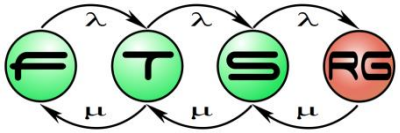
Valósidejű biztonságkritkus protokoll modellellenőrzése

Tóth Tamás



Budapesti Műszaki és Gazdaságtudományi Egyetem
Méréstechnika és Információs Rendszerek Tanszék

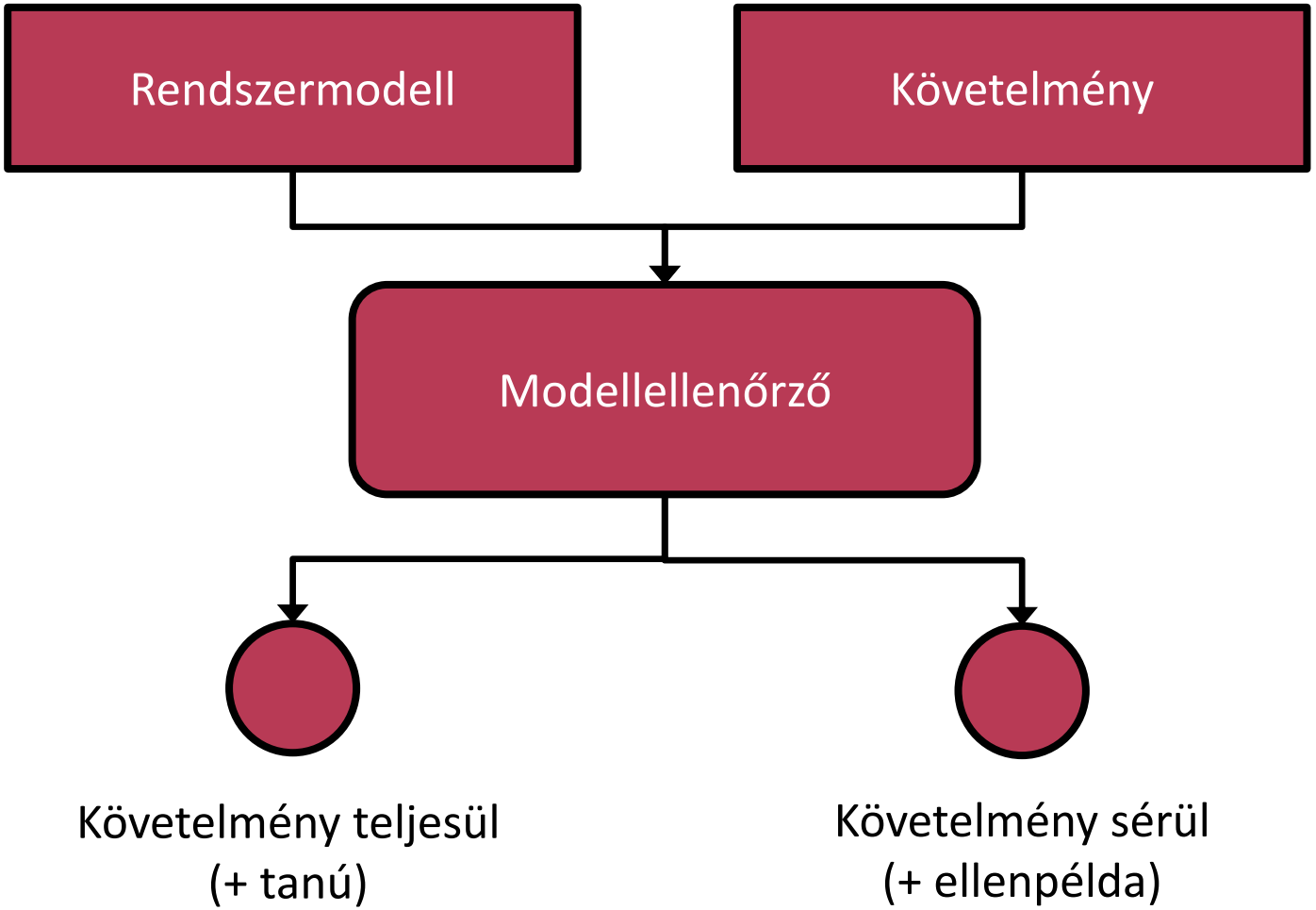
Biztonságkritikus rendszerek

- Egy rendszerszintű hibajelenség
  - emberéletet veszélyeztethet
  - súlyos természeti vagy anyagi kárt okozhat
- Példa:
  - vasúti biztosítórendszerek
- Jellemzők:
  - időzített viselkedés
  - parametrikus viselkedés
- Fontos az igazoltan helyes működés
  - formális módszerek használata

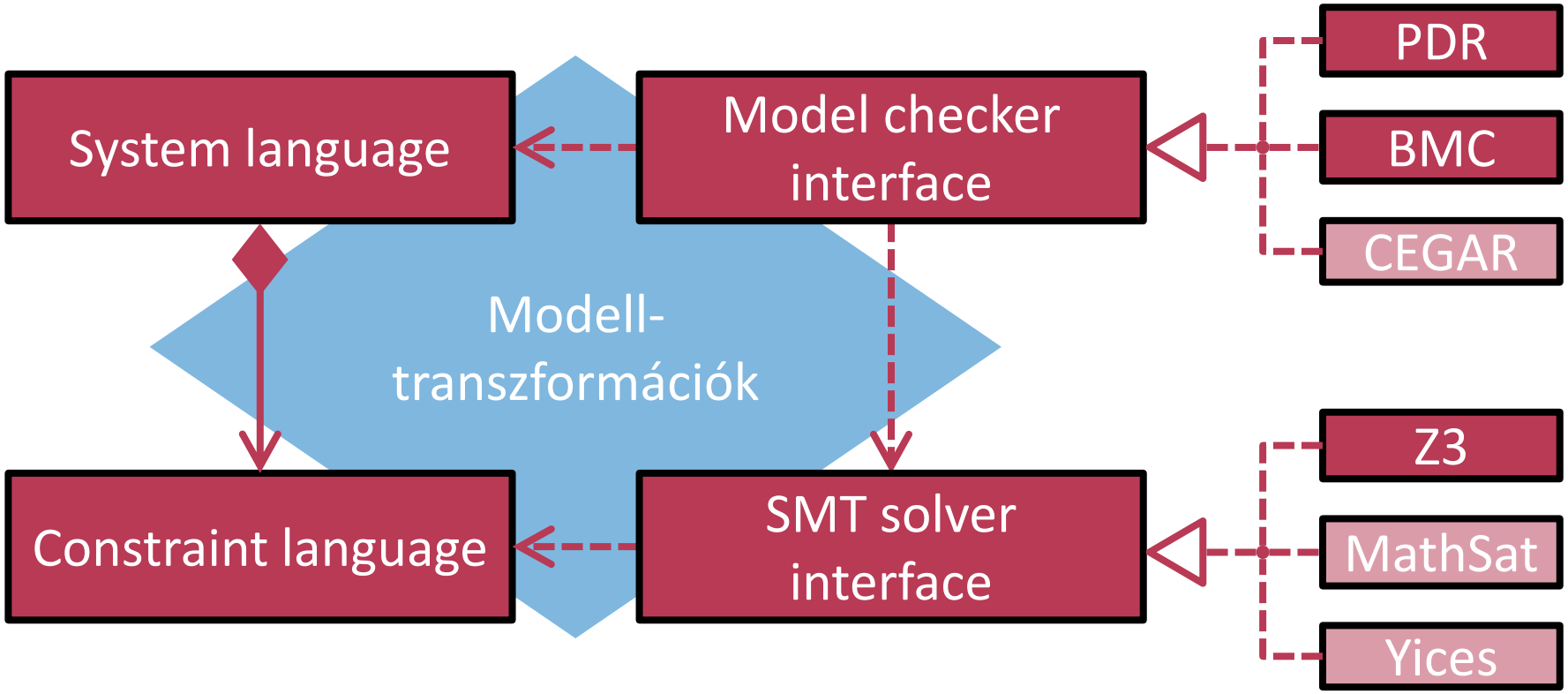
Formális módszerek

- Szoftver-hardver rendszerek különböző fejlesztési fázisaiban alkalmazható matematika módszerek
  - Specifikáció
  - Verifikáció
  - Tervezés
  - …
- Formális verifikáció kihívásai
  - Követelmény modellezése
  - Rendszer modellezése
  - Ellenőrzés

Modellellenőrzés



TTMC keretrendszer



Példa: kényszerleíró nyelv

```
specification FiniteAbelianGroup(n : natural) {
  type G : [ 1 to n ]

  const e : G := 1
  function op(a : G, b : G) : G
  function inv(a : G) : G

  constraint forall (a : G, b : G, c : G) :
    op(a, op(b, c)) = op(op(a, b), c)
  constraint forall (a : G, b : G) :
    op(a, b) = op(b, a)
  constraint forall (a : G) : op(a, e) = a
  constraint forall (a : G) : op(a, inv(a)) = e
}
```

Példa: specifikációs nyelv

```
system Fischer(this : Id) := {
  global var lock : [0 to maxId]
  local var location : Location
  local var c : clock
  invariant location = ::waiting imply c <= a

  initialization let location = ::sleeping;

  transition location = ::waiting --> {
    let location’ = ::trying
    let lock’ = this;
    let c’ = 0.0;
  }
  ...
}
```

Példa: specifikációs language

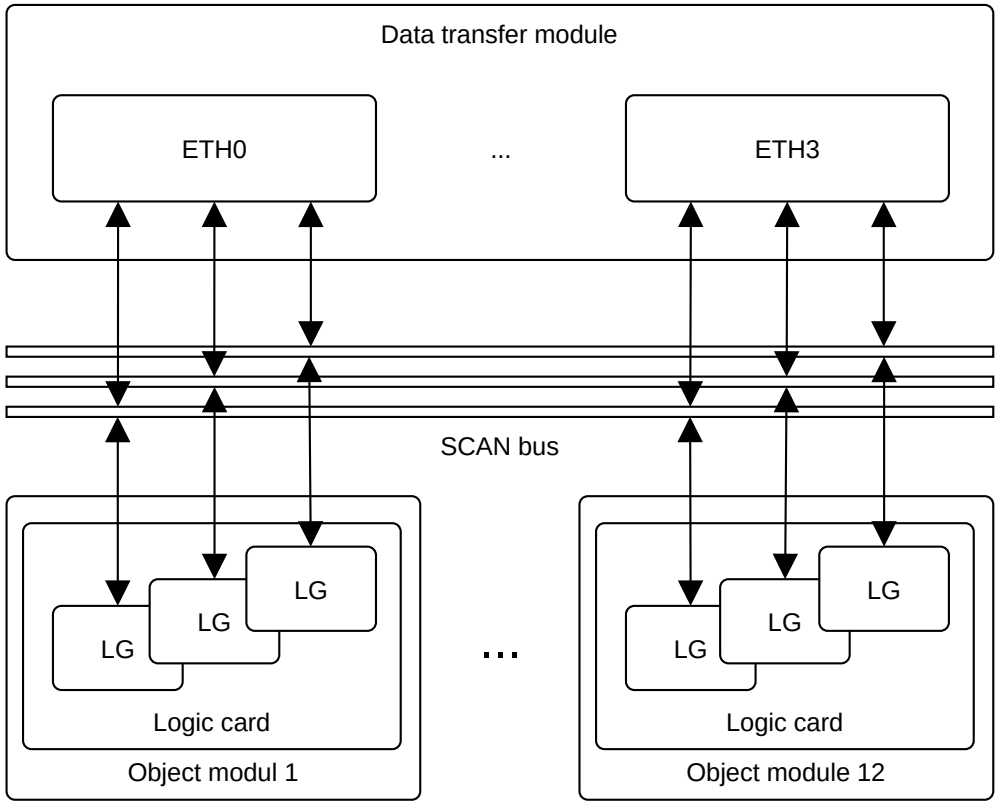
```
system System :=
  async (id : Id) : Fischer(id)


property safe : System models
  A G forall (id : Id) : (
    location[id] = ::critical imply lock = id
  )
```

ProSigma SCAN protokoll

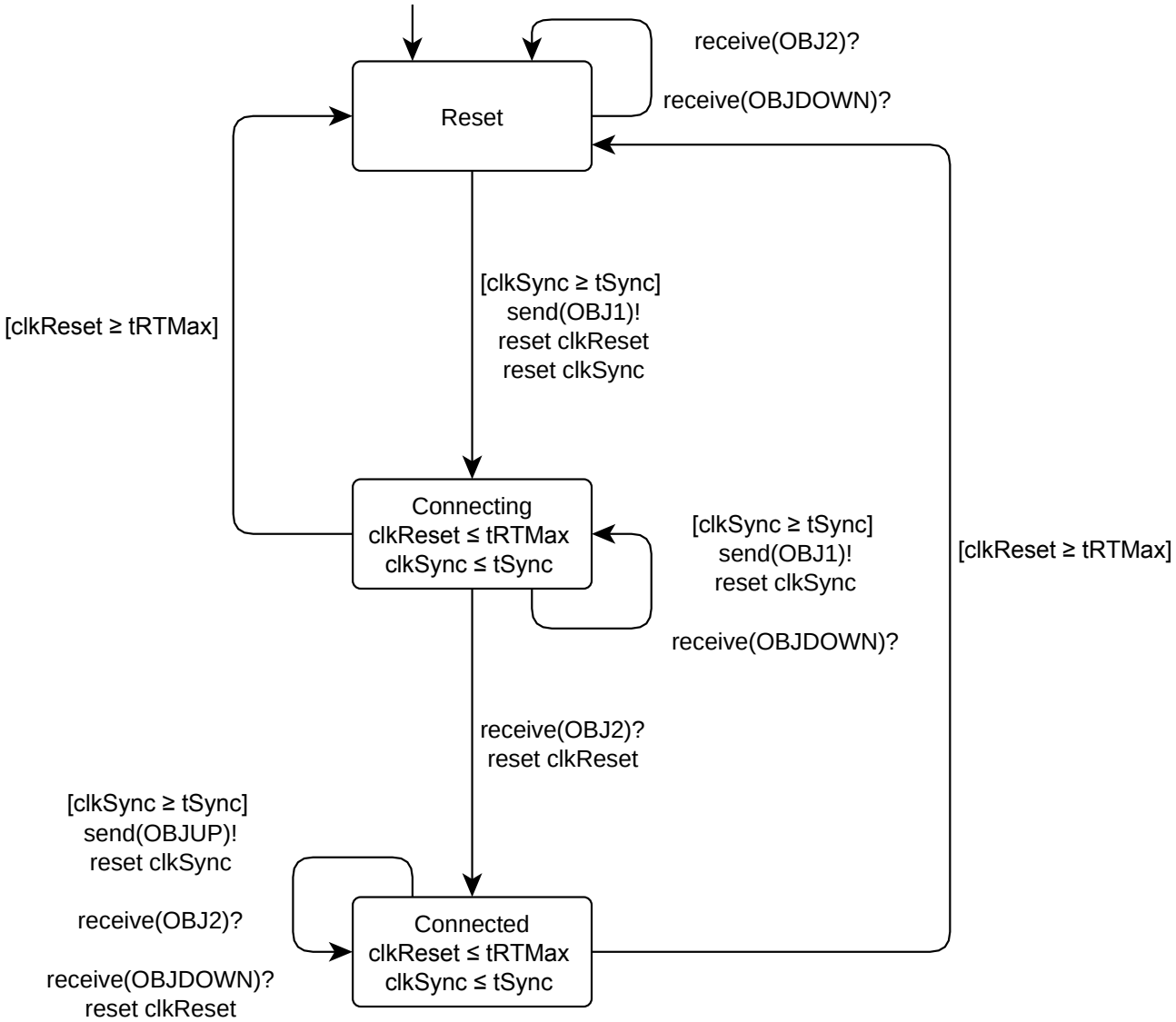
- A Prolan Irányítástechnikai Zrt. fejlesztése
- Feladata: tévesztésmentes jelátvitel biztosítása

- Elosztott
- Valósidejű
- Biztonságkritikus (SIL 4)

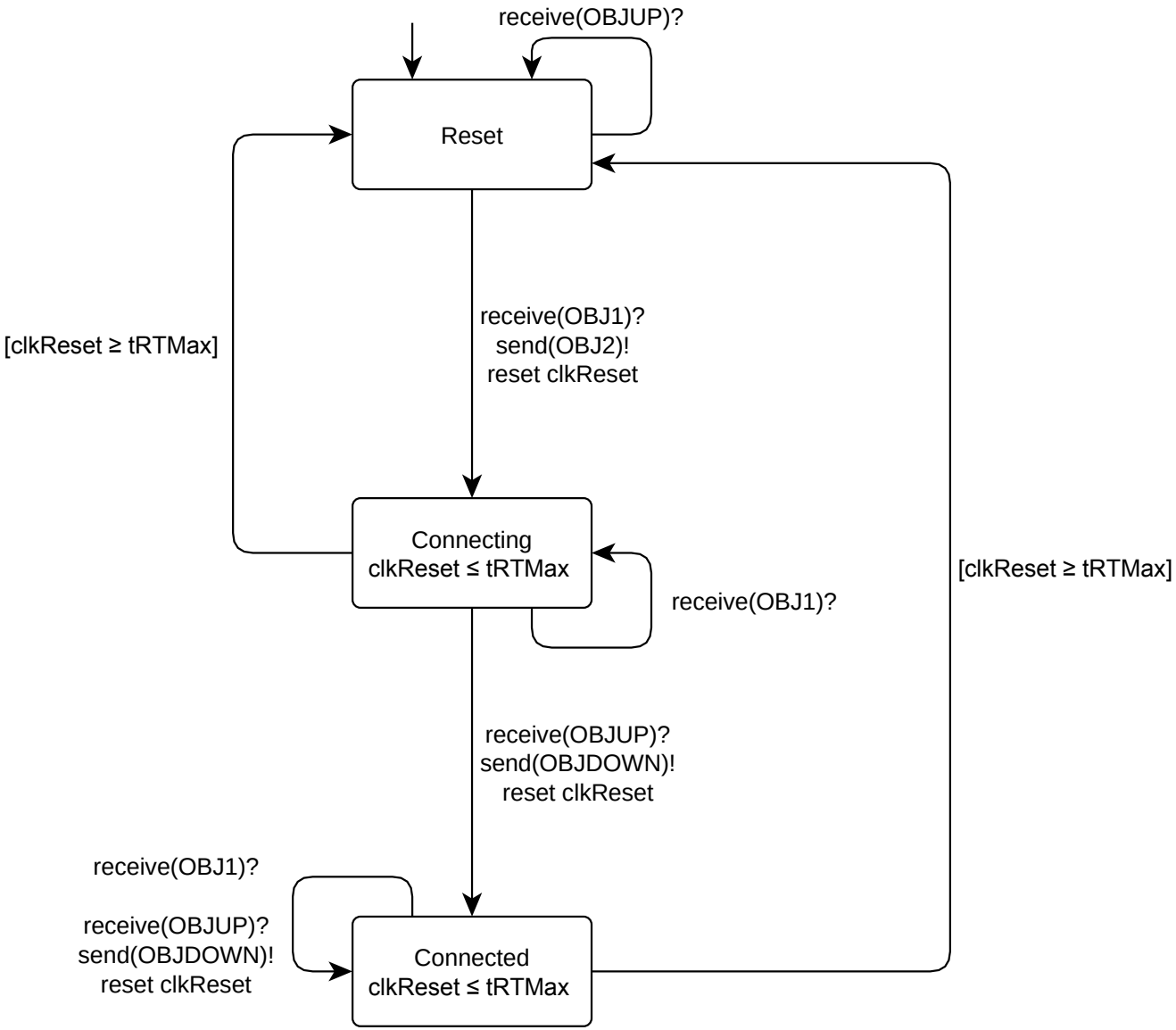
- Modellezett funkció: kapcsolatkezelés
  - Kapcsolatfelvétel
  - Objektumállapot-átvitel



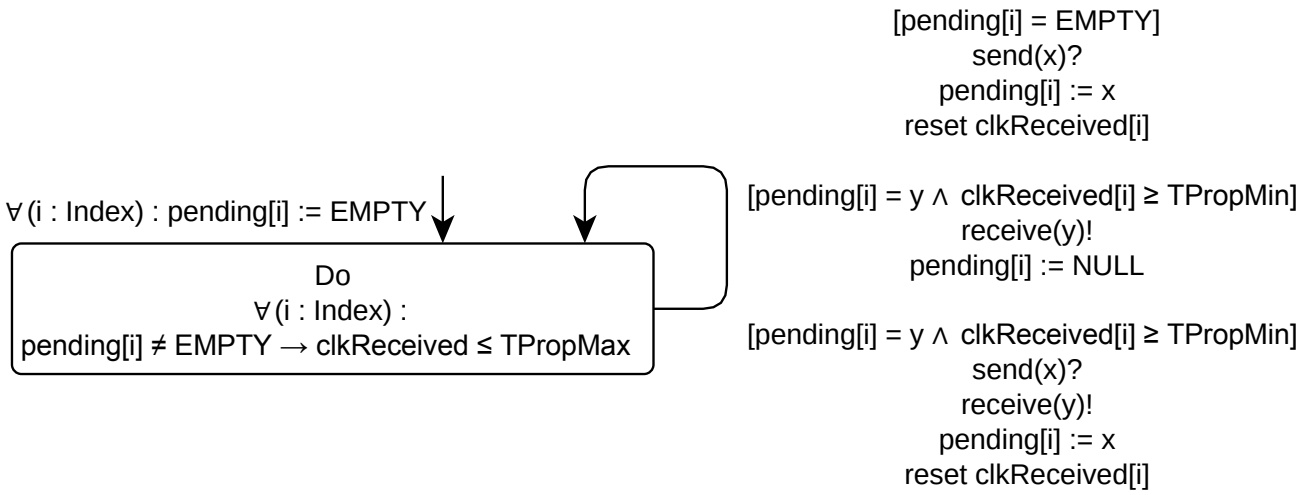
Terepi LG modul

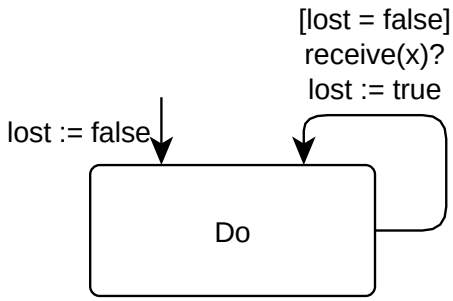


Irányítási LG modul



Kiegészítés: csatorna- és hibamodell





Modulok modellezése

```
transition
  control_location = ::reset and
  receive’ = ::obj1 --> {
    let control_location’ = ::connecting;
    let send’ = ::obj2;
    let control_reset’= 0.0;
  }
```

Csatorna modellezése

```
transition async (i : Index) :
  send’ /= ::null and
  pending[i] /= ::null and
  clkReceived[i] >= tPropMin --> {
    let pending’[i] = send’;
    let receive’ = pending[i];
    let clkReceived’[i] = 0.0;
  }
```

Szinkronizáció

```
transition
  control_location = ::reset and
  receive’ = ::obj1 --> {
    let control_location’ = ::connecting;
    let send’ = ::obj2;
    let control_reset’= 0.0;
  }

transition async (i : Index) :
  send’ /= ::null and
  pending[i] /= ::null and
  clkReceived[i] >= tPropMin --> {
    let pending’[i] = send’;
    let receive’ = pending[i];
    let clkReceived’[i] = 0.0;
  }
```

Szinkronizáció

```
transition async (i : Index) :
  control_location = ::reset and
  clkReceived[i] >= tPropMin and
  pending[i] = ::obj1 --> {
    let control_location’ = ::connecting;
    let send’ = ::obj2;
    let pending’[i] = ::obj2;
    let receive’ = ::obj1;
    let control_reset’= 0.0;
    let clkReceived’[i] = 0.0;
  }
```

Az ellenőrzés eredménye

- Egy meglepő ellenpélda a várt működésre
  - Egy üzenet elvesztése a protokoll leragadását okozza
- Javaslat a specifikáció javítására
- A módosított rendszer helyességének igazolása