Analysis of a Biphase Mark Protocol with Uppaal and PVS

Frits Vaandrager and Adriaan de Groot
Nijmegen Institute for Computing and Information Sciences

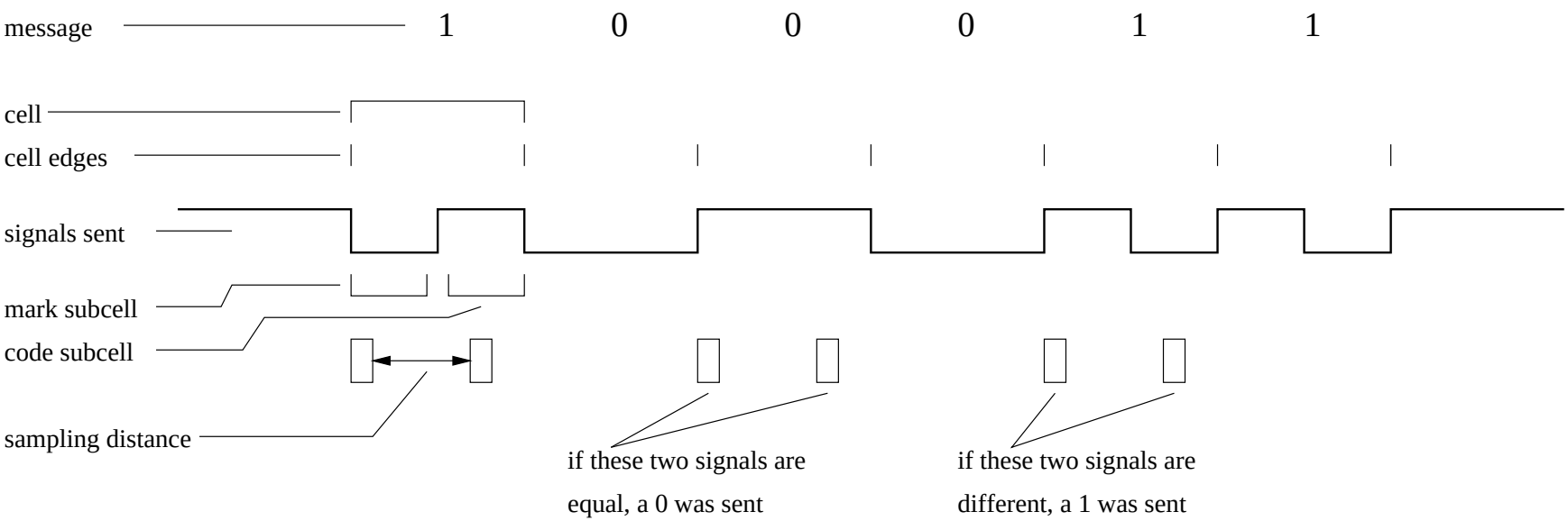
Biphase Mark Protocol

Convention for representing both a string of bits and clock edges in a square wave.

Used, for instance, in:

1. Intel 82530 Serial Communications Controller
2. Ethernet
3. Optical communications
4. Satellite telemetry applications
5. ...

Biphase Mark Protocol (cnt)



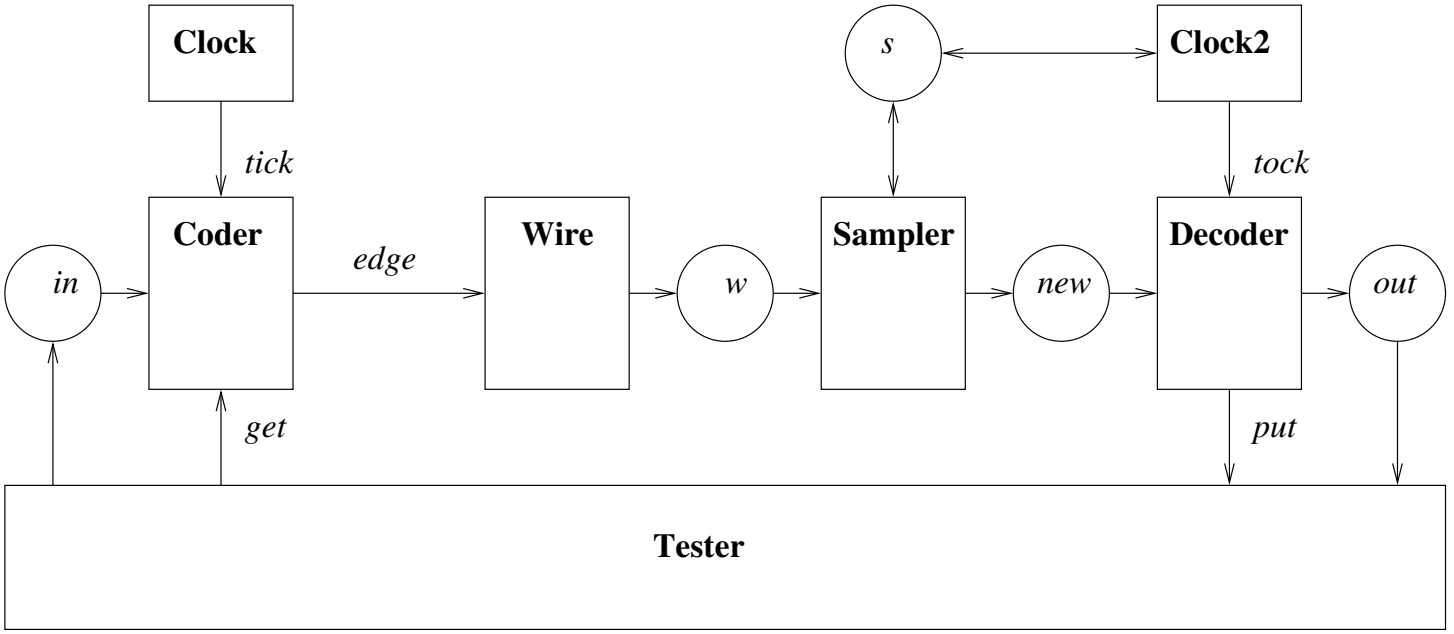
Challenges

1. During some time after the sender generates an edge, reading may produce any value.

2. Receiver samples wire nondeterministically at some point during each clock cyle.

3. Clock drift and jitter.

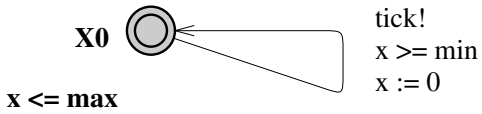
Overview of Uppaal Model



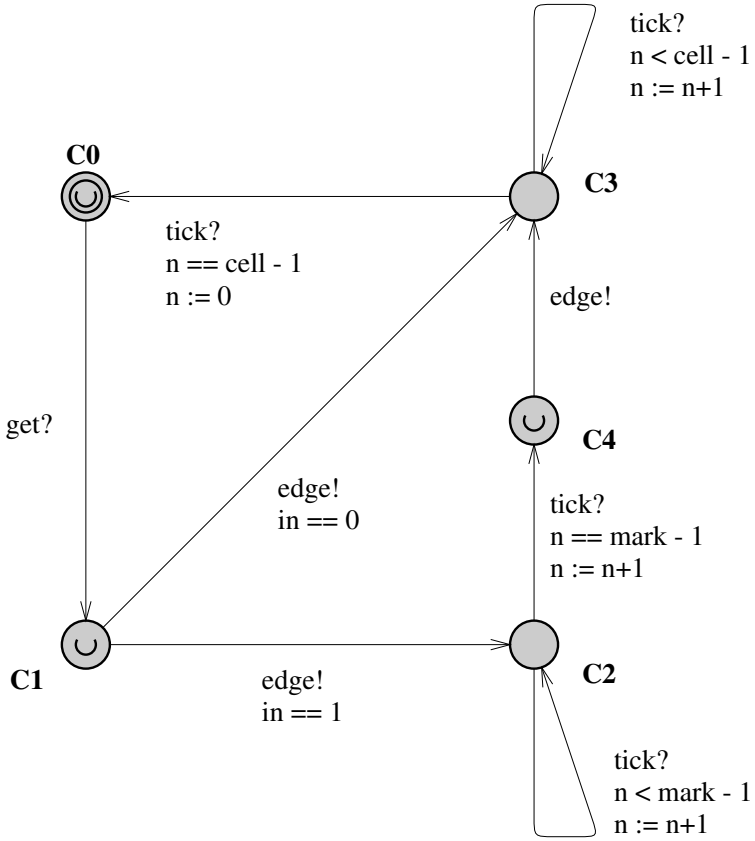
Variables and Constants in Uppaal Model (instance)

```
chan get, put, edge, tick, tock;
int m, n;
int[0,1] in, out, v, w, new, old, buf;
clock x, y, z;
const cell 32;
const mark 16;
const sample 23;
const min 81;
const max 100;
const edgelength 81;
```

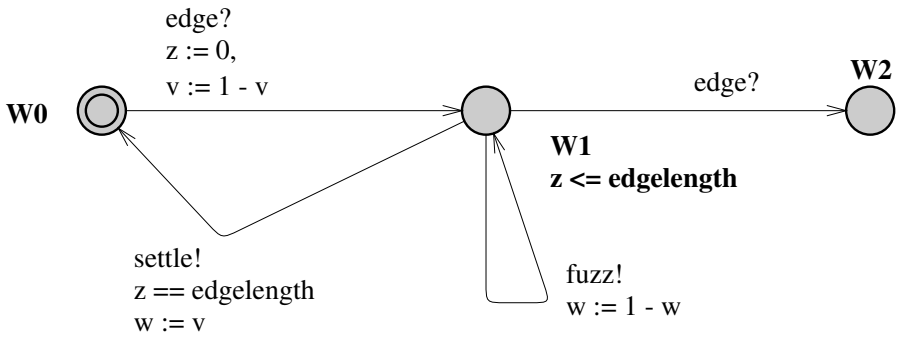
Clock



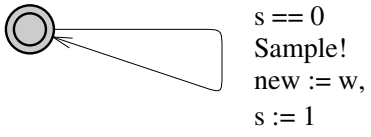
Coder


tick?
n < cell - 1
n := n+1
C0
C3
tick?
n == cell - 1
n := 0
edge!
get?
C4
edge!
in == 0
tick?
n == mark - 1
n := n+1
C1
C2
edge!
in == 1
tick?
n < mark - 1
n := n+1

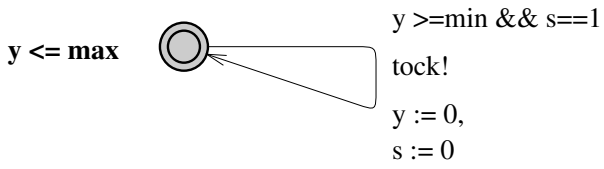
Wire


edge?
z := 0,
v := 1 - v
W0
edge?
W2
W1
z <= edgelength
settle!
z == edgelength
w := v
fuzz!
w := 1 - w

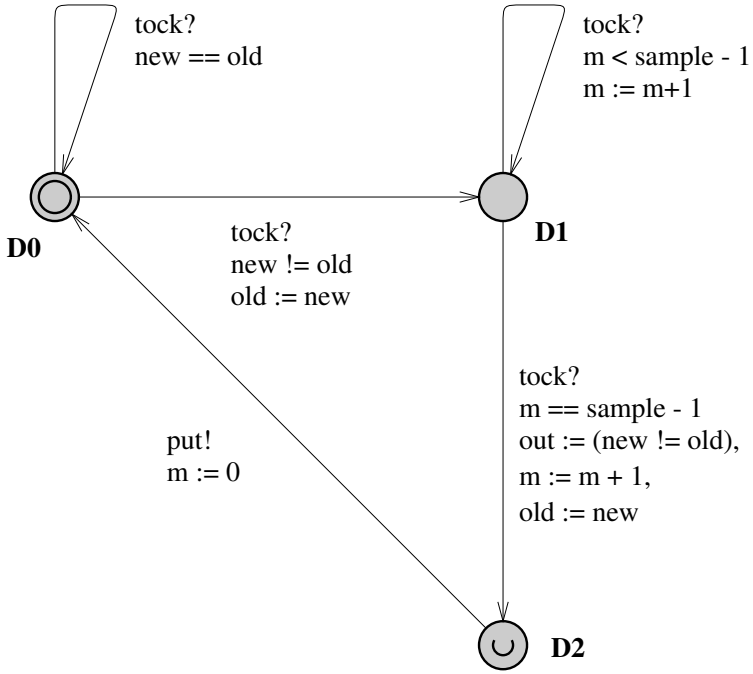
Sampler



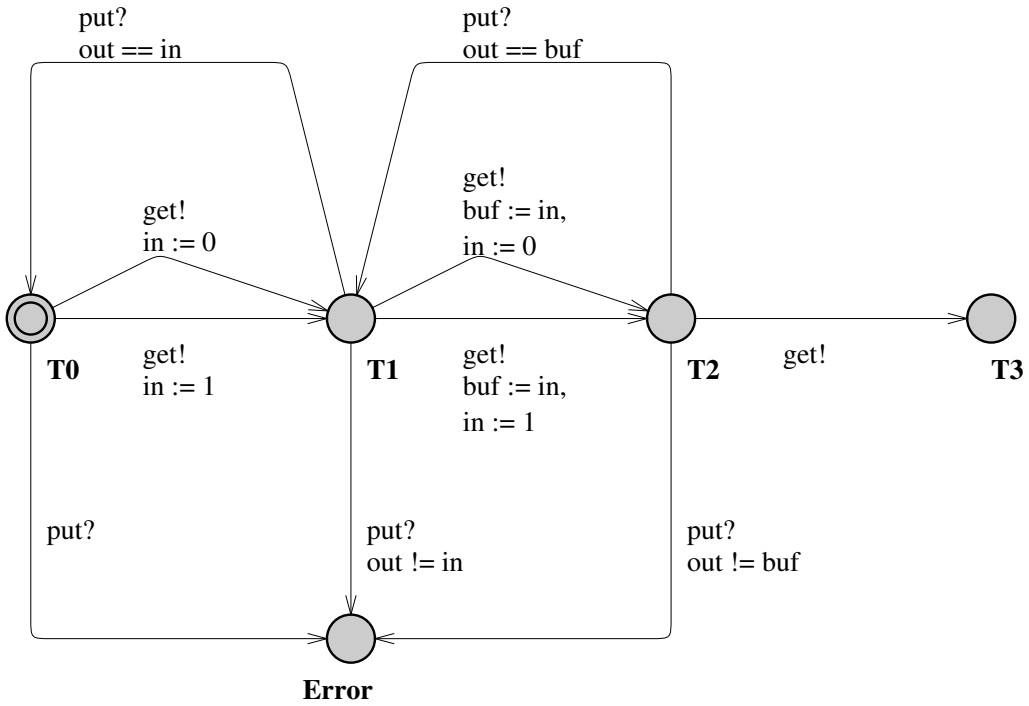
Decoder Clock



Decoder


tock?
new == old
tock?
m < sample - 1
m := m+1
D0
D1
tock?
new != old
old := new
tock?
m == sample - 1
out := (new != old),
m := m + 1,
old := new
put!
m := 0
D2

Tester


put?
out == in
put?
out == buf
get!
in := 0
get!
buf := in,
in := 0
T0
get!
in := 1
T1
get!
buf := in,
in := 1
T2
get!
T3
put?
put?
out != in
put?
out != buf
Error

Requirements for Correctness

Receiver detects edge at begin cell

$$\text{mark} \cdot \text{min} > 2 \cdot \text{max} + \text{edgelength}$$

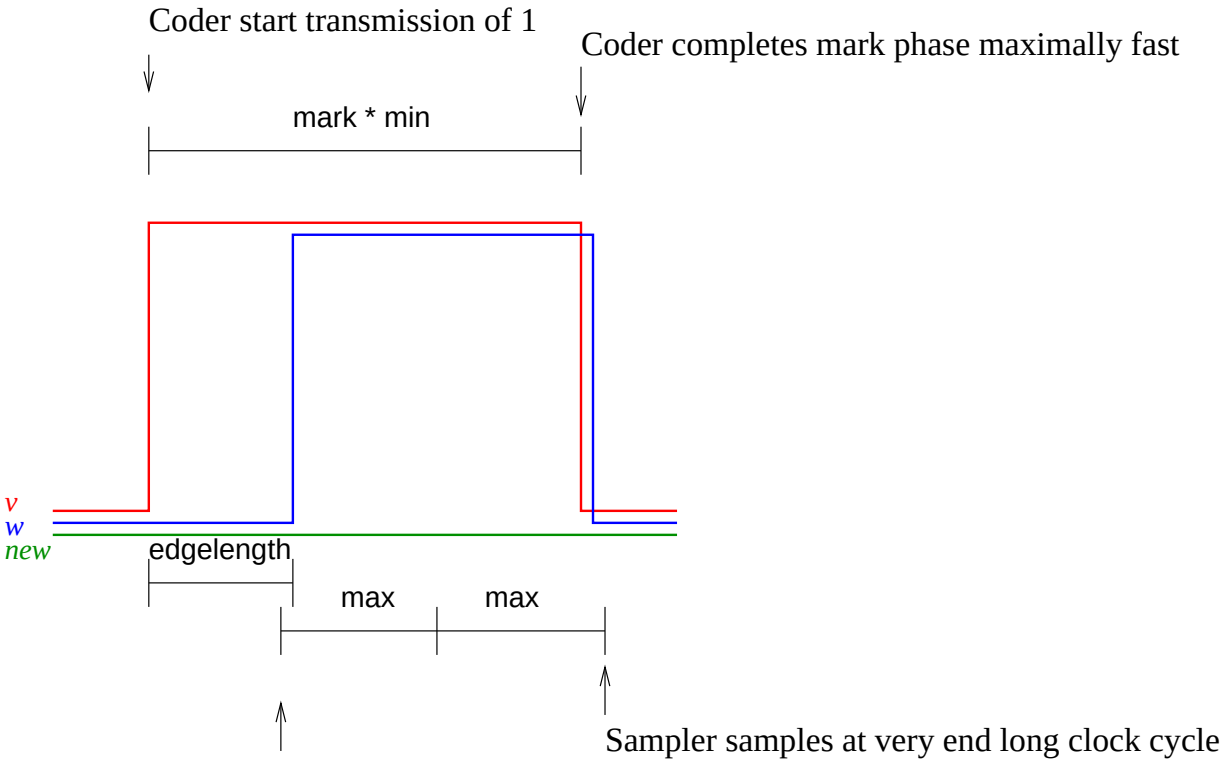
Receiver does not sample too early

$$(\text{sample} - 1) \cdot \text{min} > \text{mark} \cdot \text{max} + \text{edgelength}$$

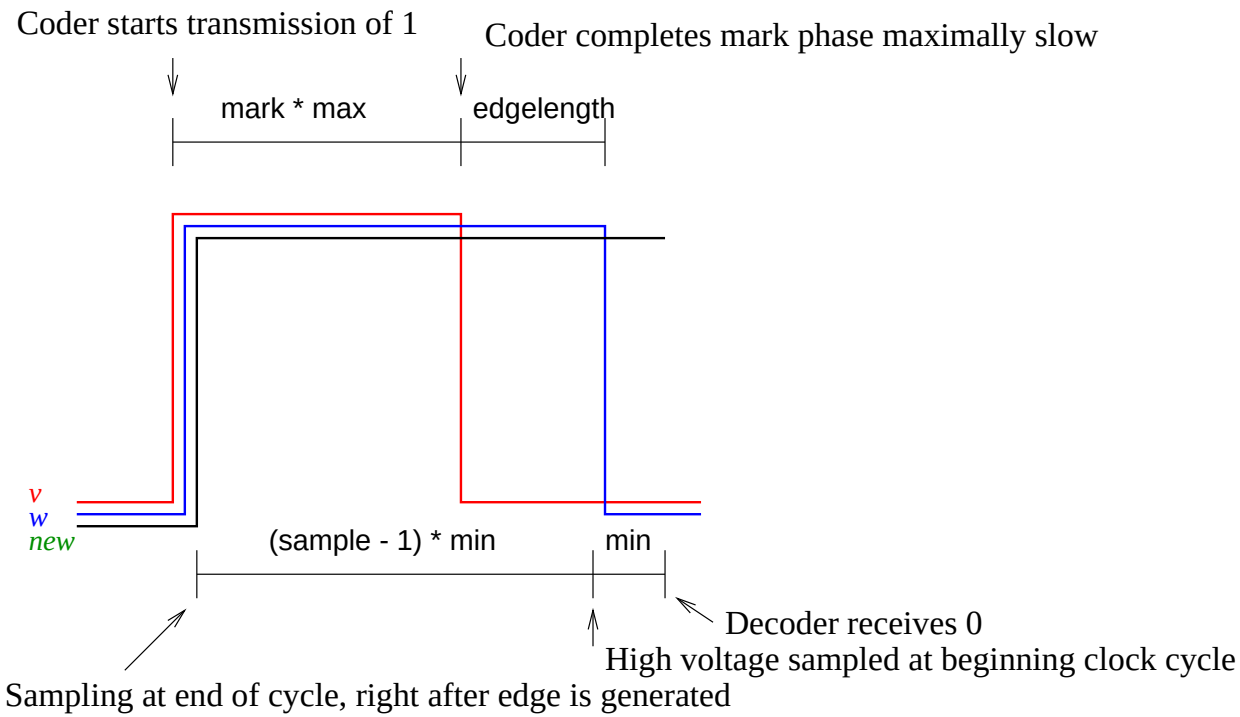
Receiver does not sample too late

$$\text{cell} \cdot \text{min} > (\text{sample} + 2) \cdot \text{max} + \text{edgelength}$$

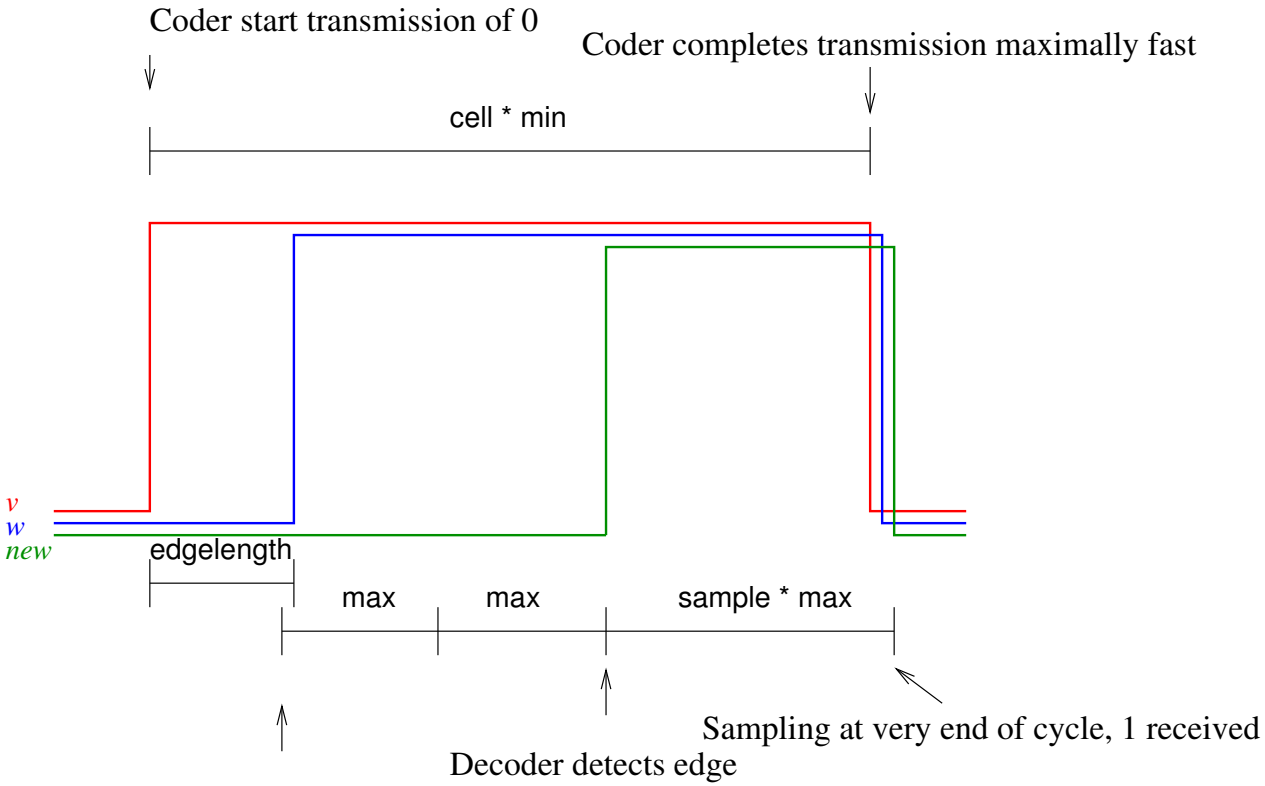
Receiver misses edge at begin cell



Receiver samples too early



Receiver samples too late



Main result

The Error state cannot be reached if and only if the three stated inequalities hold for the parameters.

Proof

Manual proof, formalized with PVS. Several instances of the 3 counterexamples and 36 auxiliary invariants (including 15 trivial ones) have been found resp. checked using Uppaal.

Example of invariant that Uppaal cannot handle in general:

$$C2 \vee (C3 \wedge in = 0) \quad \Rightarrow \quad n \cdot \mathsf{min} \leq z - x \leq n \cdot \mathsf{max}$$

Relative Time

We assume

$$0 < \text{min} \leq \text{max}$$

and define

$$\rho = \frac{\text{min}}{\text{max}}$$

$$E = \frac{\text{edgelength}}{\text{max}}$$

Requirements for Correctness (rephrased)

Receiver detects edge at begin cell

$$\mathsf{mark} \cdot \rho \quad > \quad 2 + E$$

Receiver does not sample too early

$$(\mathsf{sample} - 1) \cdot \rho \quad > \quad \mathsf{mark} + E$$

Receiver does not sample too late

$$\mathsf{cell} \cdot \rho \quad > \quad \mathsf{sample} + 2 + E$$

Maximal Tolerance on Timing

$$\rho \quad > \quad \max\left(\frac{2+E}{\mathsf{mark}}, \ \frac{\mathsf{mark}+E}{\mathsf{sample}-1}, \ \frac{\mathsf{sample}+2+E}{\mathsf{cell}}\right)$$

Example Configurations with $E = 1$

cell	16	32	18
mark	8	16	5
sample	11	23	10
ρ	0.91	0.82	0.73

Physical Clocks

Typical clocks used in hardware are incorrect by less than 15.10^{-6} seconds per second.

Thus, in practice,

$$\rho \geq \frac{1 - 15.10^{-6}}{1 + 15.10^{-6}} \approx 0.99997$$

Minimizing Cell Size

Assume $\rho = 1$ and $E = 1$. Then we derive

$$\begin{aligned} \mathsf{mark} &> 3 \\ \mathsf{sample} &> \mathsf{mark} + 2 \\ \mathsf{cell} &> \mathsf{sample} + 3 \end{aligned}$$

Hence, values of parameters are at least

$$\mathsf{mark} = 4 \quad \mathsf{sample} = 7 \quad \mathsf{cell} = 11$$

If we require $\mathsf{cell} = 2 \cdot \mathsf{mark}$ then minimal values are

$$\mathsf{mark} = 7 \quad \mathsf{sample} = 10 \quad \mathsf{cell} = 14$$

Related Work

Moore ('94)

Verification of few instances with Boyer-Moore theorem prover.
Derived timing bounds not optimal. No clock jitter, $E = 1$.

Ivanov & Griffioen ('98)

Automatic verification of few instances with HyTech.
Polling only at the end of a read cycle.

Van Hung ('96, '98)

Full parameter analysis with PVS + Duration Calculus.
Debatable modelling assumptions. No clock jitter.

Bensalem et al ('00) & Henzinger et al ('01)

Partial success in proving parameter constraints automatically.

Conclusions (cf Moore)

1. We offer our model primarily as a catalyst for thought.
   Model says certain instances will work. Will they?

2. We ignore various engineering realities: metastability, reflection, noise, and distortion, etc.

3. Uppaal very helpful in model construction, and for gaining insight.
   Model checking essential for analysis of additional features, such as termination and bus collisions.

4. PVS essential for handling parameter constraints in full generality.