

5. Szkeleton tervezése

54 – *Override*

Konzulens:

dr. László Zoltán

Csapattagok:

Kriván Bálint	CBVOEN	balint@krivan.hu
Jákli Gábor	ONZ5G1	j_gab666@hotmail.com
Dévényi Attila	L1YRH0	devenyat@gmail.com
Apagyi Gábor	X8SG3T	apagyi.gabooo@gmail.com
Péter Tamás Pál	N5ZLEG	falconsaglevlist@gmail.com

2011. március 14.

Tartalomjegyzék

5 Szkeleton tervezése	4
5.1. Errata	4
5.1.1. Objektumleírás: Wire	4
5.1.2. Objektumleírás: Node	4
5.1.3. Osztályleírás: AbstractComponent	4
5.1.4. Osztályleírás: Node	5
5.1.5. Osztályleírás: Wire	5
5.1.6. Statikus struktúra diagramok	6
5.2. A szkeleton modell valóságos use-case-ei	7
5.2.1. Use-case diagram	7
5.2.2. Use-case leírások	7
5.3. Architektúra	13
5.4. A szkeleton kezelői felületének terve, dialógusok	13
5.4.1. Program üzeneteinek formátuma	13
5.5. Szekvencia diagramok a belső működésre	16
5.6. Napló	25

Ábrák jegyzéke

5.1. Statikus struktúra nézet	6
5.2. A szkeleton modell valóságos use-case-ei	7
5.3. Áramkör inicializálása	16
5.4. Kapcsoló és Led	17
5.5. Kapcsoló, Inverter és Led	18
5.6. 2 Kapcsoló, Vagy kapu és Led (1. rész)	19
5.7. 2 Kapcsoló, Vagy kapu és Led (2. rész)	20
5.8. Inverter visszakötve és Led (1. rész)	21
5.9. Inverter visszakötve és Led (2. rész)	22
5.10. Kapcsoló, Vagy kapu visszakötve és Led (1. rész)	23
5.11. Kapcsoló, Vagy kapu visszakötve és Led (2. rész)	24

5. Szkeleton tervezése

5.1. Errata

Az előző fejezetben leírtak egy apró részletben megváltoztak. Az elemeket már nem közvetlenül kötjük össze, hanem *vezetékek* segítségével, melyeket egymással *csomópont*okkal lehet összekötni, ha szükséges. Így javítottuk a láthatósággal kapcsolatosan felmerült problémákat, ehhez fel kellett venni 2 új osztályt (Wire, Node), illetve az AbstractComponent módosítani, ezekhez tartozó objektum és osztályleírások alább olvashatóak, valamint mellékelünk a módosított statikus osztálydiagramot is. (Egy-két egyéb objektumleírás is módosult, de csak azért mert a kiértékelés logikája változott – nem hátulról megyünk, hanem az összes kiértékeli magát, ez nem szükséges a jelen fejezethez, hiszen magától értetődő)

5.1.1. Objektumleírás: **Wire**

Vezeték, mely az áramköri komponensek ki és bemeneteit köti össze. Egy vezetéket egy darab kimenetet és egy darab bemenetet köt össze. A rajta lévő értéket le lehet tőle kérdezni, illetve be lehet azt állítani.

5.1.2. Objektumleírás: **Node**

Csomópont, mely a bemenetén lévő értéket a kimeneteire adja. Segítségével lehet egy vezetéket „szétágaztatni”.

5.1.3. Osztályleírás: **AbstractComponent**

Absztrakt osztály.

- Felelősség
Egy komponens absztrakt megvalósítása, ebből származik az összes többi komponens. A közös logikát valósítja meg. A gyakran használt feladatokra ad alapértelmezett implementációt (pl. vezetékek bekötése). Tudja magáról, hogy a legutóbbi két kiértékelés között változtak-e a kimenetei.
- Ősosztályok: (nincs)
- Interfészek: (nincs)
- Attribútumok
 - `protected Wire[] inputs`: Bemeneteire kötött vezetékek.
 - `protected Wire[] outputs`: Kimeneteire kötött vezetékek.
- Metódusok
 - `addTo(Circuit c)`: Meghívja az áramkör `add(AbstractComponent ac)` metódusát.
 - `void evaluate()`: Komponens kimenetein lévő értékek kiszámolása a bemenetek alapján.
 - `boolean isChanged()`: Visszaadja, hogy a legutóbbi két kiértékelés között változtak-e a kimenetek.
 - `void setInput(int inputPin, Wire wire)`: Az adott bemeneti lábára rákötjük a megadott vezetéket.
 - `void setOutput(int outputPin, Wire wire)`: Az adott kimeneti lábára rákötjük a megadott vezetéket.

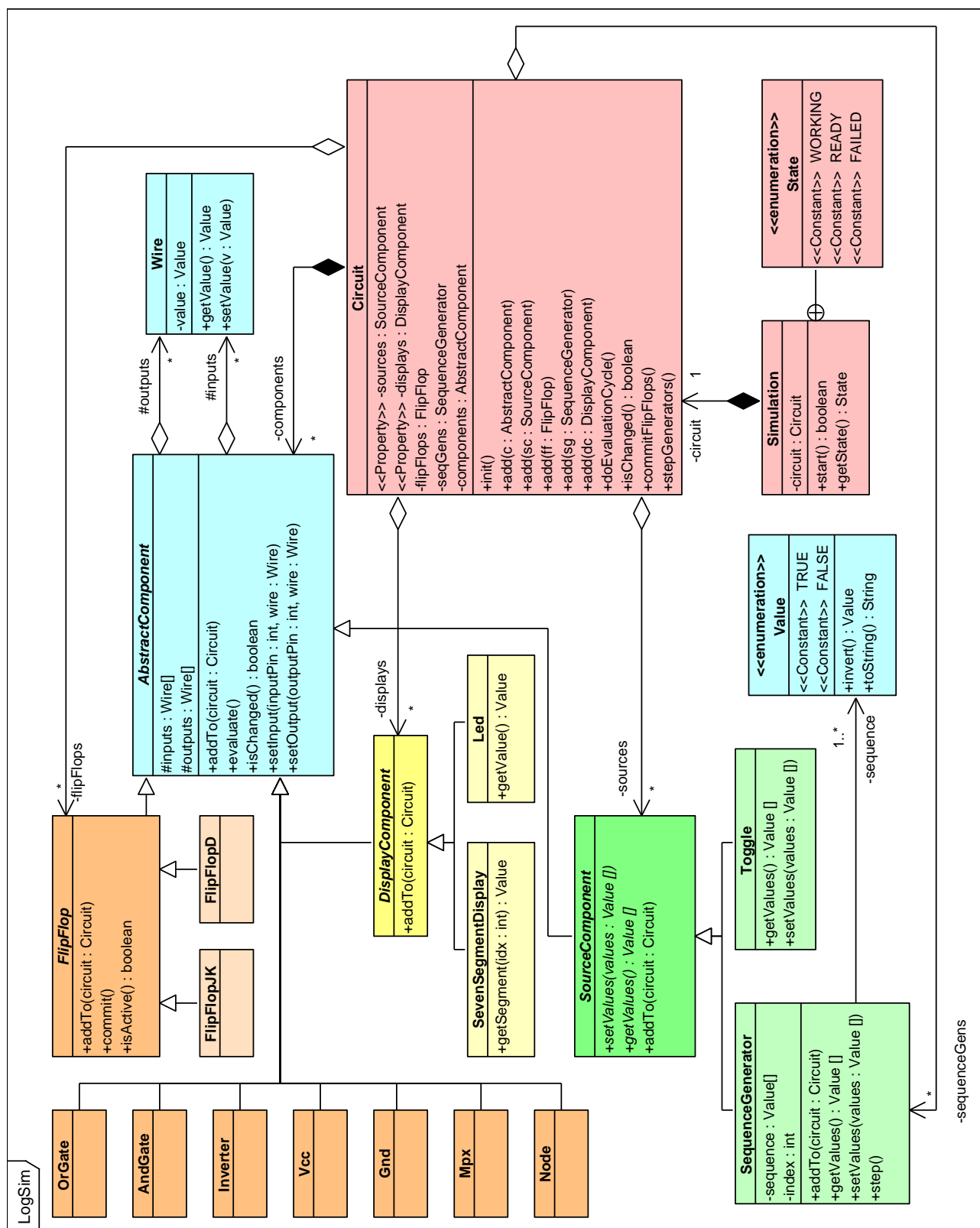
5.1.4. Osztályleírás: **Node**

- Felelősség
Csomópont, mely a bemenetén lévő értéket a kimeneteire adja. Segítségével lehet egy vezetéket „szét-ágaztatni”.
- Ősosztályok: AbstractComponent.
- Interfészek: (nincs)
- Attribútumok
 - (nincs)
- Metódusok
 - (nincs)

5.1.5. Osztályleírás: **Wire**

- Felelősség
Vezeték, mely az áramköri komponensek ki és bemeneteit köti össze. Egy vezetéket egy darab kimenetet és egy darab bemenetet köt össze. A rajta lévő értéket le lehet tőle kérdezni, illetve be lehet azt állítani.
- Ősosztályok: AbstractComponent.
- Interfészek: (nincs)
- Attribútumok
 - `private Value value`: Vezetéken lévő érték
- Metódusok
 - `Value getValue()`: Visszaadja a vezetéken lévő értéket.
 - `void setValue(Value v)`: Beállítja a vezetéken lévő értéket.

5.1.6. Statikus struktúra diagramok



5.1. ábra. Statikus struktúra nézet

5.2. A szkeleton modell valóságos use-case-ei

5.2.1. Use-case diagram



5.2. ábra. A szkeleton modell valóságos use-case-ei

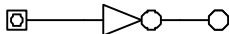
5.2.2. Use-case leírások

Lenti use-caseknél, ahol valamilyen információ szerint dönteni kell, vagy csak szükségünk van a skeleton jellegéből adódó hiányzó információkra, azt a felhasználótól kérjük be, ahhoz, hogy a szekvenciadiagramokon lévő szekvenciákat kapjunk a javasolt értéket kell beírnia a tesztelőnek.

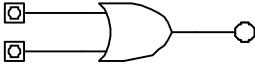
Use-case neve	Áramkör inicializálása
Rövid leírás	Ez a usecase egy áramkör és a hozzá tartozó szimuláció inicializálását mutatja be, hogyan jönnek létre a komponensek és a közöttük lévő összeköttetés. Jelen példa egy Kapcsoló és egy Led összeköttetését prezentálja.
Aktorok	Tesztelő
Forgatókönyv	<ul style="list-style-type: none"> • szimuláció létrehozása • áramkör létrehozása • áramkör bejegyzése a szimulációba • áramkör inicializálása <ul style="list-style-type: none"> – kapcsoló létrehozása – vezeték létrehozása – kapcsoló kimenetére vezeték kötése – led létrehozása – led bemenetére vezeték kötése – kapcsoló áramkörbe regisztrálása – led áramkörbe regisztrálása

Use-case neve	Kapcsoló és Led
Rövid leírás	Ez a usecase egy olyan áramkör tesztelését mutatja be, amely egy kapcsolóból és rá kötött ledből áll.
	Kapcsoló és Led

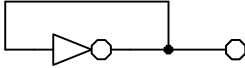
Aktorok	Tesztelő
Forgatókönyv	<ul style="list-style-type: none"> • Áramkör és komponensek létrehozása • szimuláció indítása <ul style="list-style-type: none"> – hálózat kiértékelés indítása <ul style="list-style-type: none"> * kapcsoló kiértékelése <ul style="list-style-type: none"> · kapcsoló állapotának lekérdezése (megkérdezi a tesztelőt, javasolt: 1) · kapcsoló értékének kiadása a vezetékre * led kiértékelése <ul style="list-style-type: none"> · led bemenetének lekérdezése (megkérdezi a tesztelőt, javasolt: 1) · bemenettől függően világít/nem világít. – áramkör változásának vizsgálata <ul style="list-style-type: none"> * kapcsoló változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 1) * led változásának vizsgálata (<i>idáig nem kéne eljutni, ha fent jól válaszolt a tesztelő</i>) – áramkör változott, ezért új ciklus – hálózat kiértékelés indítása <ul style="list-style-type: none"> * kapcsoló kiértékelése <ul style="list-style-type: none"> · kapcsoló állapotának lekérdezése (megkérdezi a tesztelőt, javasolt: 1) · kapcsoló értékének kiadása a vezetékre * led kiértékelése <ul style="list-style-type: none"> · led bemenetének lekérdezése (megkérdezi a tesztelőt, javasolt: 1) · bemenettől függően világít/nem világít. – áramkör változásának vizsgálata <ul style="list-style-type: none"> * kapcsoló változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 0) * led változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 0) – áramkör nem változott, stabil állapot – FF-okat véglegesítjük (nem történik semmi, mert nincs FF) – jelgenerátorokat léptetjük (nem történik semmi, mert nincs jelgenerátor) – szimuláció vége

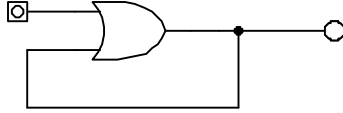
Use-case neve	Kapcsoló, Inverter és Led
Rövid leírás	<p>Ez a usecase egy olyan áramkör tesztelését mutatja be, amely egy kapcsolóból egy rá kötött inverterből és egy arra kötött ledből áll.</p> <p style="text-align: center;">Kapcsoló, Inverter és Led</p> 

Aktorok	Tesztelő
Forgatókönyv	<ul style="list-style-type: none"> • Áramkör és komponensek létrehozása • szimuláció indítása <ul style="list-style-type: none"> – hálózat kiértékelés indítása <ul style="list-style-type: none"> * kapcsoló kiértékelése <ul style="list-style-type: none"> · kapcsoló állapotának lekérdezése (megkérdezi a tesztelőt, javasolt: 1) · kapcsoló értékének kiadása a vezetékre * inverter kiértékelése <ul style="list-style-type: none"> · bemenet lekérdezése (megkérdezi a tesztelőt, javasolt: 1) · kimenet kiadása a vezetékre (megkérdezi a tesztelőt, javasolt: 0) * led kiértékelése <ul style="list-style-type: none"> · led bemenetének lekérdezése (megkérdezi a tesztelőt, javasolt: 0) · bemenettől függően világít/nem világít. – áramkör változásának vizsgálata <ul style="list-style-type: none"> * kapcsoló változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 1) * inverter változásának vizsgálata (<i>idáig nem kéne eljutni, ha fent jól válaszolt a tesztelő</i>) * led változásának vizsgálata (<i>idáig nem kéne eljutni, ha fent jól válaszolt a tesztelő</i>) – áramkör változott, ezért új ciklus – hálózat kiértékelés indítása <ul style="list-style-type: none"> * kapcsoló kiértékelése <ul style="list-style-type: none"> · kapcsoló állapotának lekérdezése (megkérdezi a tesztelőt, javasolt: 1) · kapcsoló értékének kiadása a vezetékre * inverter kiértékelése <ul style="list-style-type: none"> · bemenet lekérdezése (megkérdezi a tesztelőt, javasolt: 1) · kimenet kiadása a vezetékre (megkérdezi a tesztelőt, javasolt: 0) * led kiértékelése <ul style="list-style-type: none"> · led bemenetének lekérdezése (megkérdezi a tesztelőt, javasolt: 0) · bemenettől függően világít/nem világít. – áramkör változásának vizsgálata <ul style="list-style-type: none"> * kapcsoló változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 0) * inverter változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 0) * led változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 0) – áramkör nem változott, stabil állapot – FF-okat véglegesítjük (nem történik semmi, mert nincs FF) – jelgenerátorokat léptetjük (nem történik semmi, mert nincs jelgenerátor) – szimuláció vége

Use-case neve	2 Kapcsoló, Vagy kapu és Led
Rövid leírás	<p>Ez a usecase egy olyan áramkör tesztelését mutatja be, amely egy vagy kapura kötött két kapcsolóból és a vagy kapu kimenetére kötött ledből áll.</p> <p style="text-align: center;">2 Kapcsoló, Vagy kapu és Led</p> 
Aktorok	Tesztelő
Forgatókönyv	<ul style="list-style-type: none"> • Áramkör és komponensek létrehozása • Szimuláció indítása <ul style="list-style-type: none"> – hálózat kiértékelés indítása <ul style="list-style-type: none"> * 1. kapcsoló kiértékelése <ul style="list-style-type: none"> · kapcsoló állapotának lekérdezése (megkérdezi a tesztelőt, javasolt: 0) · kapcsoló értékének kiadása a vezetékre * 2. kapcsoló kiértékelése <ul style="list-style-type: none"> · kapcsoló állapotának lekérdezése (megkérdezi a tesztelőt, javasolt: 1) · kapcsoló értékének kiadása a vezetékre * VAGY kapu kiértékelése <ul style="list-style-type: none"> · kapu egyik bemenetének lekérdezése (megkérdezi a tesztelőt, javasolt: 0) · kapu másik bemenetének lekérdezése (megkérdezi a tesztelőt, javasolt: 1) · kapu értékének kiadása a vezetékre * led kiértékelése <ul style="list-style-type: none"> · led bemenetének lekérdezése (megkérdezi a tesztelőt, javasolt: 1) · bemenettől függően világít/nem világít. – áramkör változásának vizsgálata <ul style="list-style-type: none"> * 1. kapcsoló változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 0) * 2. kapcsoló változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 1) * kapu változásának vizsgálata (<i>idáig nem kéne eljutni, ha fent jól válaszolt a tesztelő</i>) * led változásának vizsgálata (<i>idáig nem kéne eljutni, ha fent jól válaszolt a tesztelő</i>) – áramkör változott, ezért új ciklus – hálózat kiértékelés indítása <ul style="list-style-type: none"> * 1. kapcsoló kiértékelése <ul style="list-style-type: none"> · kapcsoló állapotának lekérdezése (megkérdezi a tesztelőt, javasolt: 0) · kapcsoló értékének kiadása a vezetékre

	<ul style="list-style-type: none">* 2. kapcsoló kiértékelése<ul style="list-style-type: none">· kapcsoló állapotának lekérdezése (megkérdezi a tesztelőt, javasolt: 1)· kapcsoló értékének kiadása a vezetékre* VAGY kapu kiértékelése<ul style="list-style-type: none">· kapu egyik bemenetének lekérdezése (megkérdezi a tesztelőt, javasolt: 0)· kapu másik bemenetének lekérdezése (megkérdezi a tesztelőt, javasolt: 1)· kapu értékének kiadása a vezetékre* led kiértékelése<ul style="list-style-type: none">· led bemenetének lekérdezése (megkérdezi a tesztelőt, javasolt: 1)· bemenettől függően világít/nem világít.– áramkör változásának vizsgálata<ul style="list-style-type: none">* 1. kapcsoló változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 0)* 2. kapcsoló változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 0)* kapu változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 0)* led változásának vizsgálata (megkérdezi a tesztelőt, javasolt: 0)– áramkör nem változott, stabil állapot– stacionárius állapot, szimuláció vége
--	--

Use-case neve	Inverter visszakötte és Led
Rövid leírás	<p>Ez a usecase egy olyan áramkör tesztelését mutatja be, amely egy inverterből, amelynek kimenete egy ledbe illetve saját bemenetére van kötve. Oszcillálni fog, ezért a szimuláció rövid időn belül leáll.</p> <p style="text-align: center;">Inverter visszakötte és Led</p> 
Aktorok	Tesztelő
Forgatókönyv	<ul style="list-style-type: none"> • Áramkör és komponensek létrehozása • szimuláció indítása <ul style="list-style-type: none"> – hálózat kiértékelés indítása (3x) <ul style="list-style-type: none"> * inverter kiértékelése <ul style="list-style-type: none"> · bemenetén lévő értékek lekérése · kimenetére kötött érték kiszámolása és kiadása * csomópont kiértékelése <ul style="list-style-type: none"> · bemenetén lévő érték lekérése · kimeneteire az érték kiadása * led kiértékelése (világít/nem világít kijelzése) – áramkör változásának vizsgálata – harmadik lépés után sincs stacionárius állapot, szimuláció vége

Use-case neve	Kapcsoló, Vagy kapu visszakötte és Led
Rövid leírás	<p>Ez a usecase egy olyan áramkör tesztelését mutatja be, amely egy kapcsolóból, egy VAGY kapuból, melynek egyik bemenetére a kapcsoló, másik bemenetére a saját kimenete van kötve és egy ledből, melyre szintén a VAGY kapu kimenetét kötöttük. Ez egy olyan visszakötéses hálózat, mely stabil állapotban van.</p> <p style="text-align: center;">Kapcsoló, Vagy kapu visszakötte és Led</p> 
Aktorok	Tesztelő

Forgatókönyv	<ul style="list-style-type: none"> • Áramkör és komponensek létrehozása • szimuláció indítása <ul style="list-style-type: none"> – hálózat kiértékelés indítása (2x) <ul style="list-style-type: none"> * kapcsoló kiértékelése (állapotának kijelzése) * VAGY kapu kiértékelése <ul style="list-style-type: none"> · bemenetén lévő értékek lekérése · kimenetére kötött érték kiszámolása és kiadása * csomópont kiértékelése <ul style="list-style-type: none"> · bemenetén lévő érték lekérése · kimeneteire az érték kiadása * led kiértékelése (világít/nem világít kijelzése) – áramkör változásának vizsgálata – második lépés után stacionárius állapot¹, szimuláció vége
--------------	--

5.3. Architektúra

5.4. A szkeleton kezelői felületének terve, dialógusok

Az általunk elkészített szkeleton egy program váz melynek felülete egy egyszerű konzolos megjelenítési felület, amely alkalmas arra, hogy a use case-k által leírt teszteseteket bemutassuk. Az egyes tesztesetek a neki megfelelő use case sorszámaival van elnevezve, így program indítás után egy szám bevitelét követően a kiválasztott teszteset lefut. A teszteset futása közben kiír minden objektumot amin metódust hív, illetve kiírja a metódus nevét a paraméterekkel együtt, majd a visszatérési értéket. Ez azért lehetséges, mert a szkeleton már tartalmazza az elkészítendő szoftver összes fontos osztályát és metódusát, azonban az üzleti logikát még nem. Így könnyen eldönthető, hogy a use case-nek megfelelően viselkedik a program és továbbiakban képes lesz-e megfelelően működni. A tesztelési folyamat során döntési helyzet léphet fel. Ilyenkor a program felteszi a kérdést, majd a kapott válasz alapján folytatja a további futást. Ezzel csökkentjük a tesztesetek számát, anélkül, hogy bizonyos esetek kimaradnának a tesztelés alól. Futás közben megjegyzés formájában a program tájékoztat néhány elem valamilyen tulajdonságáról (például az adott bemenet miatt a kijelző világít, vagy nem) vagy bizonyos fontosabb lépésekről (például inicializálás). Az elvárás, hogy a szkeleton a szekvenciadiagramok által leírt működést mutassa. A program egyszerű és könnyen összehasonlítható formában írja ki a működését, amelyet könnyen összevethetjük a szekvencia diagrammokkal.

5.4.1. Program üzeneteinek formátuma

A program a következő eseményeket jelzi ki:

- Konstruktor hívás.
Formátum: `CREATE osztálynév objektumnév`
Példa: `CREATE Circuit circuit`
- Tagfüggvény hívás.
Formátum: `CALL objektumnév.metódus (paraméterek)`
Példa: `CALL wire.setValue (Value.TRUE)`

¹ ha a kapcsoló 0-ás állapotban van, akkor egy lépés alatt bekövetkezik, de érdekesebb szituáció, amikor 1-es állapotban van, ezt ábrázoljuk diagramon

- Konstruktor/tagfüggvény visszatér

Formátum: RETURN vagy RETURN objektumnév/érték

Ha van visszatérési érték az kétféle lehet: referencia valamelyik objektumra (ekkor az objektumnevet írjuk ki), vagy egy konkrét érték (ilyenkor az értéket, pl: false).

- Kérdés a felhasználónak.

Formátum: QUESTION objektumnév üzenet? [opciók]

Példa: QUESTION wire vezetéken lévő érték? [0/1]

Objektumnév annak az objektumnak a neve, aki kérdezi a felhasználót. Az opciókban lévő elemek / jellel vannak elválasztva. Addig nem megy tovább a program, amíg nem ezek közül kap választ.

- Megjegyzés kijelzése.

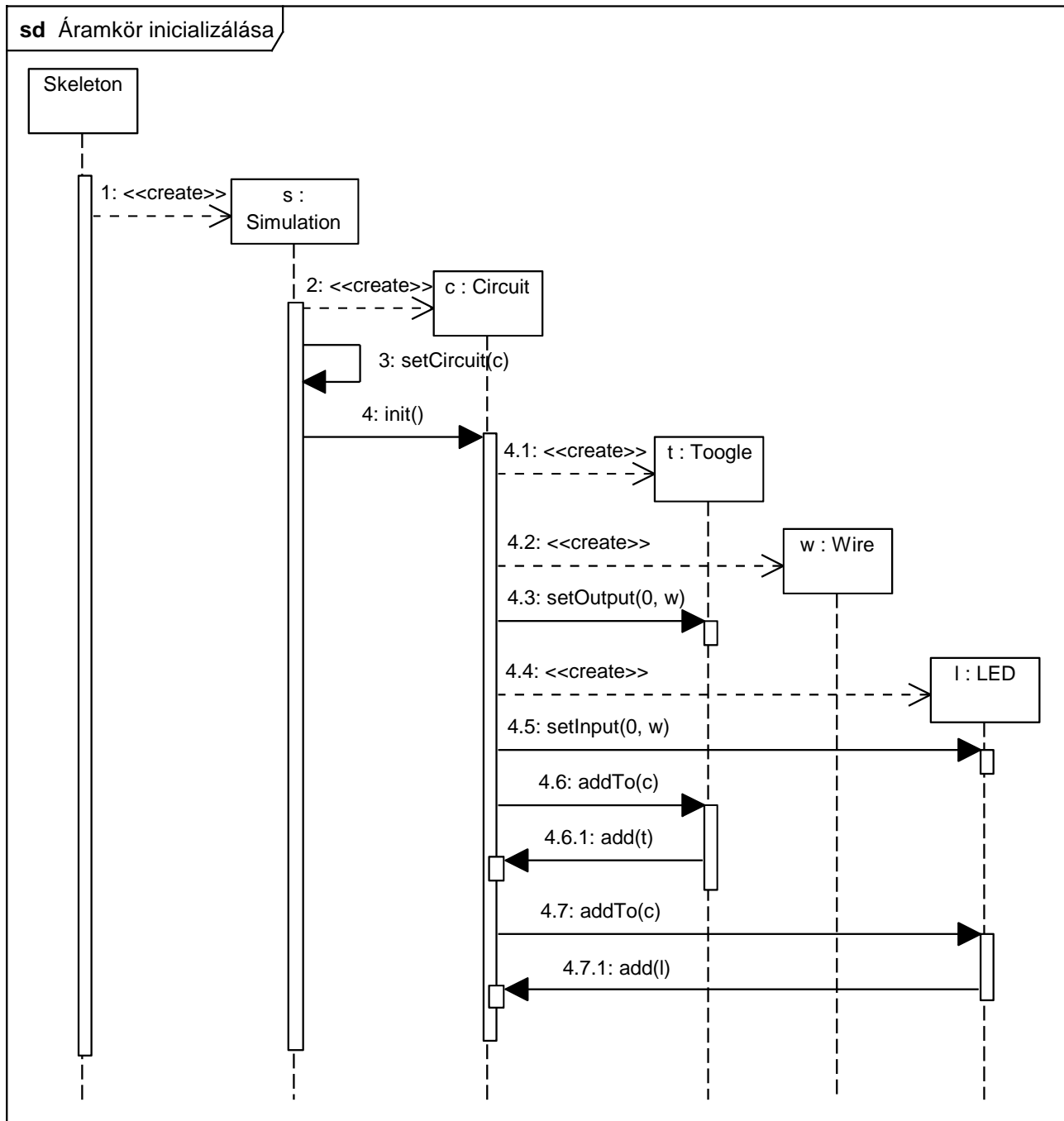
Formátum: # üzenet

Formátum bemutatásához egy összetettebb példa:

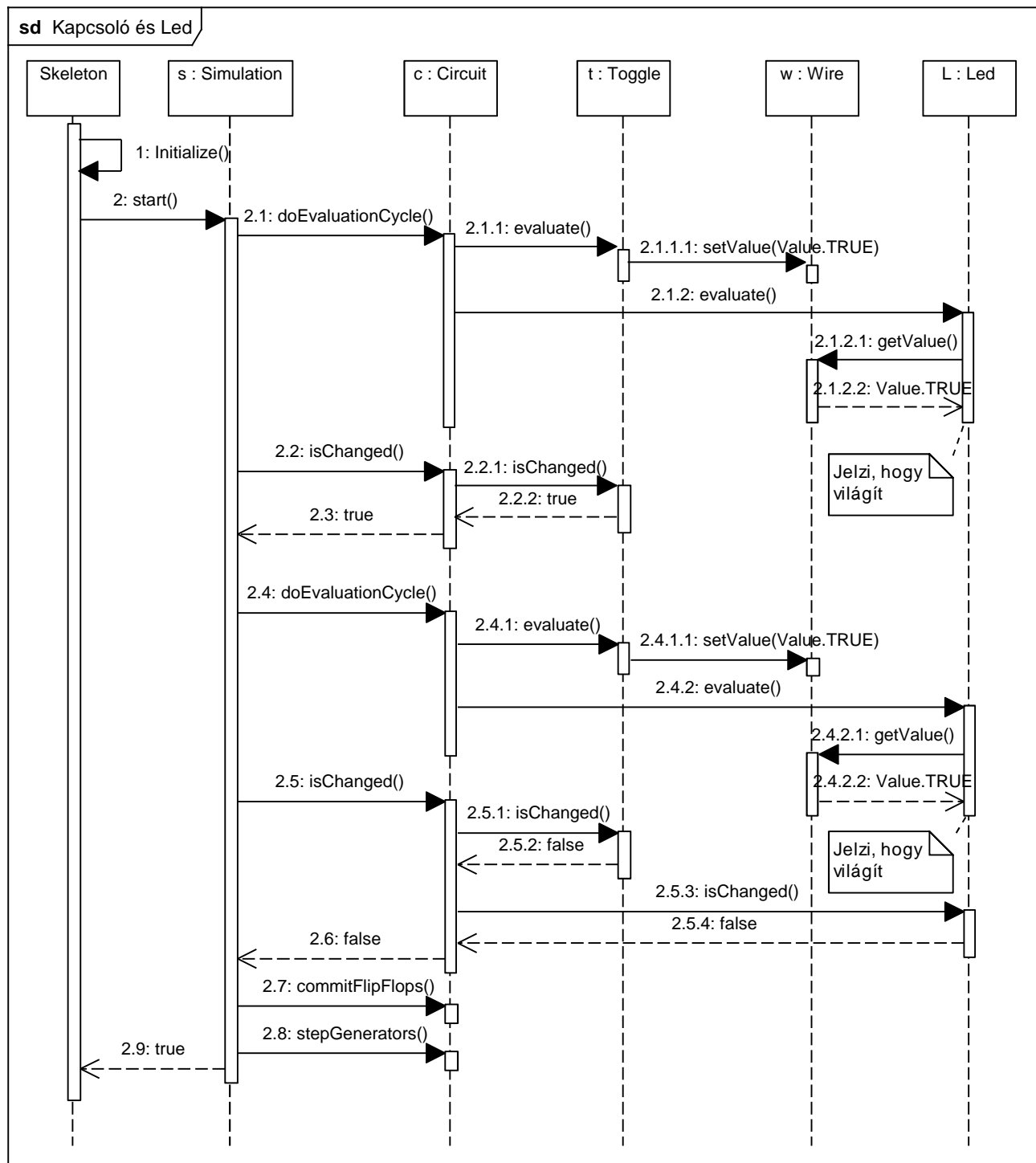
```
CALL simulation.start()
CALL circuit.doEvaluationCycle()
  CALL toggle.evaluate()
    QUESTION toggle állapot? [0/1] 1
    CALL wire.setValue(Value.TRUE)
    RETURN
  RETURN
  CALL led.evaluate()
    CALL wire.getValue()
    QUESTION wire vezetéken lévő érték? [0/1] 1
    RETURN Value.TRUE
    # világít
  RETURN
RETURN
CALL circuit.isChanged()
  CALL toggle.isChanged()
  QUESTION toggle változott? [0/1] 1
  RETURN true
RETURN true
CALL circuit.doEvaluationCycle()
  CALL toggle.evaluate()
    QUESTION toggle állapot? [0/1] 1
    CALL wire.setValue(Value.TRUE)
    RETURN
  RETURN
  CALL led.evaluate()
    CALL wire.getValue()
    QUESTION wire vezetéken lévő érték? [0/1] 1
    RETURN Value.TRUE
    # világít
  RETURN
RETURN
CALL circuit.isChanged()
  CALL toggle.isChanged()
  QUESTION toggle változott? [0/1] 0
  RETURN false
```

```
    CALL led.isChanged()
    QUESTION led változott? [0/1] 0
    RETURN false
RETURN false
CALL circuit.commitFlipFlops()
RETURN
CALL circuit.stepGenerators()
RETURN
RETURN true
```

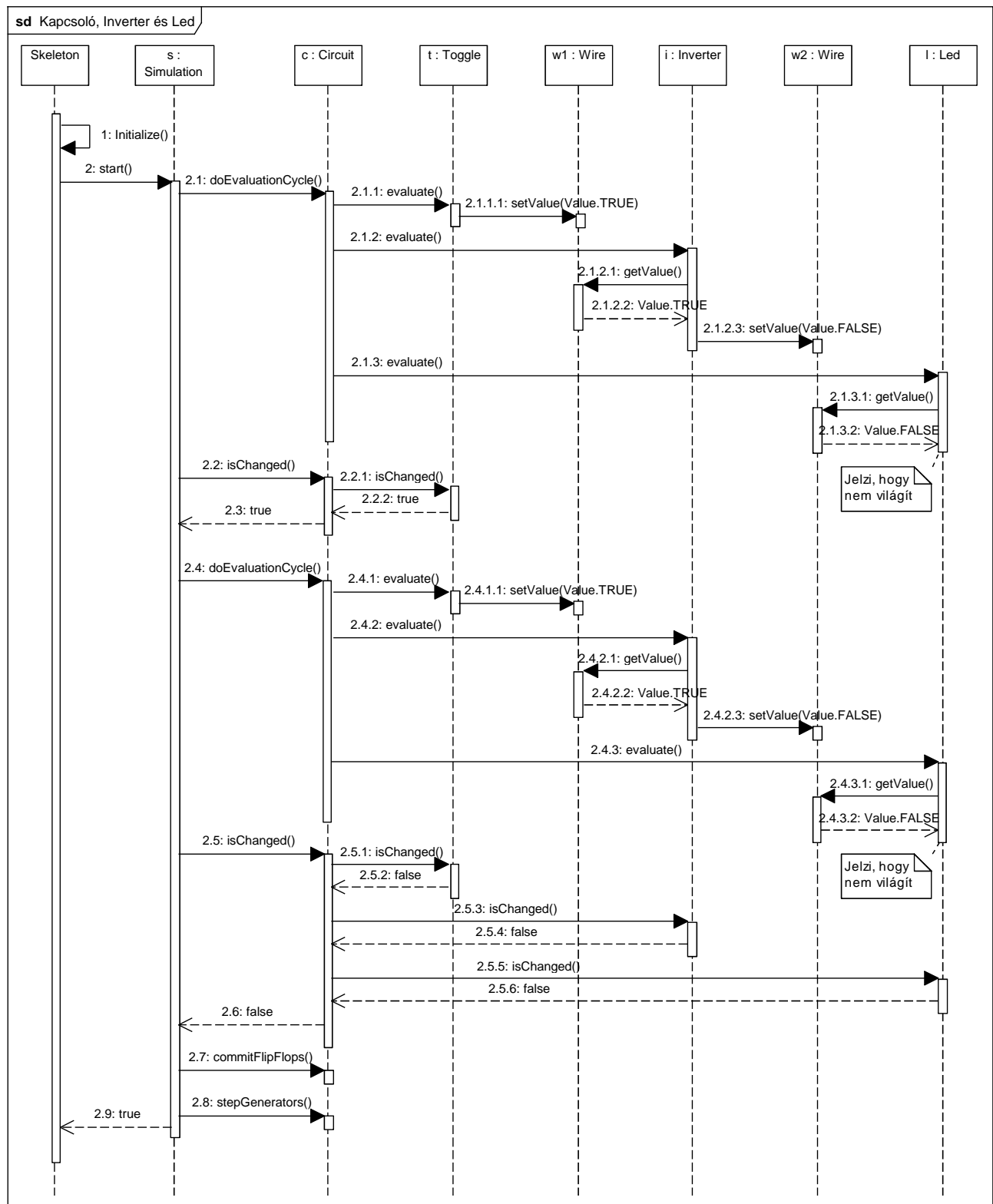
5.5. Szekvencia diagramok a belső működésre



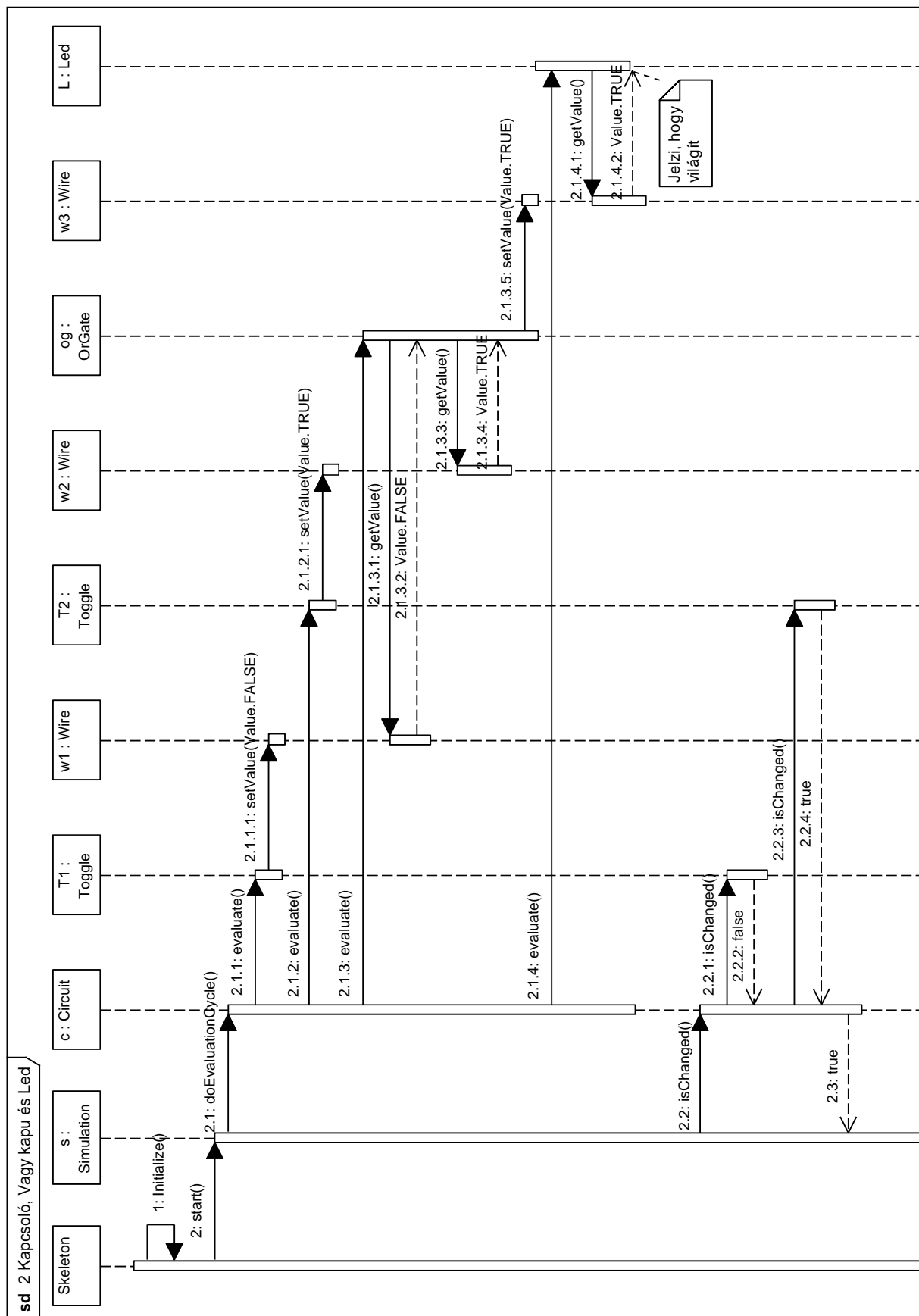
5.3. ábra. Áramkör inicializálása



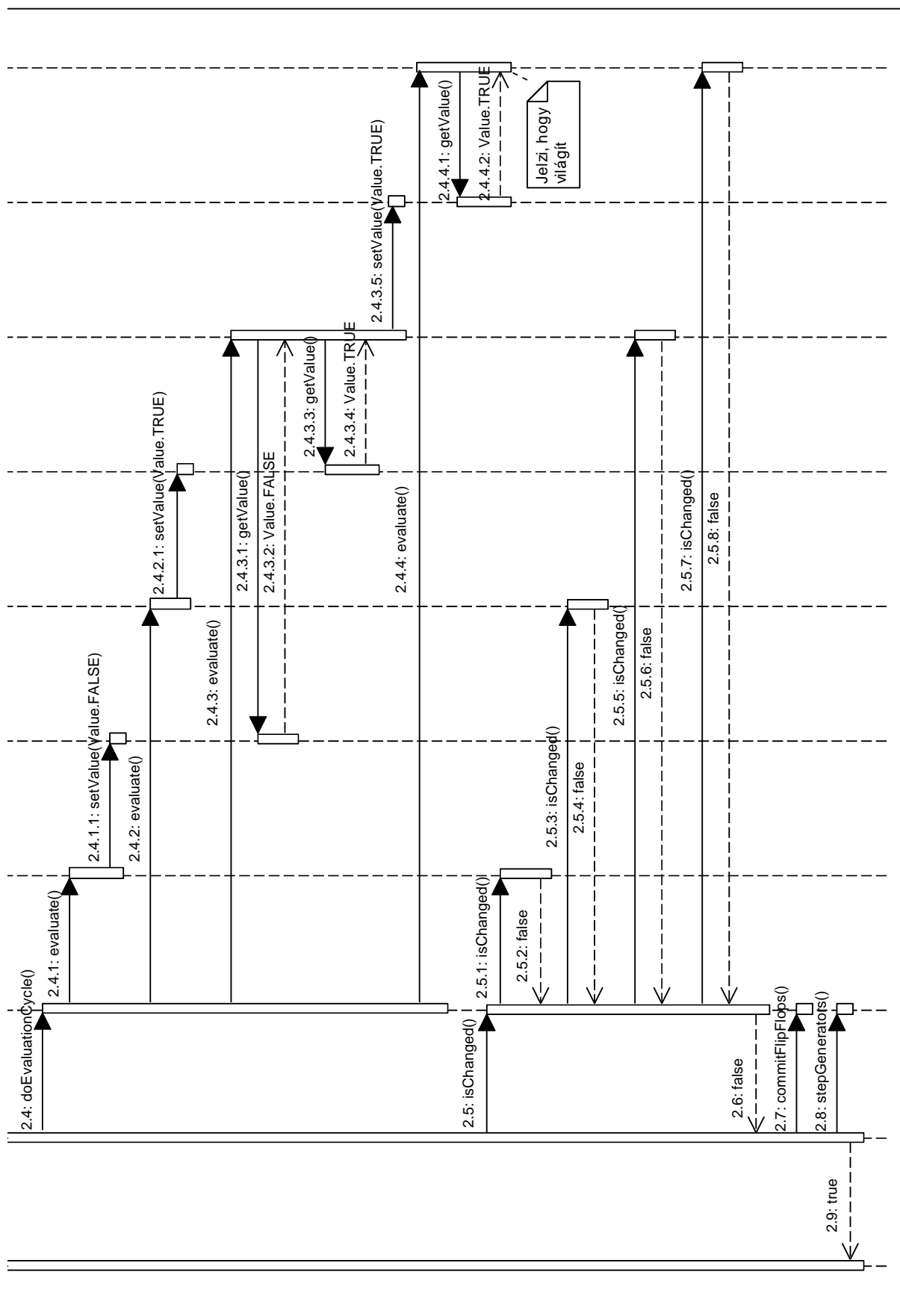
5.4. ábra. Kapcsoló és Led



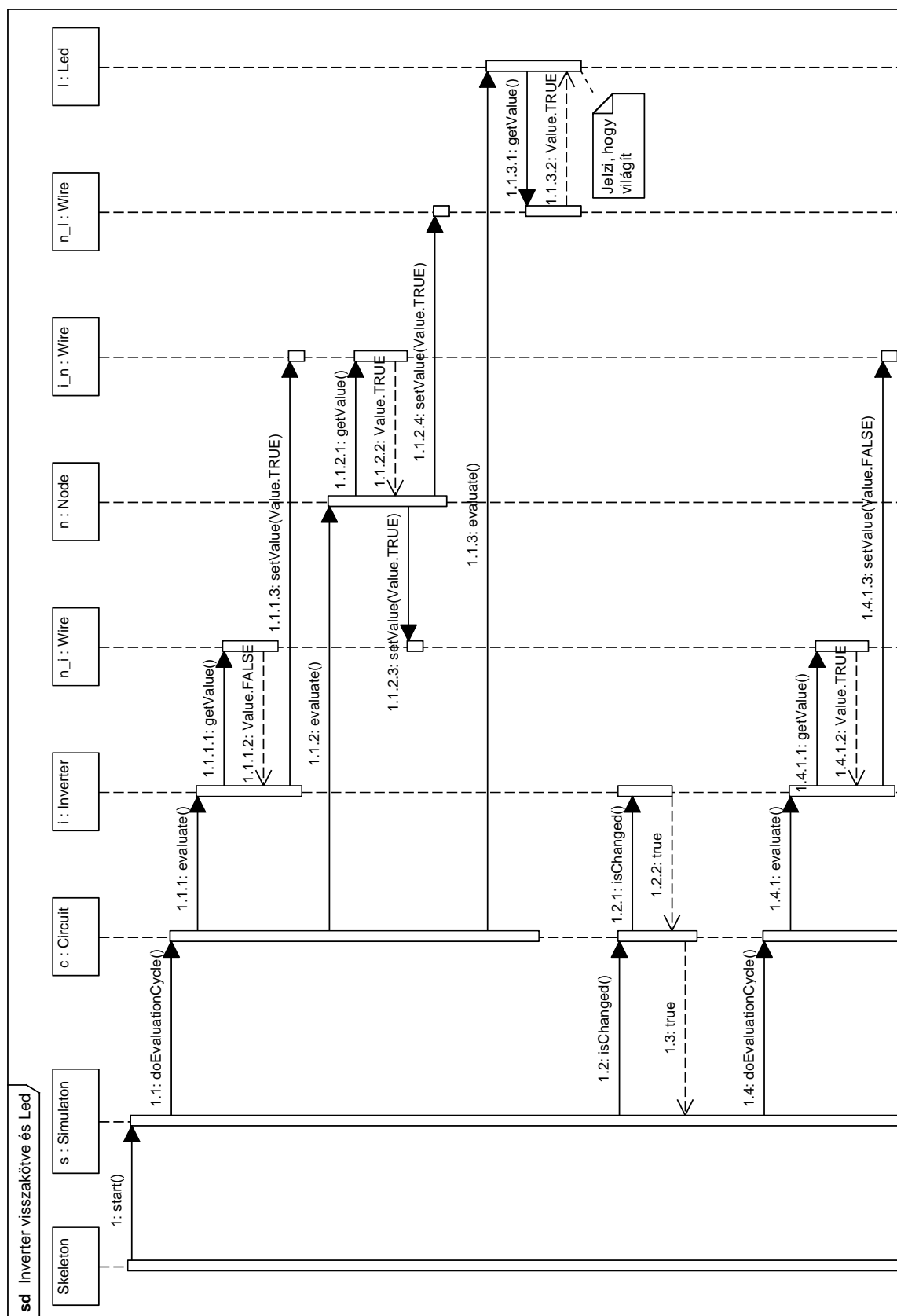
5.5. ábra. Kapcsoló, Inverter és Led



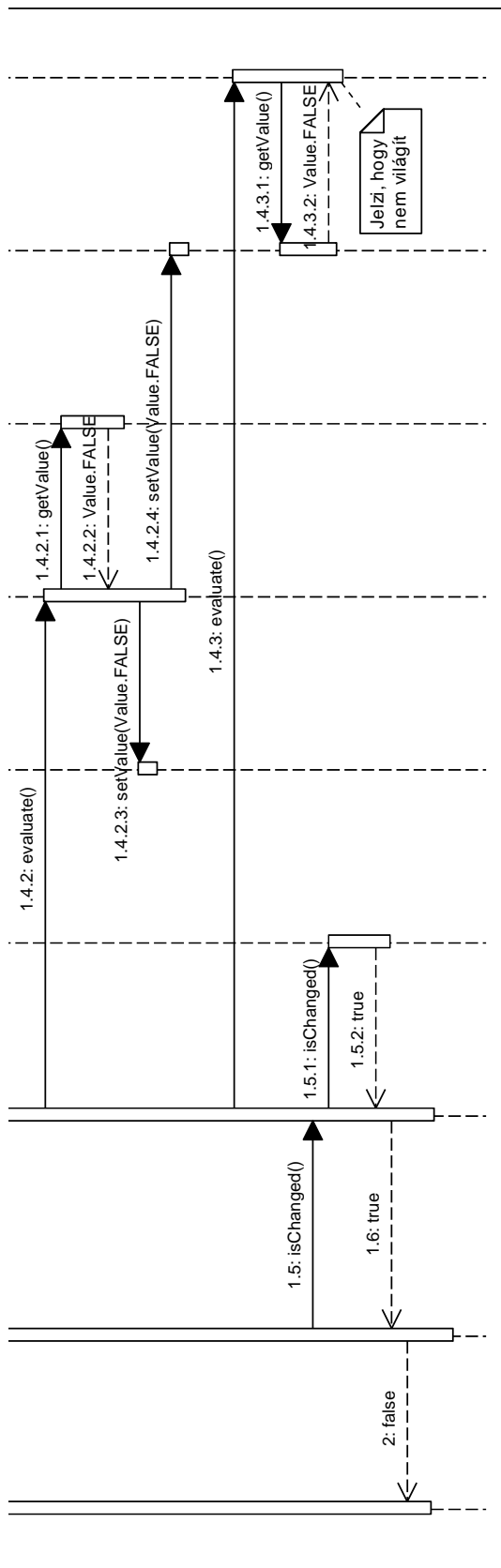
5.6. ábra. 2 Kapcsoló, Vagy kapu és Led (1. rész)



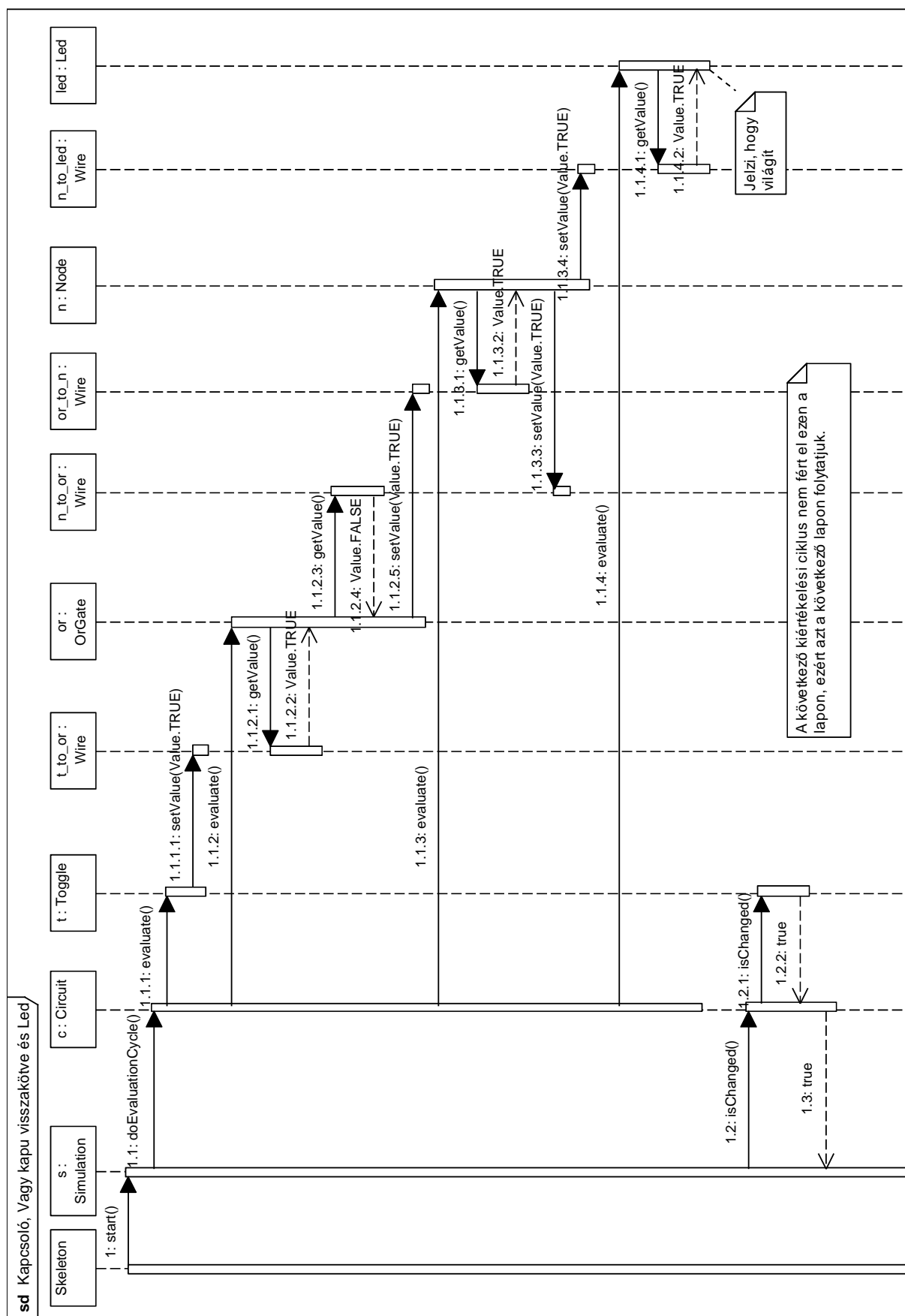
5.7. ábra. 2 Kapcsoló, Vagy kapu és Led (2. rész)



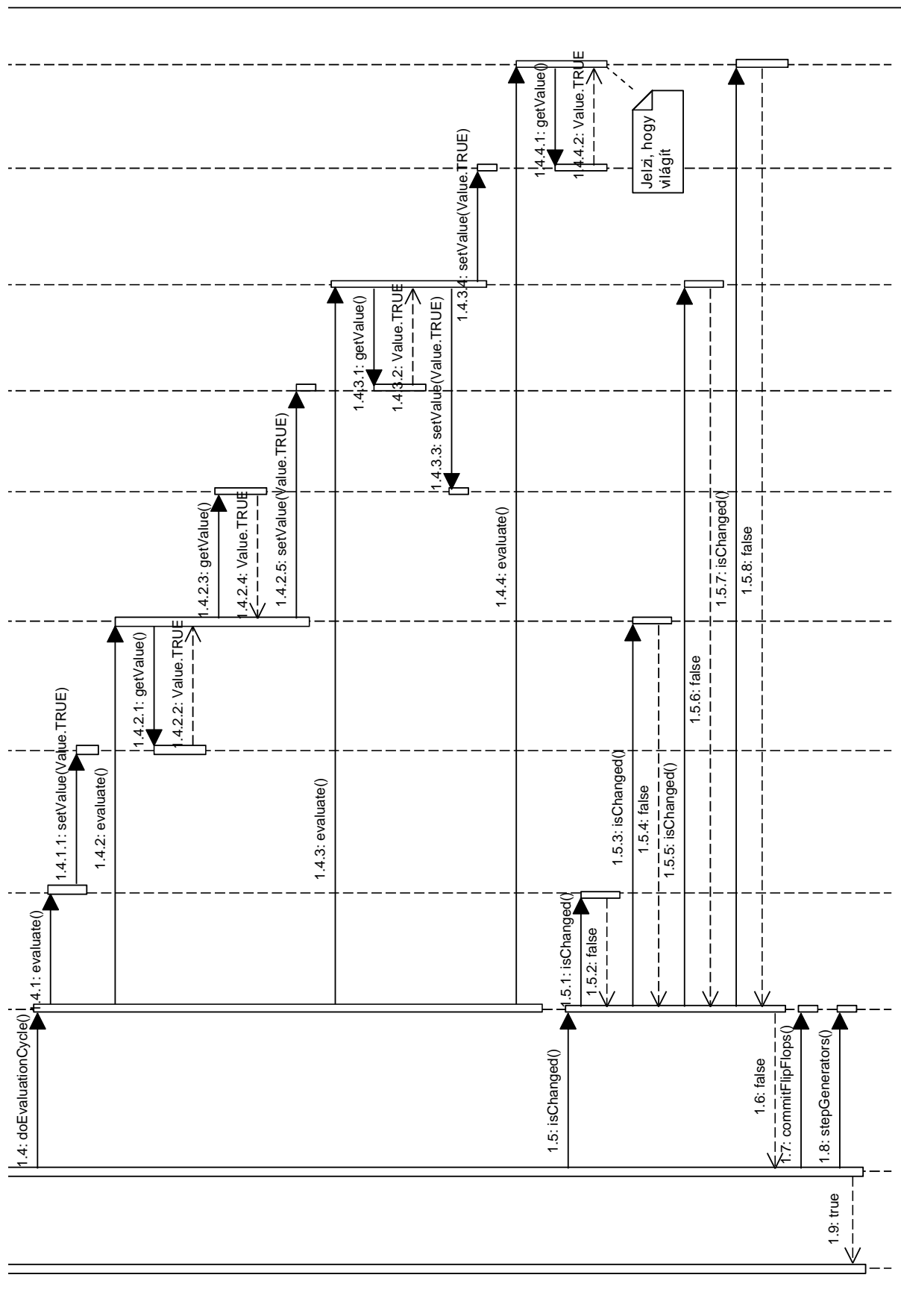
5.8. ábra. Inverter visszakötte és Led (1. rész)



5.9. ábra. Inverter visszakötte és Led (2. rész)



5.10. ábra. Kapcsoló, Vagy kapu visszakötte és Led (1. rész)



5.11. ábra. Kapcsoló, Vagy kapu visszakötte és Led (2. rész)

5.6. Napló

Kezdet	Időtartam	Résztevők	Leírás
2010.03.12. 14:00	1,5 óra	Kriván B.	Javasolt módosítások elvégzése az előző fejezetben, rövid errate készítése jelen fejezet elé.
2010.03.13. 00:00	2 óra	Péter T.	Use-casek leírása szöveges formátumban
2010.03.13. 09:30	30 perc	Kriván B.	Use-case diagram megrajzolása
2010.03.13. 10:00	2 óra	Kriván B.	Use-casek leírásának L ^A T _E X formátumra való alakítása, apróbb finomítások
...