Technical University Berlin

Telecommunication Networks Group

Markov Modeling of PROFIBUS Ring Membership over Error Prone Links

Andreas Willig

willig@ft.ee.tu-berlin.de

Berlin, May 1999

TKN Technical Report TKN-99-004

TKN Technical Reports Series
Editor: Prof. Dr.-Ing. Adam Wolisz

Abstract

The PROFIBUS is a well-known and widely used fieldbus. On the MAC layer it uses a token passing protocol on top of a broadcast medium, the protocol is similar to the IEEE 802.4 Token-Bus. All stations form a logical ring, which is created and maintained by the exchange of special control frames. In this paper we investigate the stability of the ring in the presence of transmission errors, which may hit the control frames. An analytical model is developed, which captures the ring membership behaviour of the PROFIBUS protocol. As the main modeling tool we use a discrete time markov chain. This model is validated against simulations, and it is shown, that it predicts some important stability measures very accurately.

Contents

Chapter 1

Introduction

In this paper we describe and evaluate an analytical model for the ring membership behaviour of the PROFIBUS MAC protocol over error prone links. This protocol uses token passing similar to IEEE 802.4, where a logical ring is built on top of a broadcast medium. The aim of this analytical model is to gain some insight in the ring stability of the PROFIBUS over error prone links[1]. Roughly speaking, the term ring stability (or instability) refers to the times, where not all stations are member of the ring, although they want to be. Stations can get lost from the ring e.g. due to transmission errors in management frames. It turns out, that the ability to perform a "hearback" on the medium (i.e. a station can read the results of its own transmissions on the ring and determine, whether an error has occured) is critical to the stability of the ring.

The paper is structured as follows: in the next chapter 2 we describe the PROFIBUS token passing protocol and shortly discuss the main causes for ring instability. In chapter 3 we define some measures for ring stability. In chapter 4 we derive our analytical model, using a discrete time markov chain as the main modeling tool. This model is validated against simulations in chapter 5. Our conclusions are given in chapter 6.

[1]An analysis of the IEEE 802.4 Token Bus with error prone links can be found in [5].

Chapter 2

Token Passing and Ring Maintenance Protocol

In this section we give a brief description of the token passing and ring maintenance protocol of the PROFIBUS, followed by an explanation of the mechanisms that may lead to station losses from the ring. The PROFIBUS is defined in the german national standards [1], [2] and in the additional documents [9], [7]. A european standard is also available under [10].

PROFIBUS is a well known and widely used field bus, which is also standardized ([4], with some corrections in [8], the european standards document is [10]). It is designed for delivering real-time services in a harsh, industrial environment. The PROFIBUS supports this on the MAC layer by using a token passing

2.1 Protocol

The PROFIBUS token passing protocol uses a broadcast medium. A logical ring is formed by ascending station addresses. The address space is small, a station address is in the range of 0 to 126. Every station (denoted as TS: This Station) knows by the ring maintenance mechanism the address of its logical successor (NS: Next Station) and its logical predecessor (PS: Previous Station). If TS receives a token frame with TS as destination address, it checks whether it was sent by its PS. If so, the token is accepted, otherwise the token frame is discarded. In the latter case, if the same token frame is received again as the very next frame, the token is accepted and the token sender is registered as new PS. In any case after accepting the token TS may send some data and then must pass the token to NS. After sending a token frame, TS listens on the medium for some activity. This can be reception of a valid frame header (indicating that NS has accepted the token) or reception of some erroneous transmission. However, TS listens on the medium only for a short time (called *slot time*) which is typically chosen very sharp, e.g. in the range of 100 μsec to 400 μsec. If this time passes without any bus activity the token frame is repeated. If there is again no activity, NS is assumed to be dead and TS determines the next station in the ring (i.e. the successor of NS), makes this the new NS and tries to pass the token to it, following the same rules. The new station can be determined from information gathered during ring maintenance (LAS), see below. If TS finds no other station, it sends a token frame to itself. A special requirement is the following: TS must read back its own token frame while transmitting ("hearback"), in order to detect a defective transceiver. If TS encounters a difference the first time, it behaves as after a correct token telegram, i.e. it waits for some response. If there is no activity on the medium it repeats the token telegram. If TS again encounters a difference, it discards the token immediately and removes itself from the ring, behaving as newly switched on and "forgetting" all knowledge previously obtained.

If TS is newly switched on, it is required to first listen passively on the medium, until it has received two successive identical token cycles (this state is referred to as "Listen-Token" state). During this time it is not allowed to send or answer to data frames or to accept the token. Every station address found in a token frame belonging to this two cycles is included into a locally maintained *list of active stations* (LAS). After that TS can enter the ring if it is included by its predecessor. In addition, TS is required to maintain its LAS by inspecting every received token frame. An important rule in this maintenance process is the following: if TS feels itself as already included in the logical ring and reads a token frame, where TS is "skipped" (i.e. the address of TS lies truly within the address range spanned by sender and receiver of the token frame) it removes itself from the ring and behaves as newly switched on.

For intentionally leaving the ring it suffices to just stop all transmissions, then PS will detect the station loss when passing the token. Including a new station in a ring is more complicated. Every station maintains a gap list (GAPL), containing all station addresses between TS and NS. TS is required to scan periodically all stations in GAPL by explicitly sending a special request frame to a single address and waiting for an answer, indicating the type of the station and its current status (ready / not ready for the ring). A station which tries to detect two identical token cycles will respond with a "not ready" status. Within every token cycle TS pings at most one station address in GAPL (using the "Request-FDL-Status"-Frame). If a station in GAPL responds as "ready" TS will change its NS, shorten its GAPL, update its LAS and then sends a token frame to the new station. The period for scanning the GAPL is created by a special timer ("gap timer"), which is set as an integral multiple ("gap factor", the standard requires values between 1 and 100) of the target token rotation time (TTRT).

A special mechanism is used for the very first ring initialization or for token loss due to system crash of the current token owner: every station listens permanently on the bus. If there is no bus activity for some time (the corresponding timer is called *timeout timer*), TS "claims the token", i.e. it starts transmitting either data or a token frame. The timeout value is linearly dependent on the stations address. This timeout timer is one of the reasons for introduction of the hearback feature: the latter is necessary for resolving collisions, which may occur e.g. when two stations are newly switched on and their timeout timers expire at the same time, leading to a collision and retirement of all colliding stations. Subsequently the timeout timers will expire in strict order, leading to a valid ring.

2.2 Major Causes for Instability

During our investigation we found two basically different scenarios, where station losses may happen. The first scenario is triggered by the fact, that the token telegram has no checksum. It is armed only with parity bits and start- and stopbits incorporated in transmission using an UART. Thus there is a non-negligible probability that a token frame can be corrupted such that no station except the sender will experience an error. Consider now the case with two stations z and a with $z < a$ with respect to ring ordering. If z sends a token telegram to a where the destination address is corrupted and equal to b with $a < b < z$ (w.r.t. ring ordering), a feels itself being skipped and removes immediately from the ring (we refer to this as "error skipping"). If z retransmits the token, a has not yet built a valid LAS and does not accept the token. After another retransmission z is sure that a is lost. The second scenario occurs under the presence of the "hearback" feature. It can be described as follows: when a station a experiences errors in two successive trials to send a token telegram, it removes itself immediately from the ring and makes no further transmission. The result is, that the medium is idle, until the timeout timer of the station with the lowest address expires. Furthermore, a forgets all of its state, especially the LAS. Within this scenario two cases can be distinguished: whether a has the lowest station address or not. If a has the lowest address, it is the timeout timer of a that expires. Since there was no transmission during the idle time and a has forgotten its LAS, a now thinks it is alone in the ring and sends a token telegram to itself. Then, by the first scenario all other stations remove themselves from the ring, being skipped. We refer to the latter scenario as "ring-jacking". If a has not the lowest address, the remaining ring keeps alive and a is included later.

Chapter 3

Ring Stability Measures

Be K the number of stations and $\{N(t)\}_{t\in\mathbb{R}}$ a family of integer-valued random variables, denoting the number of stations, which are member of the ring at time t (more precisely: which feel themselves being member). We have $0 \leq N(t) \leq K (t \in \mathbb{R})$ and $N(t)$ changes only at discrete points in time, by the operation of the protocol. For simplicity we assume that there is no data traffic on the ring and that all stations want to be member of the ring all the time. We introduce the following global measures for ring stability:

- Consider that at time t we have $N(t) = K$ and $N(t^-) < K$. We are interested in the mean value and distribution $C(s) = \Pr[C \leq s]$ of the time duration C that the ring is complete, before the next time it looses a station.

- We are also interested in the "dual" of $C(s)$, namely in the distribution of the time that is necessary to re-enter the state of a full ring after the full ring breaks. The distribution of these delays is named $D(s)$.

- Mean number of stations in the ring during interval $[0, t]$:

  $$\bar{N}(t) = \frac{1}{t}\int_0^t N(s)ds$$

  additionally we are interested in the steady state mean value $\bar{N} = \lim_{t\to\infty} \bar{N}(t)$

- Fraction of time where not all stations are being member of the ring during time interval $[0, t]$:

  $$\bar{M}(t) = \frac{1}{t}\int_0^t \chi_{N^{-1}(K)}(s)ds$$

  where $\chi_{N^{-1}(K)}$ is the characteristic function for the set $\{t \in \mathbb{R} : N(t) = K\}$. Additionally we are interested in the steady state fraction $\bar{M} = \lim_{t\to\infty} \bar{M}(t)$.

Furthermore, for every single station i we are interested in the distribution of times between station losses for this station, the duration of station outages and the overall fraction of time that i is not member of the ring. However, these "local" stability measures cannot be obtained with our analytical model, only with simulations. Some simulation results for these measures can be found in [11].

In this paper we focus mainly on the parameters $\bar{M}(t)$, $\bar{M}$, $\bar{N}(t)$ and $\bar{N}$.

Chapter 4

The Analytical Model

In the following we describe an analytical model capturing the behaviour of a set of PROFIBUS stations w.r.t. ring membership and giving the desired performance measures. This model depends on some model parameters, for which in turn estimates are derived using only the protocol description, some static system parameters and a single correction term.

4.1 Model

The approach is to derive a discrete time markov chain (DTMC) from a somewhat simplified station life cycle. This DTMC initially has a four-dimensional state description, which, however, can be flattened into a one-dimensional and then be solved for the steady-state probability vector.

The stations life cycle is shown in figure 4.1 as a set of states, which the station visits in some order within its life. The state *Listen-Token* corresponds to the state, where a station listens to the medium, waiting for two successive identical token cycles and not being able

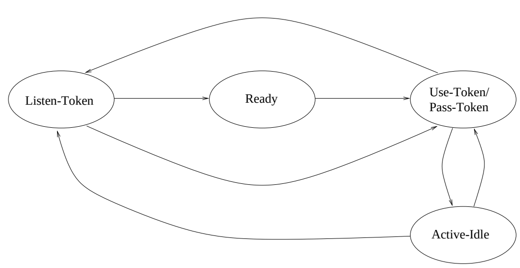

Figure 4.1: A single stations life cycle

to enter the ring (with the exception of timeout timer expiration). In state *Ready* a station has successfully received two successive identical token cycles and thus a valid LAS, however, it waits for being included in the ring. In the state *Use-Token/Pass-Token* the station is member of the ring and the current token owner, either performing some data transmission, pinging a station in the gap list or trying to pass the token to its successor. Finally, in state *Active-Idle* the station is member of the ring, but not the current token owner. All possible transitions can be explained by normal protocol operation, only the transition from Active-Idle to Listen-Token needs the error conditions explained in section 2.2. In order to obtain a markovian model, we make the following assumptions:

- Time is slotted, slots have fixed length. This assumption is reasonable, if there is no load in the system and only token frames and request frames for ring inclusion are exchanged. One slot corresponds to one token frame including the slot time.
- The system is assumed to be time-homogeneous.
- All state transitions occur, if currently possible, for every single station with fixed probability p_{XY}, where X and Y denote shortly the source and target state (e.g. p_{LU}), exceptions see below. These fixed probabilities depend on environmental conditions, e.g. the error rate. The stations are independent from each other.
- The protocol allows for two different transitions from the state Ready to the state Use-Token: in the first there is no token owner and a station in state Ready experiences a timeout, in the second a station in Ready state is explicitly included by the current token owner. For the first transition type we assume that all stations are independent and that every stations makes this transition with probability p_{RU}, while for the second transition type only a single station can be included, this happens with the state-dependent probability $p_I(i, j)$, where i denotes the number of current ring members and j denotes the number of stations which are in state Ready.
- The probability $p_{LR}(n)$ for transition from the state Listen-Token to the state Ready depends on n, where n is the sum of the numbers of stations in Use-Token and in Active-Idle (see below).
- Bit errors occur independently with fixed rate.
- Station addresses are uniformly distributed in $[0, 126]$

- If token frames are transmitted correctly, either a transition from Listen-Token to Ready or a transition from Ready to Use-Token is possible (only one of them at a time); if the token is erroneous, there can be a hearback error, causing a station to leave the ring, and additionally it can happen that other stations are skipped (these events do not exclude each other).

- For experiencing a hearback error, we assume that a single erroneous token suffices, however this occurs with a probability corresponding to two successive erroneous token frames. Furthermore we assume that the event of an active station being "skipped" by token frames with unfortunate error patterns is tied to the event of experiencing a hearback error. This assumption is reasonable, since this event is relatively rare.

We construct a DTMC with a four-dimensional state space

$$\mathcal{S}(K) = \{(L, R, U, A) \in \mathbb{N}_0^4 | (L + R + U + A = K) \wedge (U \leq 1)\}$$

where for $s = (L, R, U, A) \in \mathcal{S}(K)$ we define

- $L = L(t) = \#$ Stns in Listen-Token at time t
- $R = R(t) = \#$ Stns in Ready at time t
- $U = U(t) = \#$ Stns in Use-Token at time t
- $A = A(t) = \#$ Stns in Active-Idle at time t

for every $t \in \mathbb{R}$. However, we will omit the time argument. It is easy to see that $|\mathcal{S}(K)| = K^2 + 2K + 1$ holds.

Now we describe the possible state transitions. A state $x_n = (L, R, U, A)$ denotes the state of the system at the n-th time slot, $x_n \in \mathcal{S}(K)$. In the following we enumerate for every current state the possible transitions and give their respective probabilities. Where missing, the probability for staying within a state is implicitly defined by $(1 - \sum(\text{all other trans. prob.}))$. A concrete system usually starts with $x_0 = (K, 0, 0, 0)$.

- $L = K, R = 0, U = 0, A = 0$:

  - Exactly one station experiences a timeout:

    $$\Pr[x_{n+1} = (K-1, 0, 1, 0) | x_n = (K, 0, 0, 0)] = b(1; K, p_{LU})$$

    where $b(k; n, p) = \binom{n}{k} p^k (1-p)^{n-k}$ is the distribution function of the binomial distribution.

- $L = K-1, R = 0, U = 1, A = 0$:
  - The token owner experiences a hearback error:

$$\Pr[x_{n+1} = (K, 0, 0, 0) | x_n = (K-1, 0, 1, 0)] = p_{UL}$$

  - A number of stations enters the Ready state:

$$\Pr[x_{n+1} = (K-1-k, k, 1, 0) | x_n = (K-1, 0, 1, 0)] = (1 - p_{UL}) \cdot b(k; K-1, p_{LR}(1))$$

    where $k \in \{0, \ldots, K-1\}$ holds.

- $L = i, R = j, U = 1, A = 0$ with $i + j + 1 = K$ and $j \geq 1$:
  - The token owner experiences a hearback error:

$$\Pr[x_{n+1} = (i+1, j, 0, 0) | x_n = (i, j, 1, 0)] = p_{UL}$$

  - A number of stations enters the ready state ($i \geq 1$):

$$\Pr[x_{n+1} = (i-k, j+k, 1, 0) | x_n = (i, j, 1, 0)] = (1 - p_{UL}) \cdot (1 - p_I(1, j)) \cdot b(k; i, p_{LR}(1))$$

    where $k \in \{0, \ldots, i\}$ holds.

  - A single ready station enters the ring and becomes token owner:

$$\Pr[x_{n+1} = (i, j-1, 1, 1) | x_n = (i, j, 1, 0)] = (1 - p_{UL}) \cdot p_I(1, j)$$

- $L = i, R = j, U = 0, A = 0$ with $i + j = K$ and $j \geq 1$:
  - A single station in ready state experiences a timeout (no transitions from Listen-Token to Ready can happen):

$$\Pr[x_{n+1} = (i, j-1, 1, 0) | x_n = (i, j, 0, 0)] = b(1; j, p_{RU})$$

  - A single station in Listen-Token experiences a timeout ($i \geq 1$):

$$\Pr[x_{n+1} = (i-1, j, 1, 0) | x_n = (i, j, 0, 0)] = b(1; i, p_{LU})$$

- $L = i, R = j, U = 1, A = k$ with $i + j + k + 1 = K$ and $k \geq 1$:
  - The token owner experiences a hearback error and by that way a number of stations in state active idle feel themselves skipped:

$$\Pr[x_{n+1} = (i+1+\nu, j, 0, k-\nu) | x_n = (i, j, 1, k)] = p_{UL} \cdot b(\nu; k, p_{AL})$$

    where $\nu \in \{0, \ldots, k\}$ holds.

– A number of stations enters the ready state ($i \geq 1$):

$$\Pr[x_{n+1} = (i-\nu, j+\nu, 1, k) | x_n = (i, j, 1, k)] = \begin{cases} (1-p_{UL}) \cdot (1-p_I(k+1, j)) \cdot b(\nu; i, p_{LR}(k+1)) & : \quad j > 0 \\ (1-p_{UL}) \cdot b(\nu; i, p_{LR}(k+1)) & : \quad j = 0 \end{cases}$$

where $\nu \in \{0, \ldots, i\}$ holds.

– A single ready station enters the ring ($j \geq 1$):

$$\Pr[x_{n+1} = (i, j-1, 1, k+1) | x_n = (i, j, 1, k)] = (1-p_{UL}) \cdot p_I(k+1, j)$$

- $L = i, R = j, U = 0, A = k$ with $i + j + k = K$ and $k \geq 1$:

  – A station in Listen-Token experiences a timeout and by that way kicks all stations in Active-Idle out of the ring ($i \geq 1$):

$$\Pr[x_{n+1} = (i-1+k, j, 1, 0) | x_n = (i, j, 0, k)] = b(1; i, p_{LU})$$

  – A station in Ready experiences a timeout ($j \geq 1$):

$$\Pr[x_{n+1} = (i, j-1, 1, k) | x_n = (i, j, 0, k)] = b(1; j, p_{RU})$$

  – A station in Active-Idle experiences a timeout:

$$\Pr[x_{n+1} = (i, j, 1, k-1) | x_n = (i, j, 0, k)] = b(1; k, p_{AU})$$

This model has a four-dimensional state space. However, common solution techniques for the steady state of markov chains require one-dimensional markov chains. Thus we need to find a bijective mapping h from our four-dimensional state space $\mathcal{S}(K)$ to a one-dimensional state space of exactly the size $|\mathcal{S}(K)| = K^2 + 2K + 1$. This mapping can be explicitly constructed by simply enumerating the whole space $\mathcal{S}(K)$ and assign every state a distinct natural number. If P denotes the time-homogeneous state transition matrix for the one-dimensional markov chain, then we can determine the steady state probability vector $\pi = (\pi_1, \ldots, \pi_{|\mathcal{S}(K)|})^T$ as usual by solving the following system of linear equations:

$$\begin{aligned} \pi^T &= \pi^T \cdot P \\ \sum_{i=1}^{|\mathcal{S}(K)|} \pi_i &= 1 \end{aligned}$$

where A^T denotes the transposed matrix of matrix A. After determining π we can translate this back to a four-dimensional steady state vector $\pi_{\mathcal{S}(K)} = (\pi_{(K,0,0,0)}, \ldots, \pi_{(0,0,1,K-1)})$ using the inverse mapping h^{-1}. The steady state vector exists for all parameter values we have investigated.

4.2 Evaluation of Stability Measures

After obtaining the steady state vector $\pi_{\mathcal{S}(K)}$ we can evaluate $\bar{M}$ and $\bar{N}$ as follows:

$$\begin{aligned}\bar{M} &= 1 - \sum_{(L,R,U,A)\in\mathcal{S}(K),U+A=K} \pi_{(L,R,U,A)} \\ \bar{N} &= \sum_{(L,R,U,A)\in\mathcal{S}(K)} \pi_{(L,R,U,A)} \cdot (U+A)\end{aligned}$$

4.3 Model Parameters

The next issue to resolve is the estimation of the model parameters $p_I(i,j)$, p_{LU}, p_{RU}, p_{AL}, p_{AU}, p_{UL} and $p_{LR}(i)$ from some selected system parameters, namely:

- the (constant) bit error rate $p \in (0,1)$,
- the transmission rate b (bits/sec)
- the gap factor g,
- target token rotation time T_{TTRT}
- the number of stations K

under the assumptions given above (e.g. no load in the system). The probability p_{UL} corresponds to the case, where a single station experiences two successive hearback errors in its token frames. Since a token frame consists of three octets, each transmitted with 11 bits, the probability for a single token frame of being wrong is given by $1-(1-p)^{33}$ and thus we have

$$p_{UL} = (1-(1-p)^{33})^2$$

Next we determine an estimation for $p_I(j)$, where j denotes the current number of stations in state Ready. First, we assume that the slot time T_{Slot} is about 50 bit times:

$$T_{Slot} = \frac{50}{b}$$

The time T_T needed to transmit a single token frame including a following slot time or station delay can be estimated as 100 bit times, thus we have

$$T_T = \frac{100}{b}$$

A trial for including a station consists of a "Request-FDL-Status"-Frame and a corresponding answer frame, each of six octets size. Both frames must be correct, for which the probability is thus given by

$$p_{Req-FDL-Succ} = ((1-p)^{66})^2$$

If there are currently i stations in the ring ($i = A + U$), then the duration of a single token cycle is given by

$$i \cdot T_T$$

and the mean length of the gap list is given by

$$\frac{126 - i}{i}$$

So the mean time duration of a single complete scan of a stations gap list is given by

$$i \cdot T_T \quad \cdot \quad \frac{126 - i}{i} \quad = \quad (126 - i) \cdot T_T$$

Since the gap interval is given by $g \cdot TTRT$ we expect the mean fraction of time, that a single station, specifically: the current token owner, is in the course of performing a gap list scan is given by

$$\Pr[\text{Current Token-Owner performs Scan}|i \text{ Stns in Ring}] = \frac{(126 - i) \cdot T_T}{g \cdot TTRT}$$

If we now assume, that currently j stations are in state Ready, and the current token owner performs a ping on a single station address, and that station addresses are uniformly distributed, we can estimate the probability, that a station in state Ready is successfully hit by

$$p_{Req-FDL-Succ} \cdot \frac{j}{126}$$

Finally, the probability $\tilde{p_I}(i,j)$ that with i stations currently in the ring and j stations in state Ready the current token owner successfully includes a new station is given by

$$\tilde{p_I}(i,j) \quad = \quad p_{Req-FDL-Succ} \cdot \frac{j}{126} \quad \cdot \quad \frac{(126 - i) \cdot T_T}{g \cdot TTRT}$$

However, this probabilistic approximation includes the case that a single station address is pinged by different ring members in subsequent slots, which cannot happen in the protocol.

Thus the approximate value $\tilde{p_I}(i,j)$ is in fact a lower bound for $p_I(i,j)$. So it is reasonable to introduce a correction term f for estimation of $p_I(i,j)$ as follows:

$$p_I(i,j) \quad = \quad p_{Req-FDL-Succ} \cdot \frac{j+f}{126} \quad \cdot \quad \frac{(126-i)\cdot T_T}{g \cdot TTRT}$$

Next we determine an estimation jointly for p_{LU}, p_{AU} and p_{RU}, since they all occur in a scenario, where a station in a given state experiences a timeout and claims the token. A coarse estimation can be developed as follows: if we assume station addresses to be uniformly distributed over $[0,126]$ the mean station address is given by $e \approx 63$. The timeout time for a station with address n is defined to be

$$T_{TO} = T_{Slot} \cdot (6 + 2 \cdot n)$$

so the mean timeout time is then $132 \cdot T_{Slot}$ (with $n = e$). We assume that timeouts occur for each station independent from each other with fixed probability, thus the number of slots necessary for experiencing a timeout is a geometric random variable with success probability p_{LU} (or p_{AU}, p_{RU} respectively) and mean value $\frac{1-p_{LU}}{p_{LU}}$. For this mean value we expect

$$T_T \cdot \frac{1-p_{LU}}{p_{LU}} = 132 \cdot T_{Slot}$$

and now we conclude

$$p_{LU} = p_{AU} = p_{RU} = \frac{T_T}{T_T + 132 \cdot T_{Slot}}$$

For determining the probability p_{AL} we must note that the event, that an active station feels itself skipped can happen only under the following circumstances: the token frame consists of three bytes: a start delimiter (specifying the frame type), a source address and a destination address. The token frame is not equipped with a checksum, however, each byte is transmitted with start- and stopbits and a parity bit. If an error occurs in a start- or stopbit or in the start delimiter, this will be always detected. However, there is a possibility of an even number of errors occuring in one of the address bytes, which cannot be detected by the parity check. We will restrict ourselves to the case where only two bit errors occur within a token frame, higher numbers of bit errors are neglected. The probability for exactly two bit errors is given by $b(2;33,p)$. The number of placements of two errors within a token frame, such that they are not detected is given by $16 \cdot 7$, since for the first erroneous bits there are 16 possibilities, while for the second we are restricted to the remaining seven bits of the same byte. Since the overall number of placing two bit errors over 33 bits is given by $\frac{33!}{(33-2)!}$ we have finally:

$$p_{AL} = b(2;33,p) \cdot \frac{16 \cdot 7}{\frac{33!}{(33-2)!}} = \frac{7}{66} \cdot b(2;33,p)$$

For the last unknown probability, $p_{LR}(i)$ (where i is the number of stations currently in the ring) we remember, that two successive identical token cycles without transmission errors have to be detected. We can think of each token frame as a single Bernouilli experiment with success probability $\tilde{p} = p_{TokenCorrect} = (1-p)^{33}$. Then a station enters the state ready, if it encounters a run of $2 \cdot i$ subsequent successes. For the probability f_n that at the n-th Bernouilli experiment we observe the first time r subsequent successes (here with $r = 2 \cdot i$) it is known that its moment generating function is given by [3, Sec. XIII,7]

$$F(s) = \sum_{\nu=0}^{\infty} f_\nu s^\nu = \frac{\tilde{p}^r \cdot s^r}{1 - \tilde{q}s(1 + \tilde{p}s + \ldots + (\tilde{p}s)^r)}$$

where $\tilde{q} = 1 - \tilde{p}$ and the mean value given by

$$\mu(\tilde{p}, r) = F'(1) = \frac{1 - \tilde{p}^r}{\tilde{q}\tilde{p}^r}$$

Finally, we assume the number of slots necessary for moving from Listen-Token to Ready is a geometric random variable with mean $\frac{1}{p_{LR}(i)}$ such that

$$\frac{1}{p_{LR}(i)} = \mu(\tilde{p}, 2 \cdot i)$$

holds. From this we conclude

$$p_{LR}(i) = \frac{1}{\mu(\tilde{p}, 2 \cdot i)}$$

Chapter 5

Model Validation

We present simulation results on some of the stability measures defined in chapter 3, some other results regarding PROFIBUS performance over error prone links can be found in [11]. The simulations are performed in order to validate our analytical model. We have built a detailed simulation model using the CSIM simulation library [6]. This model includes the PROFIBUS link layer, the PROFIBUS MAC protocol and a shared medium with the property that all attached stations see the same signals and bits on the medium (thus with "hearback"). While all time intervals which belong to the behaviour of the medium (e.g. bit times, required idle times) are considered within the model, we assume that protocol processing time within the stations is negligible, with the exception of the station delay between a request telegram and the corresponding immediate ack. The validation of the simulator was done by inspection. During all simulations there are $K = 10$ stations and no load in the system, thus there occur only token frames and Request-FDL-Status Frames. All simulations are run for 3600 seconds. Since $N(t)$ is sampled every 100 μsec the confidence intervals are very tight and thus not shown. The gap factor was chosen to be $g = 6$, the slot time was chosen to be 400 μsec, the station delay is 100 μsec, the target token rotation time is $TTRT = 20$ msec and the bitrate is $b = 500$ kBit/sec. Clearly, we have also evaluated the analytical model with these parameters, using $f = 2$ as correction factor. In figure 5.2 we show $\bar{N}(3600)$ as obtained from our simulations, and $\bar{N}$, as calculated by the analytical model. Similarly, in figure 5.1 we show $\bar{M}(3600)$ from the simulations and $\bar{M}$ from the analytical model. Our assumption is, that the simulations reach the steady state very fast as compared to the simulation durations of 3600 seconds. In both cases we have varied the bit error rate using an independent bit error model, i.e. bit errors occur independent from each other, each with a fixed rate. It can be seen that the model captures the simulated system

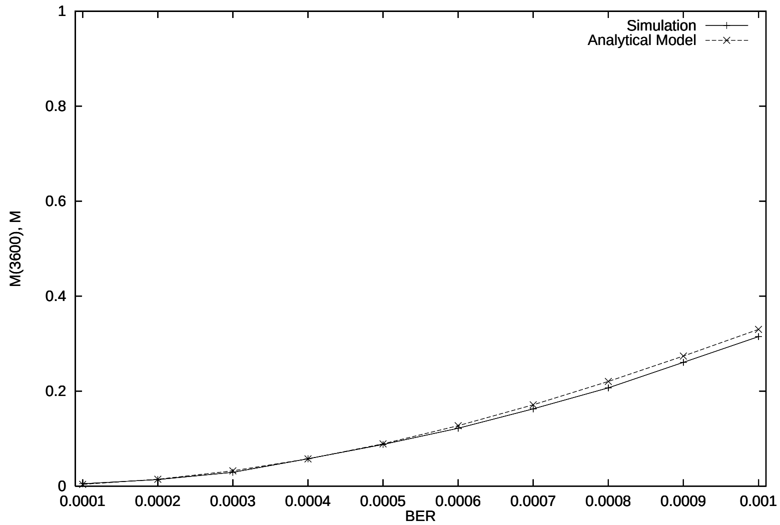

Figure 5.1: M(3600) and M vs. BER (Independent Errors)

behaviour quite good, the differences between them are small.

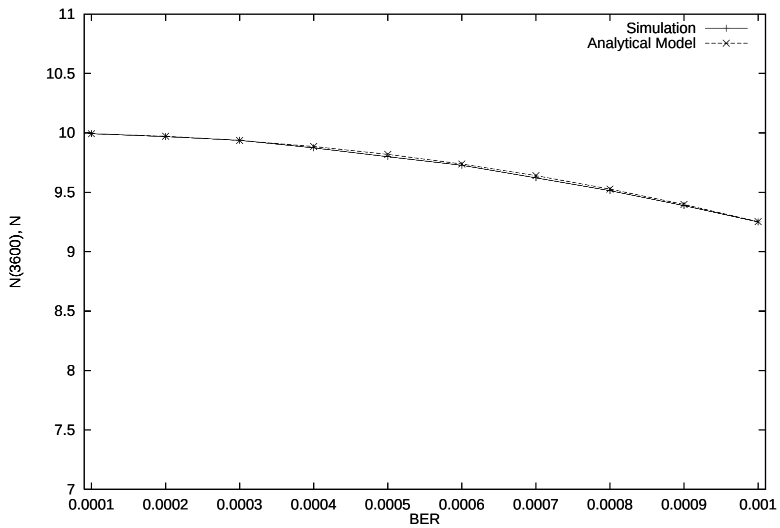

Figure 5.2: N(3600) and N vs. BER (Independent Errors)

Chapter 6

Conclusions

This paper makes two main contributions:

- It analyzes the notion of ring stability and identifies two main causes for ring instability.
- An analytical model for PROFIBUS ring membership was developed, using a discrete time markov chain approach. This model takes the identified causes of instability into account.

The analytical model relies on a set of model parameters (e.g. the transition probabilities p_{LU}, p_{UL}), which can in turn be estimated from some static parameters of a PROFIBUS system, e.g. the number of stations K, the bitrate b or the bit error probability p, and a single correction factor. The correction factor can be chosen such that the model gives quite accurate results for two main stability measures, as compared to simulations.

Bibliography

[1] DIN - Deutsches Institut für Normung, Beuth Verlag Berlin. *DIN 19245 Teil 1 - PROFIBUS: Übertragungstechnik, Buszugriffs- und Übertragungsprotokoll, Dienstschnittstelle zur Anwendungsschicht, Management*, April 1991.

[2] DIN - Deutsches Institut für Normung, Beuth Verlag Berlin. *DIN 19245 Teil 2 - PROFIBUS: Kommunikationsmodell, Dienste für die Anwendung, Protokoll, Syntax, Codierung, Schnittstelle zur Schicht 2, Management*, April 1991.

[3] William Feller. *An Introduction to Probability Theory and Its Applications - Volume I.* John Wiley, New York, third edition, 1968.

[4] German Institute of Standardization (DIN). *PROFIBUS Standard Part 1 and 2*, 1991.

[5] Hong ju Moon, Hong Seong Park, Sang Chul Ahn, and Wook Hyun Kwon. Performance Degradation of the IEEE 802.4 Token Bus Network in a Noisy Environment. *Computer Communications*, 21:547–557, 1998.

[6] Mesquite Software, Inc., T. Braker Lane, Austin, Texas. *CSIM18 Simulation Engine – Users Guide*, 1997.

[7] PROFIBUS Nutzerorganisation e.V., PROFIBUS Nutzerorganisation e.V., Haid-und-Neu-Str. 7. *PROFIBUS – Entwurf Technische Richtlinie – Implementierungshinweise zur DIN 19245 Teil 2*, December 1993.

[8] PROFIBUS Nutzerorganisation e.V., PROFIBUS Nutzerorganisation e.V., Haid-und-Neu-Str. 7. *Implementation Guide to DIN 19245 Part 1*, August 1994.

[9] PROFIBUS Nutzerorganisation e.V., PROFIBUS Nutzerorganisation e.V., Haid-und-Neu-Str. 7. *PROFIBUS – Entwurf Technische Richtlinie – Implementierungshinweise zur DIN 19245 Teil 1*, August 1994.

[10] Union Technique de l'Electricité. *General Purpose Field Communication System, EN 50170, Volume 2: PROFIBUS*, 1997.

[11] Andreas Willig. Analysis and tuning of the profibus token passing protocol for use over error-prone links. TKN Technical Report Series TK-N99-001, Telecommunication Networks Group, Technical University Berlin, March 1999.