U.S.NRC
UNITED STATES NUCLEAR REGULATORY COMMISSION
Protecting People and the Environment

WGRISK Digital I&C Failure Mode Taxonomy

Summary of BNL Failure Modes and Effects Analysis (FMEA) Approach

Gabriel Taylor
Division of Engineering
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
March 28-29, 2011

Background

- The goal of developing a failure mode taxonomy is to support both qualitative and quantitative assessment of digital system reliability models that will eventually be integrated into nuclear power plant (NPP) probabilistic risk assessments (PRAs).
- Materials relevant to developing failure mode taxonomy submitted by Brookhaven National Laboratories (BNL)
  - The generic FMEA approach applied to reliability assessment of a digital feedwater control system using “traditional” PRA methods.
  - Software FMEA documented in “Appendix C.”
  - For a reactor protection system, failure modes of software were provided by the experts of a workshop on software reliability held at BNL.

U.S.NRC
United States Nuclear Regulatory Commission
Protecting People and the Environment

Issues with Digital System FMEA

- There is no well-established definition of the failure modes and their effects and there is no specific guidance of how to undertake FMEAs for digital systems. More specifically,
  - system failure modes are generally defined according to system functions. However, digital system functionalities are in general highly complicated.
  - determining failure effects requires propagating failure modes. The propagation may not be straightforward due to the complexity of digital systems, especially for failure modes defined at a more detailed level.
  - selecting an appropriate level of detail at which to define failure modes is necessary but difficult because of the need to consider requirements such as
    - Failure mode meaningfulness, genericness, failure effects propagation, availability of data
    - Modeling of hardware and software interaction
    - Capturing fault-tolerance features.

Some Observations of Digital Systems

U.S.NRC
United States Nuclear Regulatory Commission
Protecting People and the Environment

- A digital system is envisioned as consisting of modules, each comprised of generic components, such as analog/digital (A/D) converters, multiplexers (MUXs), microprocessors etc., which
  - are commonly used and perform the same functions in general digital systems.
  - if failed, directly affect the associated output signals; this enables a definition of generic failure modes and propagation of failures based on output signals.
  - are supported by some publicly available failure data; it is expected that much more data is possessed by component vendors.
  - directly interact with software and fault-tolerance features; this interaction needs to be accounted for in reliability assessment.

BNL's Generic FMEA Approach

U.S.NRC
United States Nuclear Regulatory Commission
Protecting People and the Environment

- The FMEA approach was developed based on the previous observations
  - The entire digital system is decomposed into different levels of detail until the level of the generic components is reached.
  - The number of intermediate levels depends on the complexity of, and information available about, the particular system.
  - Failure modes are postulated at the lowest levels (i.e., at the level of generic components), and their effects propagated to higher levels until the impacts on the entire system can be determined.
- The key points in performing FMEAs are that
  - the status of the system eventually is determined by module signals that reflect the interactions between the modules and between the digital system and the plant and
  - these signals are directly affected by the status of the generic components of the modules.

Example Failure Modes of Generic Components

- Microprocessor of the Main central processing unit (CPU): Failure modes are (1) the microprocessor seems to be running normally but sends erroneous output and (2) the microprocessor stops updating output.
- Associated Components of a Microprocessor, such as the industry standard architecture (ISA) bus, random access memory (RAM), read only memory (ROM), basic input/output system (BIOS), flash disk, buffer, and serial port: It is conservatively assumed that each component has only one failure mode, i.e., a loss of the component, which entails the loss of the functions performed by the component.
- MUX and de-multiplexer (DEMUX): Failure modes of MUXs and DEMUXs are defined in terms of the analog signals they process, which include a loss of one or all signals.
- Software: Three failure modes for individual software failures and software common cause failures (CCFs) are considered: (1) the software on the main CPU seems to be running normally but sends erroneous output; (2) the software halts and hence, the CPU stops updating output; and (3) software runs with misleading commands to the user, incomplete or incorrect display of information. Note, only the first two failure modes were used in the digital feed water control system (DFWCS) reliability assessment.

Levels of Software Failure Modes

- Software failure modes can be defined at the following three levels: a system level, a microprocessor level, and a sub-level.
  - System level failure modes indicate the failure modes of the collection of all software of a digital system.
  - Microprocessor level failure modes indicate the failure modes of a software running on a microprocessor of a digital system.
  - Sub-level indicates a more detailed level than the microprocessor and definition of a sub-level may vary in different studies.

BNL Software Failure Modes: Microprocessor Level

- In the proposed FMEA approach, the software of a specific module was treated as a generic component like the microprocessor of a module because
  - software runs on a microprocessor and interacts with other hardware components via the microprocessor.
  - this enables modeling of interactions between digital system hardware and software and capturing some fault-tolerance feature, e.g., the external watchdog timer of the main CPU in the DFWCS can detect the failure of software to update output and cause a fail-over to the backup CPU.

BNL Software Failure Modes: Sub-level

- A software running on a microprocessor is considered to consist of the following elements
  - (1) input, (2) output, (3) communication, (4) resource allocation, and (5) processing elements.
- This enables a definition of software failure modes at sub-level (i.e., software element levels) namely
  - (1) Timing/order failure; (2) Interrupt induced failure; (3) Omission of a function or an attribute; (4) Unintended function or attribute; (5) Incorrect implementation of a function or an attribute; (6) Data error.

Reactor Protection System (RPS) Software Failure Modes from a Workshop on Software Reliability at BNL

- Generic software modes at system level and microprocessor-level were provided by experts at the workshop for the application to protection system software.

Level	Failure Modes
System Level	(1). Failure to generate signal in time (omission failure); (2) Spurious signal; and (3) Adverse effects on other functions (systems, operators).
Microprocessor Level	(1) Hang; (2) Abort; (3) Missing operation; (4) Extra operation; and (5) Erroneous operation.

Conclusions

- Pros of BNL FMEA approach
  - Failure modes for generic components are meaningful, generic, supported by some publically available data and can be propagated feasibly;
  - Enables integrated modeling of hardware and software interaction
  - Can capture fault-tolerance features
- Cons of BNL FMEA approach
  - Failure mode definition is not straightforward for all components and additional research is needed.
  - Component failure propagation is highly complicated and a manual implementation of the approach is extremely difficult if not impossible.
- The BNL FMEA approach has been applied to a DFWCS supported by a source-code based automated tool to propagate the failures and determine the system status.
- The proposed FMEA approach is also applicable to a protection system.