An assessment of hazard probability due to Pentium processor errata in automatic train control applications

C. Bantin
Alcatel, Canada

Abstract

The Alcatel automatic train control products make use of a single-board computer that has been designed specifically for the railway environment. The computer is based on the Pentium processor and is extensively used for automatic train operation and automatic train protection functions. It is also used as a security authentication gateway interfacing to the radio-based data communications system.

The development of an automatic train control system must be accompanied by a detailed and extensive safety case in order to demonstrate that the required safety integrity level can be obtained. For the Pentium-based processor, the safety case must include the occurrences of errata, or faults in the design and implementation of the processor that are not discovered at the time of manufacture. It may be argued that, since errata are design and manufacturing errors, they are systematic. However, because of the way these faults manifest themselves it could be argued they are random. In fact, for any one processor, there is a random errata discovery process based on the fact that all the processors in use are operating simultaneously with different applications and/or different data. There is a particular probability that one of them will discover a fault, or errata. A statistical model is developed based on an in-depth analysis made of Pentium errata, and assumptions about the number of processors in use over the time period of the analysis. A probability is calculated that previously undiscovered errata will be found in one of the processors in an ATC system, and it is demonstrated to become lower than the required hazard probability well before any such ATC system containing these processors is out into revenue service.

WIT Transactions on The Built Environment, Vol 88, © 2006 WIT Press
www.witpress.com, ISSN 1743-3509 (on-line)
doi:10.2495/CR060121

1 Introduction

The EN 50129 standard for safety related electronic systems classifies failure integrity (faults) as either systematic or random. The systematic part is considered non-quantifiable and there are procedures for developing a safety case under these conditions. When it comes to microprocessor faults, it may be argued that they are design and manufacturing errors, and therefore they are systematic. However, because of the way these faults manifest themselves it could also be argued they are random. For example, although human error may be at the root of the problem, it is not reasonable to assume they could have been systematically eliminated ahead of time. Therefore, we are left with a random discovery process based on the fact that all the processors in use are operating simultaneously with different applications and different data. There is a particular probability that one of them will discover a fault, or errata.

2 List of major assumptions

i. Errata present in the Pentium could not reasonably have been systematically eliminated ahead of time. Errata have to be discovered and the discovery process is assumed to be random and therefore amenable to statistical modeling.
ii. The rate of discovery of errata follows a constant failure rate model, and is independent of any changes that are made as a result of discovering previous errata.
iii. The number of processors in operation follows a normal distribution.
iv. The probability of discovering errata is directly proportional to the number of processors in operation.
v. With reference to [2], all *fault types* must be considered. Only 83.3% of the *configuration types* are considered. Only the 89.1% under the *normal operational mode* are valid. All of the *dependency* events will be considered. Only the 31.9% *lesser severity* events are considered, because the *denial of service, hang, and crash* events do not result in a hazard threat.
vi. The categories are orthogonal.
vii. The rate of discovery of new errata during each month applies uniformly over that month.
viii. There are 10 million Pentium II processors in operation at the present time, all of which are operating independently, and there is a uniform probability that any one of these processors could discover errata.
ix. There are 85 TAS platforms in a given ATC system.

3 Main hazard

The main hazard can be identified as **action taken on an unintended telegram** that exposes to the system to a safety-related incident. In this case the hazards are caused by threats from undiscovered errata in the operation of the Pentium

processors in the TAS Platform. The target value for the probability of the main hazard is 10^{-9} per train-hour of operation.

4 Threats from the processor

The main hazard will result from threats imposed by undiscovered errata in the processor. Therefore, the probability of the hazard existing amounts to calculating the probability that the threats exist, as modified by the defense mechanisms available to mitigate this threat. The following analysis derives a model of the process, which demonstrates that the probability of threats from the errata is within the target value.

5 Errata probability models

There are a number of failure rate models that apply to hardware failures in electronics equipment [1]. These range from a simple constant-rate model to more sophisticated models that take into account repairs and upgrades to the equipment. The situation with processor errata, however, is that they are not simply hardware failures that occur as a result of normal operation. Errata have to be uncovered by events that trigger them through the operation of the processor itself in an otherwise perfectly normal manner, using a wide range of data input. Nevertheless, we can base an errata model on the normal failure rate models.

5.1 Constant rate model

The constant failure rate model assumes that the probability of a failure is constant throughout the lifetime of the equipment and that it does not change as a result of any action take to improve the equipment. The failure rate is given by,

$$h = \frac{1}{\lambda}, \tag{1}$$

where λ is commonly called the mean time to failure (MTTF). The probability density function over time for the constant-rate model is given by,

$$p(t) = \frac{1}{\lambda} e^{-t/\lambda}. \tag{2}$$

The corresponding cumulative distribution function is given by,

$$P(t) = \int_0^t p(t')dt' = 1 - e^{-t/\lambda}. \tag{3}$$

These functions are shown in figures 1 and 2. The probability of a device failing is greatest at the start of its life and diminishes as time goes on. This is a reasonable assumption for the discovery of processor errata, but it assumes a constant sample size, that is to say the failures are not a function of the number of units in operation. On the other hand it is reasonable to assume that the rate of discovery of processor errata is proportional to the number of processors in operation, therefore, the constant-rate model cannot be used directly.

5.2 Model with learning

Another model for equipment failure is known as the Weibull model and it accounts for the fact that the failure rate may change over the lifetime of the device because of improvements made as a result of previous failures (e.g. changing the operating environment to reduce the number of failures). The parameter β is introduced to account for the reduced, or even increased, rate of failures. Thus,

$$h(t) = \frac{\beta}{\lambda}\left(\frac{t}{\lambda}\right)^{\beta-1}, \qquad p(t) = \frac{\beta}{\lambda}\left(\frac{t}{\lambda}\right)^{\beta-1} e^{-(t/\lambda)^{\beta}} \tag{4}$$

and

$$P(t) = 1 - e^{-(t/\lambda)^{\beta}}. \tag{5}$$

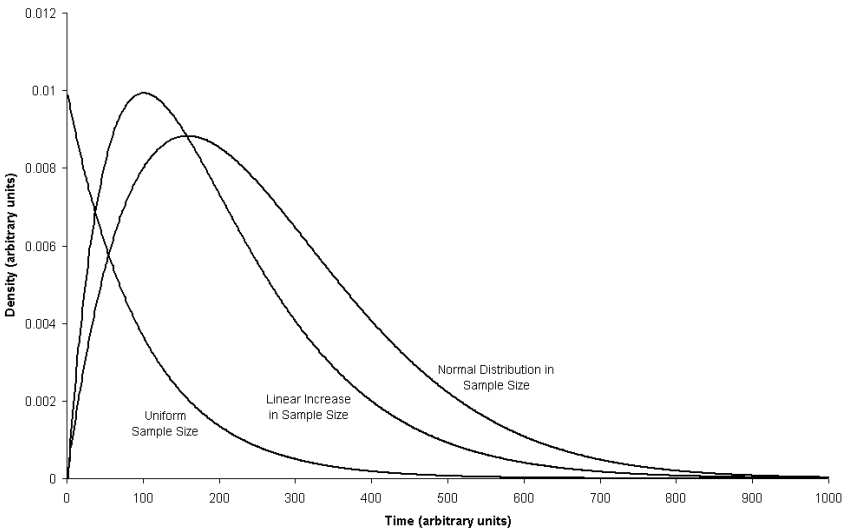

Figure 1: Probability density distribution of errata discovery.

The Weibull model is not really an appropriate model for processor errata because it is reasonable to assume that the rate of failure, meaning the rate of discovery of faults, is independent of any changes that are made as a result of discovering previous failures. The value of β is typically between 0.95 (improvement) and 1.05 (worsening). A value of 1.0 gives the constant rate model.

5.3 Model with linear sample size

In order to have a realistic errata discovery rate model we need to combine the constant rate model with a reasonable model of the number of processors in operation. If we assume a linear growth in the number of units contributing to the process of discovering errata, then we can write,

$$n = Nt, \tag{6}$$

where n is the number of processors and N is the rate of growth. There is a tacit assumption here that there is a limit reached at some time, beyond which there are no further units (or at least some other model applies). We now assume that the probability of discovering errata is directly proportional to the number of units contributing, therefore,

$$p(t) = \frac{N}{\lambda} t e^{-t/\lambda}, \tag{7}$$

$$P(t) = N\int_0^t (t/\lambda)e^{-t/\lambda}dt = N\lambda\int_0^{t/\lambda} t' e^{-t'}dt' = N\left(1-\left(1+\frac{t}{\lambda}\right)e^{-t/\lambda}\right). \tag{8}$$

The results are also shown in figures 1 and 2. The rate of discovery early in the device life is clearly reduced because of the small number of units in operation. However, the model is still not very realistic because there is no limit to the number of units in operation and it is not reasonable to assume the rate of sales of Pentium processors was in fact linear.

5.4 Model with realistic sample size

A more realistic model for the number of processors in use is to assume a normal distribution where there is a small number at first, a large number during the mature period of sales, and a small number again at the end of the production life. Such a model is given by,

$$n = Ne^{-((t-t_o)/\tau)^2}, \tag{9}$$

where $\tau = \sqrt{2}\sigma$ and σ is the standard deviation, t_o is the mean and N is a normalizing factor (depending on the total units sold). Then,

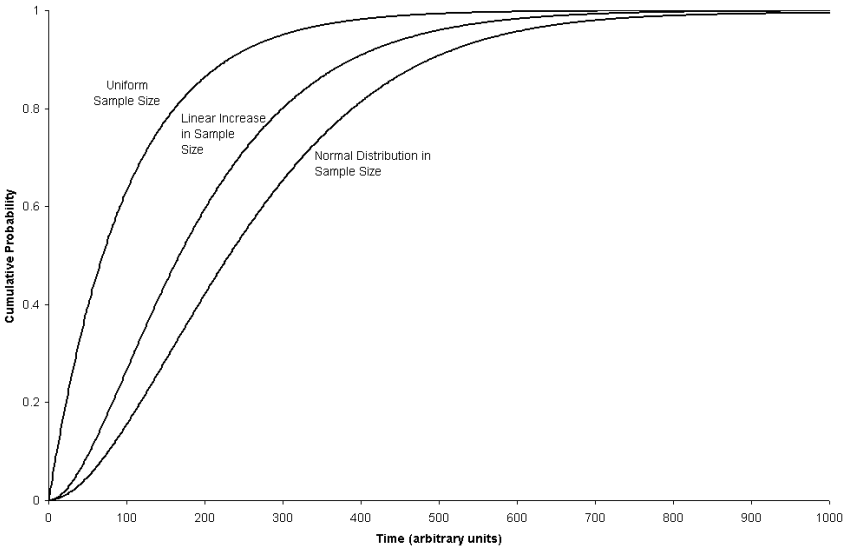

Figure 2: Cumulative distribution of errata discovery.

$$p(t) = \frac{N}{\lambda} e^{-((t-t_o)/\tau)^2} e^{-t/\lambda}, \tag{10}$$

$$P(t) = \frac{N}{\lambda} \int_0^t e^{-((t'-t_0)/\tau)^2} e^{-t'/\lambda} dt'. \tag{11}$$

Figure 3 illustrates the cumulative distribution of processors in use. In addition, figures 1 and 2 show how the new model is more reasonable and we will show next that it is a very good fit to actual errata rate data.

6 Processor errata data

The behaviour of the Pentium II processor up to April 1999 is examined in detail in a study of microprocessor entomology [2], and its application to high-confidence computing systems [3]. Not all of the errata discovered for the Pentium II processor are threats to the main hazard, and the results reported in [2] are used here as both the source of data for the rate of discovery of errata and to determine whether the errata discovered are of the type that can cause a hazard.

6.1 Errata discovered to date

The cumulative distribution of errata reported over the period May 1997 to April 1999, taken from [2], is shown in figure 4. There are a total of 73 errata, and it should be noted that 27 of them were reported immediately.

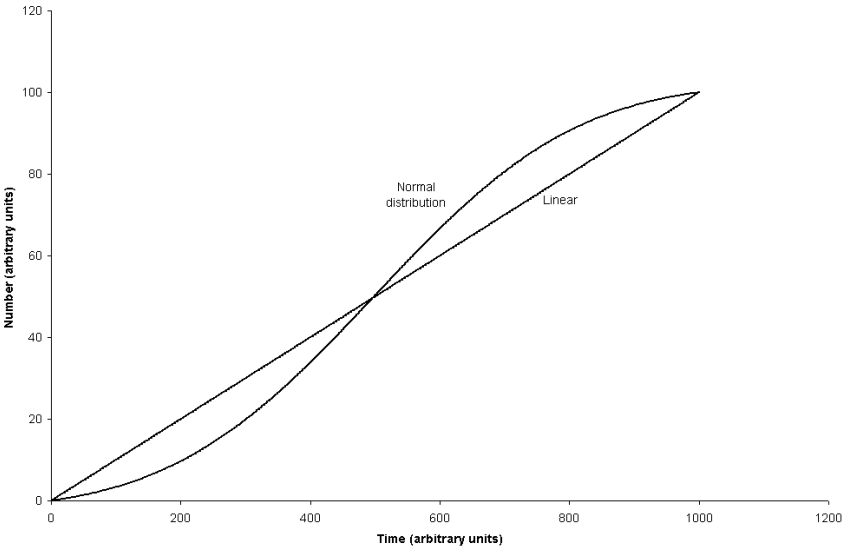

Figure 3: Cumulative distribution of units in operation.

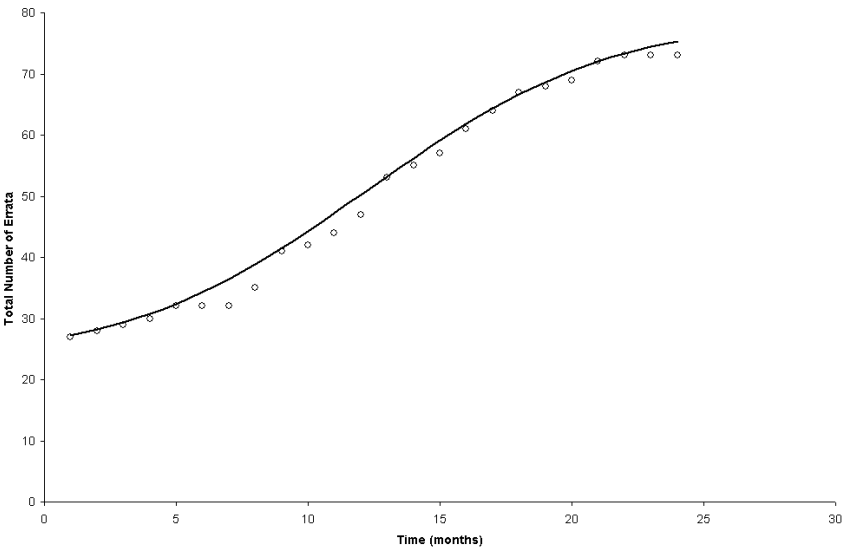

Figure 4: Errata reported from May, 1997 to April, 1999.

6.2 Model fit

A curve of the type described in 5.4 is shown in figure 4 fitted to the data. Note that a worst-case assumption is made in fitting the curve to the upper data

samples on the graph. Also the initial 27 are treated as a delta function in the density distribution and show up as an offset in the curve. The parameters that provide a good fit are as follows:

$N = 3.3$, which is the maximum number of errata reported in a month,
$t_0 = 13$, which is the mean in months of units in operation,
$\tau = 10$, which is corresponds to a standard deviation of 7.1 months for units in operation,
$\lambda = 15$, which is the "mean time to fault discovery" in months,
Offset = 26.5, which accounts for the initial delta function.

This model predicts that there should have been 75.26 reported errata by April 1999, as opposed to the actual value of 73.

6.3 Expected future errata

Using the curve fitted to the data we can calculate the maximum expected number of errata for the Pentium II throughout its lifetime. To do this we simply extend the curve to very large values of time. The result is 77.86. This means that 2.6 more errata are to be discovered during the remaining lifetime of the Pentium II processor (since April 1999). As a worst case, however, we assume that there are 77.86 – 73 = 5 (rounded up to the nearest integer) more errata to be found than have currently been reported. Note, that by 2006 virtually all 5 should have been found, but since there are no records beyond April 1999 we must assume that all 5 are still waiting to be found. We will discuss this assumption further later on.

6.4 Classification of errata

The data in [2] is also presented by the classification of errata. The classifications used are *fault type, configuration, operational mode, dependency* and *severity*.

We will assume here that all *fault types* must be considered. For the *configuration*, only the 83.3% of the errata under the *uniprocessor configuration* are valid candidates. The *multiprocessor* and *F.R. checking* categories are not valid since the TAS Pentium IIs are not operated in these modes while doing ATC functions. Similarly, only the 89.1% found under the *normal operational mode* are valid, as opposed to *starting, test, recovery,* and *system management* modes. All of the *dependency* events will be considered, but only the 31.9% *lesser severity* events are considered. This is because the *denial of service, hang, and crash* events do not result in a hazard threat.

If we assume these are independent (orthogonal) categories then the fraction of the new errata to be found that could be a threat can be computed as,

$$F = 1.0 \text{ x } 0.833 \text{ x } 0.819 \text{ x } 1.0 \text{ x } 0.319 = 0.218 \text{ or } 21.8\%. \quad (12)$$

The data is further categorized according to where the errata occurred in the processor hardware. Of the errata, 50.5% were in the performance *delivery architecture*, and are valid candidates, while 49.6% were in the *confidence*

assurance architecture and therefore do not present a threat. Therefore, we can modify the fraction by the factor 0.505,

$$F = 0.505 \text{ x } 0.218 = 0.119 \text{ or } 12\%. \quad (13)$$

In summary, based on historical data, only 12% of the errata could have potentially caused a threat.

7 Threat model based on non-redundant processors

This section calculates the probability that a threat exists assuming that there is no advantage to be gained from the 2-out-of-3 (2oo3) processor redundancy that is implemented on the TAS Platform. The probability is based entirely on the above model derived to fit to the reported data [1].

7.1 Target hazard rate

The target probability for the main hazard is,

$$P_h = 10^{-9} \text{ per train hour.} \quad (14)$$

In the context of this analysis we interpret this to mean that there is a probability of P_h that a particular train will experience a hazard in any one hour period due to a threat from undiscovered errata in the Pentium II processors associated with that train during the hour.

7.2 Existing errata with no fix

There are 44 known errata that are listed as having "no fix" [2]. In the context of this analysis this means that there are 44 known errata that could be triggered by the Pentium processors. We consider known errata to be faults of a systematic nature. These are not amenable to statistical analysis and the defenses against the threats posed by such faults are evaluated qualitatively. None of the errata reported to date for the Pentium II, which do not have a fix, impact the TAS platform in a safety critical manner.

7.3 New errata

According to the model developed in section 5, we assume there are 5 errata yet to be discovered in the lifetime of the Pentium II.

7.4 Errata causing a threat

Of the errata yet to be discovered, only the fraction $F = .12$ will cause a threat to the main hazard, according to the classifications in [1]. Therefore the number of threatening errata yet to be discovered is,

$$N_e = 5 \text{ x } F = 5 \text{ x } 0.12 = 0.6 \text{ .} \quad (15)$$

7.5 Probability of discovering a threat per operating hour

In the interim period from April 1999 to July 2006 no errata have been identified. It does not matter whether this is because they were not discovered, not reported or simply that the records were not updated. Our model assumes they are yet to be discovered, and the potential threat of concern is they will be discovered by one of the TAS platform Pentiums in an ATC project, given all the devices in operation at the time. What this means for our model is that we can remove a block of time from April 1999 to, say, July 2006 and continue the probability model at the later time. We can then calculate the probability of errata being discovered during any one hour over the time period of interest starting in July 2006 (or another suitable starting date).

According to our model of the errata discovery process, the rate of discovery is a maximum at the start of the time period and decreases with each succeeding month. The value for the first month is 0.678 errata per month. If we assume this applies uniformly over the month then the rate per hour is 0.678 / (30 x 24) = 0.000942. The rate after 5 years is virtually zero (less than 10^{-27}). This means that the worst-case rate of discovery of errata that will cause a threat is,

$$P_e = 0.6 \text{ x } 0.000942 = 0.000565 \tag{16}$$

per hour, and this corresponds to the first hour of operation beyond April 1999.

7.6 Probability of threat per train processor

Our model includes all of the Pentium II processors operating in the world, including those in a particular ATC project. These processors are operating on either different applications or different data on similar applications. Either way it is reasonable to assume that all of these processors are operating independently and there is a uniform probability that any one of these processors could discover errata.

Of interest here is the probability that one of the TAS platform Pentiums will discover the errata. We assume there are 85 TAS platform modules planned for a typical ATC system. There are conservatively 10 million Pentium II devices in operation, all of which are performing calculations more or less continually as are the TAS platform Pentiums. Therefore the probability that a TAS platform Pentium II will be the unlucky candidate is,

$$P_{TP} = 85 / 10^{7} = 8.50 \text{x} 10^{-6} . \tag{17}$$

According to the assumptions of this section the operation of the redundant processors in the TAS platform module provides no advantage. There are three Pentium devices configured to detect independent random failures using a 2oo3 majority-voting scheme. At one extreme we can argue that since the three devices operate with identical software on identical data there is a high probability that they would all discover the same errata at the same time and the

redundancy is of no use. Therefore, for this section of the analysis the three devices will be considered as one and the probability of the redundancy failing is by default,

$$P_f = 1.0 . \tag{18}$$

We can now calculate the probability of one of the ATC TAS platform devices encountering threatening errata in the first hour of operation as,

$$P_t = P_f P_e P_{TP} = 4.81 \text{ x } 10^{-9} . \tag{19}$$

7.7 Probability of hazard

Encountering threatening errata does not necessarily present a hazard. Even if an erroneous telegram is transmitted, and there are no errors in transmission, the received telegram must undergo some checks before it is accepted. Of the many checks performed we cannot rely on any of those involving data appended to the telegram after it is generated. These include sequence numbers, CRC checksums, authentication hashes or encryption. The only candidates for protection against a threatening telegram are the persistency and consistency checks. A threat can only become a hazard if these checks fail.

Again, according to the assumptions of this section, the operation of these telegram checks, at this time, is assumed to provide no advantage. Therefore, the probability of the checks failing on all three Pentium devices is by default,

$$P_{cf} = 1.0 . \tag{20}$$

The final calculation for the probability of a hazard during the first hour of operation is then,

$$P_h = P_t P_{cf} = 4.81 \text{ x } 10^{-9} . \tag{21}$$

Using the model for errata discovery we can plot the probability over time as shown in figure 5. Note that the target of 10^{-9} is met after 5 months of operation. This will cover the period of testing before the system goes into operational service.

8 Conclusions

We have defined a main hazard as action taken on an unintended telegram, and mathematically modeled the probability of occurrence due to undiscovered Pentium processor errata that are a threat to the system. We have used standard probability distribution functions with parameters selected by fitting the model to actual field data in order to predict the number of threatening errata yet to be

discovered. With worst case assumptions the results indicate that the probability of a hazard meets the target of 10^{-9} per train hour of operation within 5 months from power-up (start of computing), a period of time which will easily be covered by system testing and commissioning and long before the system goes into revenue service.

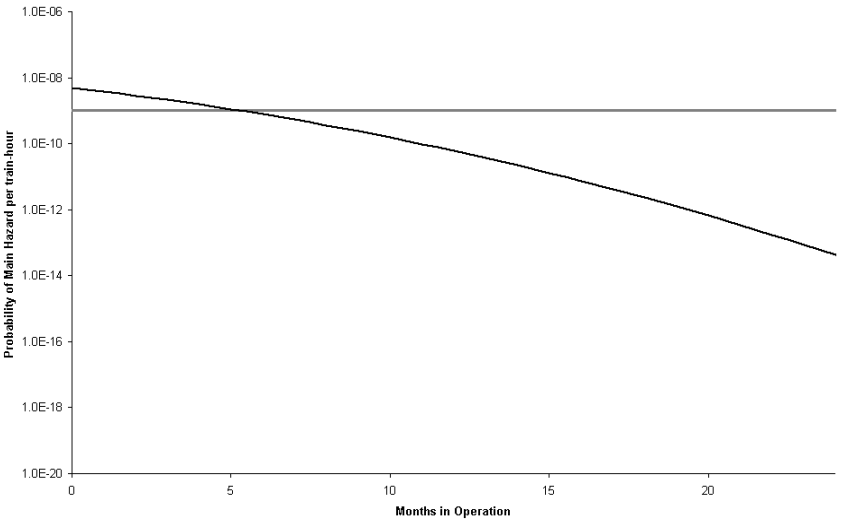

Figure 5: Probability of occurrence of the main hazard per train-hour of operation based on only the rate of discovery of errata in the Pentium II processor.

9 List of variables

h = failure rate ($1/h$ = MTBF)
$p(t)$ = probability density function
$P(t)$ = cumulative probability function
β = learning factor
n = number of processors in operation
N = rate of increase in the number of processors in operation
t_0 = mean (in months) of units in operation
$\tau = \sqrt{2\sigma}$
σ = standard deviation (in months) for units in operation
λ = mean time to fault discovery (in months)
F = fraction of errata that are threatening
P_h = main hazard probability (per train hour)
N_e = number of threatening errata to be found
P_e = probability of a processor discovering errata
P_{TP} = probability of a TAS platform processor being involved

P_f = probability of redundancy process failing
P_t = probability of TAS platform processor discovering threatening errata
P_{cf} = probability of data checking process failing

References

[1] Zang, H., Cutright, E., Giras, T., Time-varying failure rate for system reliability analysis in large scale railway risk assessment simulation, Computers in Railways IX, WIT press, 2004, pp313-322.
[2] Avizienis, A., He, Y., Microprocessor Entomology: A Taxonomy of Design Faults in COTS Microprocessors, Dependable Computing for Critical Applications 7, 1999, 6-8 Jan 1999, pp 120-121.
[3] He, Y., Avizienis, A., Assessment of the Applicability of COTS Microprocessors in High-Confidence Computing Systems: A Case Study, Proc. International Conference on Dependable Systems and Networks, 25-28 June 2000, pp 492-500.