CAN message response time
Embedded systems engineering

David Kendall
Northumbria University
\[ T_{\text{delay}} = T_{\text{pre}} + T_{\text{wait}} + T_{\text{tx}} + T_{\text{post}} \]

- We have seen that in general the end-to-end delay of an interaction between two nodes in a distributed system can be decomposed as above.
- Here we focus on the message response time \((T_{\text{pre}} + T_{\text{wait}} + T_{\text{tx}})\) for CAN.
- This problem was first considered by Tindell and Burns in 1994. Their analysis has been widely used.
- Their original analysis was revised in 2007 by Davis et al.
At the start of every arbitration, each node enters the highest priority message in its queue.

Message \( m \) is queued by a software task taking a time between 0 and \( J_m \) to queue the message (queuing jitter).

Event that causes queuing of message occurs with minimum inter-arrival time \( T_m \) (message period).

Each message has a deadline \( D_m \) – maximum allowed time from occurrence of initiating event to end of successful transmission of message.

\( R_m \) is the worst-case response time of the message – longest time from occurrence of initiating event to reception of message.

Message \( m \) is schedulable if \( R_m \leq D_m \). System is schedulable if all messages are schedulable.
CAN message response time (Tindell and Burns 1994)

\[ R_m = J_m + w_m + C_m \]

- \( J_m \) – queuing jitter
- \( w_m \) – worst-case queuing delay (\( T_{wait} \))
- \( C_m \) – worst-case message transmission time
Message transmission time

\[
C_m = \left( g + 8s_m + 13 + \left\lfloor \frac{g + 8s_m - 1}{4} \right\rfloor \right) \tau_{\text{bit}}
\]

- \( C_m \) – transmission time
- \( g = 34 \) for standard CAN frame
- \( s_m \) – number of data bytes in message
- \( \tau_{\text{bit}} \) – transmission time for 1 bit
- For 11-bit identifiers, formula simplifies to

\[
C_m = (55 + 10s_m) \tau_{\text{bit}}
\]
Queuing delay $w_m$ consists of

- **Blocking time** $B_m$ – maximum time that message might be delayed waiting for completion of transmission of a lower priority message
- **Interference** – maximum time that message might be delayed by higher priority messages (that win arbitration)

Maximum blocking occurs when a message $m$ is queued just as the longest lower priority message begins transmission

$$B_m = \max_{k \in lp(m)} (C_k)$$

- where $lp(m)$ is the set of messages with lower priority than $m$
Tindell and Burns 1994 gives

\[ w_m = B_m + \sum_{\forall k \in hp(m)} \left\lfloor \frac{w_m + J_k + \tau_{bit}}{T_k} \right\rfloor C_k \]

This equation can be solved using the recurrence relation

\[ w_{m+1}^n = B_m + \sum_{\forall k \in hp(m)} \left\lfloor \frac{w_m^n + J_k + \tau_{bit}}{T_k} \right\rfloor C_k \]

where a suitable starting value is \( w_0^m = B_m \)

Iterate until either

- \( J_m + w_{m+1}^n + C_m > D_m \): m not schedulable, or
- \( w_{m+1}^n = w_m^n \): worst-case response is \( J_m + w_{m+1}^n + C_m \)
A problem with the analysis

Davis et al. 2007 showed that the previous analysis is flawed.

Consider this simple example

<table>
<thead>
<tr>
<th>Message</th>
<th>Priority</th>
<th>Period</th>
<th>Deadline</th>
<th>TX time</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>2.5 ms</td>
<td>2.5 ms</td>
<td>1 ms</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
<td>3.5 ms</td>
<td>3.25 ms</td>
<td>1 ms</td>
</tr>
<tr>
<td>C</td>
<td>3</td>
<td>3.5 ms</td>
<td>3.25 ms</td>
<td>1 ms</td>
</tr>
</tbody>
</table>

and its analysis for worst-case response of message C

Analysis gives $R_C = 3$ but example shows $R_C = 3.5$ !!!
Sufficient but not necessary test

- Davis et al. 2007 give a detailed account of the reasons for the flaw in the original analysis.
- They provide a revised, exact analysis that is not flawed.
- The simplest revision that they propose is to assume that the blocking time for each message is given by the longest possible transmission time of a message on the bus, leading to the formula below:

\[
 w_{m}^{n+1} = B^{\text{MAX}} + \sum_{\forall k \in hp(m)} \left[ \frac{w_{m}^{n} + J_{k} + \tau_{\text{bit}}}{T_{k}} \right] C_{k}
\]

- where \( B^{\text{MAX}} \) is the longest possible transmission time of a message on the bus.
Acknowledgements