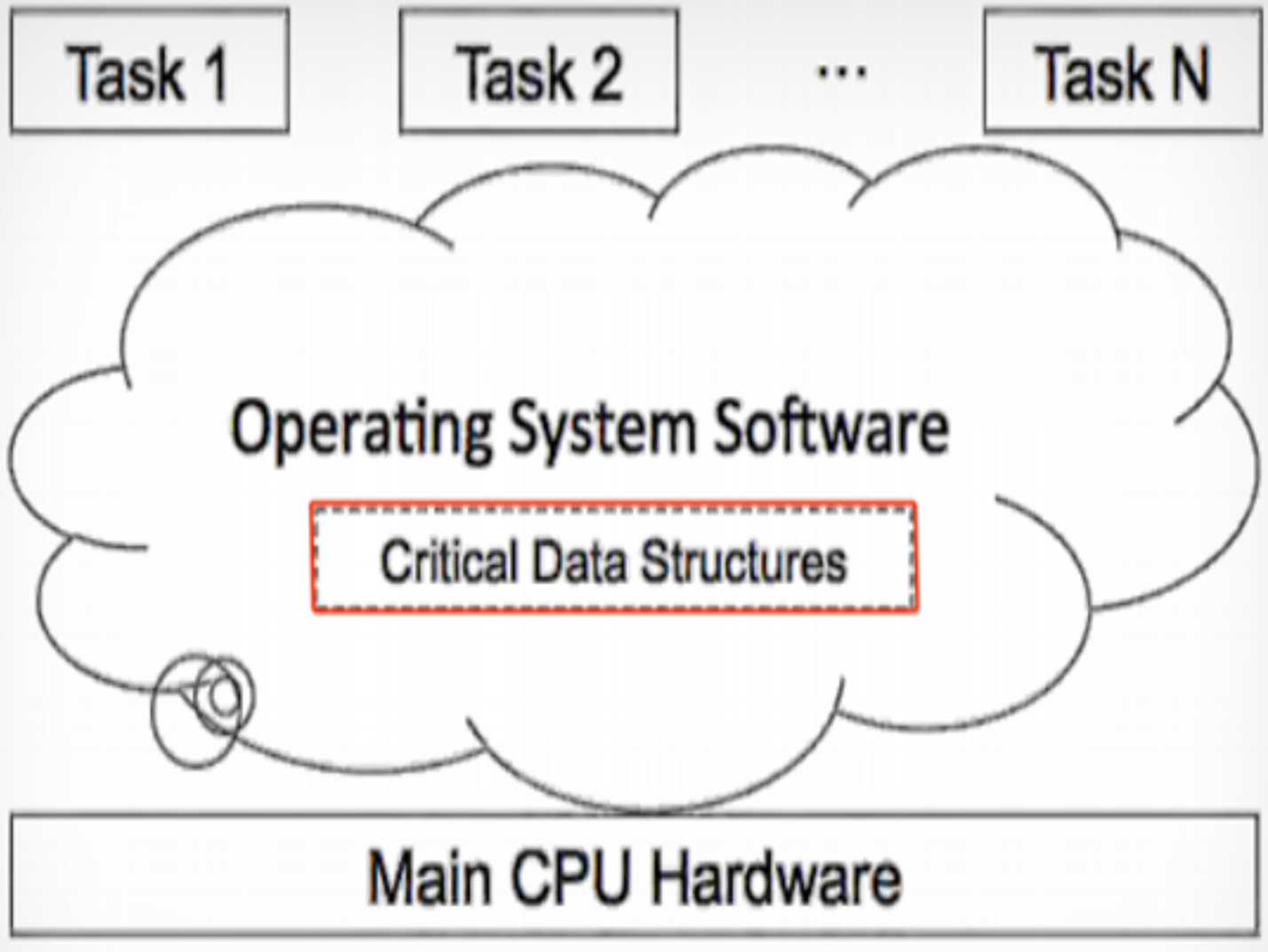
Misbehaviors of Toyota's ETCS are a cause of UA

BARR group

Th
is
im
ag

THE TOYOTA OPERATING SYSTEM (OSEK)

TOYOTA'S OPERATING SYSTEM (OSEK)

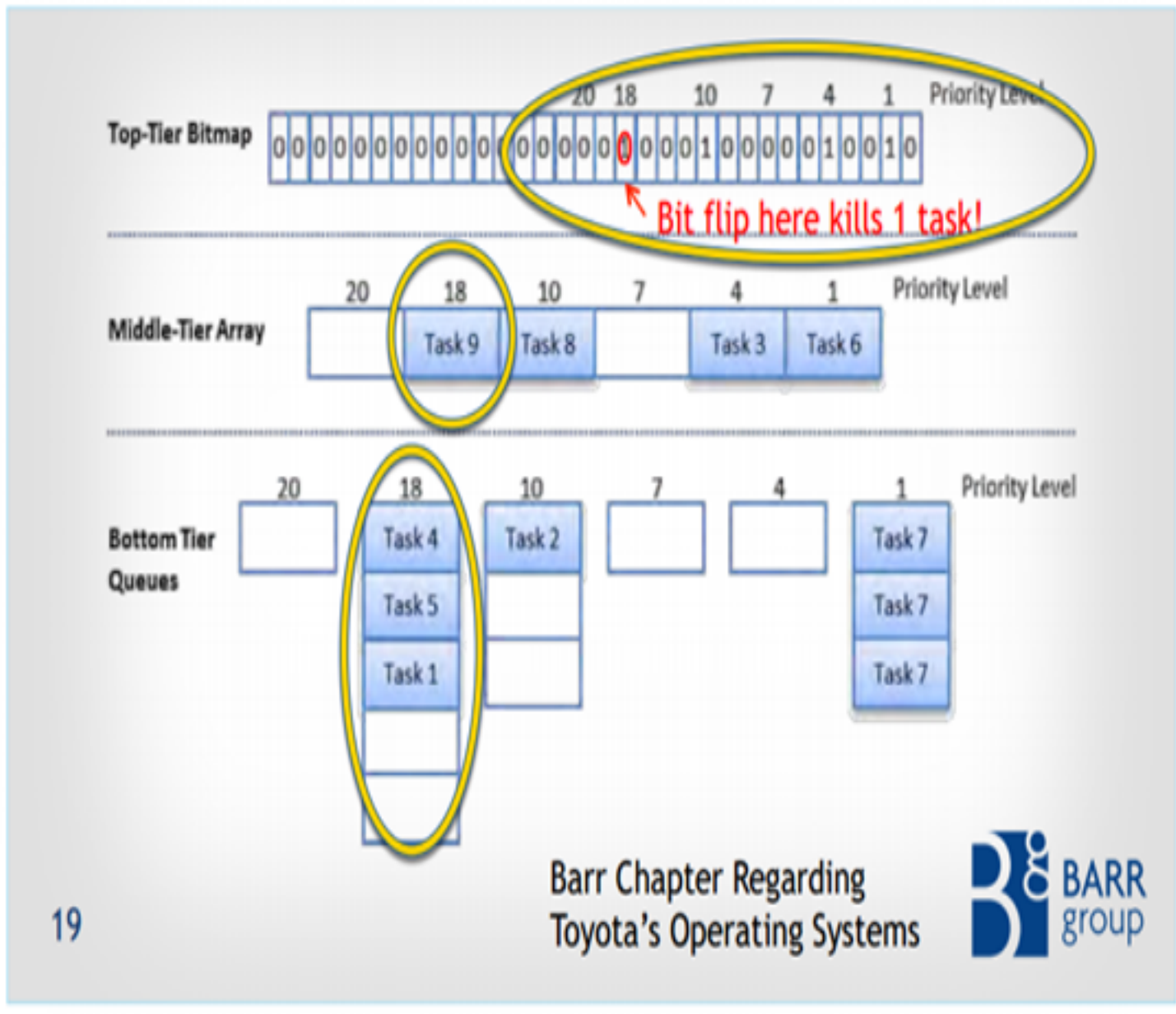


Barr Chapter Regarding
Toyota's Operating Systems

BARR group

MEMORY CORRUPTION CAN CAUSE FAILURE

MEMORY CORRUPTION AND TASK DEATH



- Page 89, lines 10-25

WHAT CAUSES MEMORY CORRUPTION

SOFTWARE CAUSES OF MEMORY CORRUPTION

Type of Software Defect	Causes Memory Corruption?	Defect in 2005 Camry L4?
Buffer Overflow	Yes	Yes
Invalid Pointer Dereference/Arithmetic	Yes	Yes
Race Condition (a.k.a., "Task Interference")	Yes	Yes
Nested Scheduler Unlock	Yes	Yes
Unsafe Casting	Yes	Yes
Stack Overflow	Yes	Yes

21

Barr Chapter Regarding Toyota's Software Bugs

BARR group

Th is im aq

CODE METRICS FOR TOYOTA CODE

TYPES OF SPAGHETTI CODE

Data-flow spaghetti

- Complex coupling between software modules and between tasks
- Count of global variables is a software metric for "tangledness"

  2005 Camry L4 has >11,000 global variables (NASA)

Control-flow spaghetti

- Many long, overly-complex function bodies
- Cyclomatic Complexity is a software metric for "testability"

  2005 Camry L4 has 67 functions scoring >50 ("untestable")

  The throttle angle function scored over 100 (unmaintainable)

Barr Chapter Regarding Toyota's Code Complexity

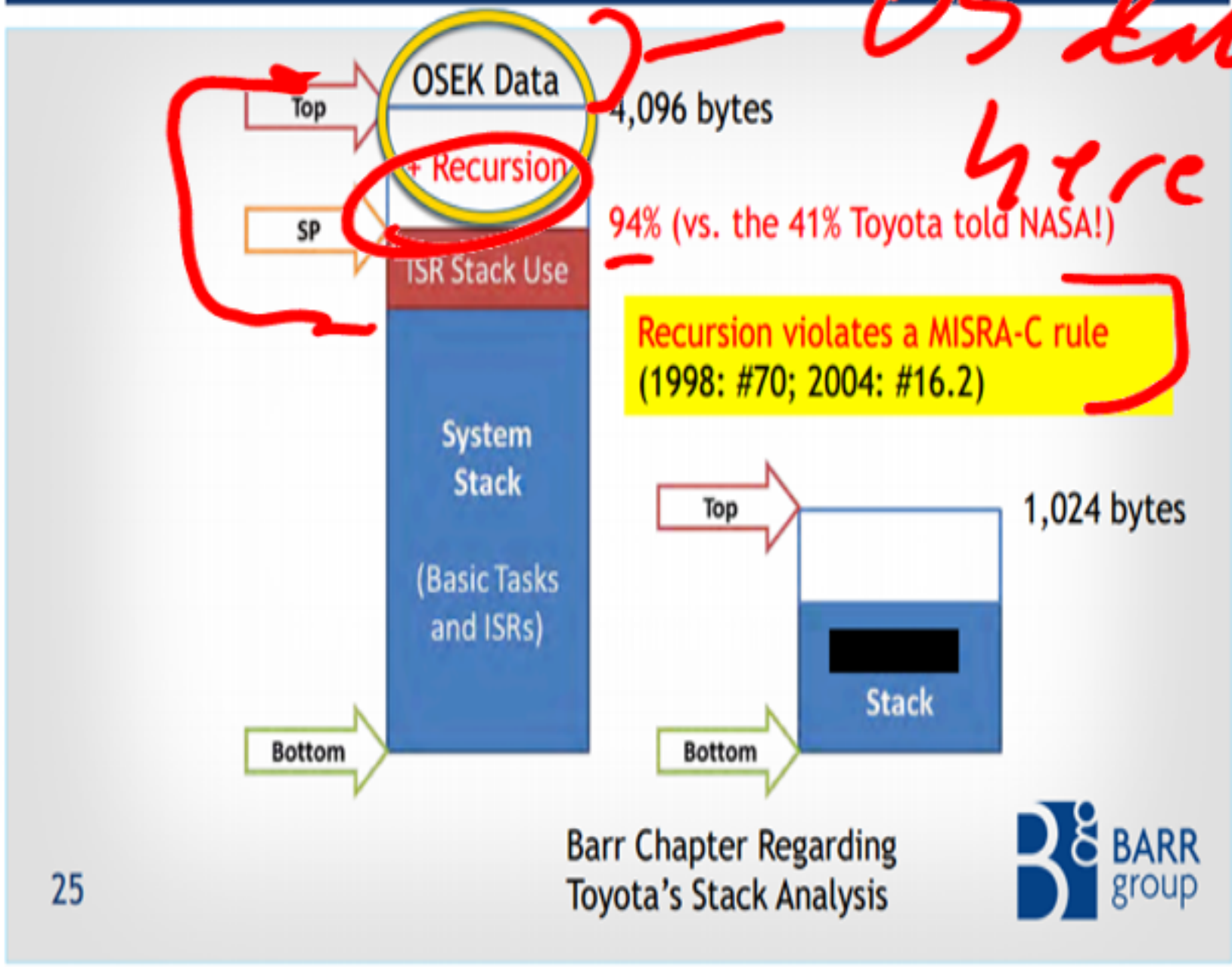
BARR group

Wow!

Cyclomatic complexity

TOYOTA CODE

STACK ANALYSIS FOR 2005 CAMRY L4



Page 36, line 23-37:15

OS data is here.

MISRA C

Q. And in the review of what Toyota had done did NASA fine any violation of these codes

A. Yeah, NASA found a number of violations of MISRA rules.

Q. Did you find violations?

A. Yes. NASA looked at about 35 of the rules. There's in total, I forget the exact number. It's basically the same set of rules in 1998 and 2004. But as I recall, it's over 100 rules total. NASA looked at 35 of them and they found over 7,000 violations, and they reported that on page 29.

I checked the full set. There were a couple that were difficult to test, but basically the full set and found more than 80,000 violations in the 2005 Camry.

Q. There was also a discussion about compliance with MISRA rules that we heard from Mr. Ishii, I think he said something like maybe 50 percent of compliance of used MISRA rules. In your code review did you find that to be true?

A. No.

Q. Was did you find?

A I actually wrote on whole report on Toyota's coding standard in one of my chapters, and what I found studying their coding standard was that actually -- the MISRA rules are over 100 rules and the Toyota rules -- I have an appendix that lists them all -- I think it's about the same number, about 100, maybe 119, but only 11 of Toyota's coding standard rules overlap with the MISRA C rules. And interestingly, five of those rules are violated in Toyota's code.

So when they say 50 percent overlap between the two, our rules and MISRA rules, no.

WATCHDOG DESIGN

TOYOTA'S DEFECTIVE WATCHDOG DESIGN

Toyota's watchdog supervisor design is unreasonable

- Incapable, ever, of detecting death of majority of tasks
- Incapable of properly and reliably detecting *CPU overload*
- Allows vehicle misbehavior due to overloads lasting up to 1.5s
- Resets the watchdog timer hardware in a timer tick ISR
- Explicitly ignores and discards most operating system error codes

Ignoring error codes violates a MISRA-C rule (1998: #86; 2004: #16.10)

Reasonable design alternatives were well known

- Indeed the primary purpose should've been to detect task death
- 2005 Prius (HV-ECU) watchdog is better

Barr Chapter Regarding
Toyota's Watchdog Supervisor

BARR group

THE DEFECTIVE SOFTWARE PROCESS

TOYOTA'S DEFECTIVE SOFTWARE PROCESS

FMEA was incomplete; single points of failure are present

- Because: Toyota didn't adopt a formal safety process

Peer reviews not done on OS code and ESP-B2 code

- Because: Toyota didn't perform code reviews; used non-standard OSEK

Toyota's own "power train" coding standard not enforced

- Because: Toyota didn't follow through with software suppliers

Watchdog supervisor doesn't detect most task's deaths

- Generally costs less to push the limits than upgrade to faster CPU

No EDAC protection against hardware bit flips

- Generally costs less to make memory chips without EDAC

If confident, why let NASA believe there was EDAC?

46

HW

BARR group

WHAT HAVE WE LEARNED FROM THIS?