## 5.4 Szekvencia diagramok a belső működésre

A belső működést leíró use case-ek szekvencia diagramjai elérhetőek a dokumentáció és 4.4 fejezetében, a változtatásra szoruló diák új diagramjai alább találhatóak.

A Tesztelő meghívhatja a DigitalObject Count metódusát, hogy ellenőrizze stabilitását.



A Tesztelő léptetheti a hálózatot a dB:DigitalBoard StepComponents() metódusának hívásával .



A többi diagramon nem történt változás.

*Ez nem igaz, mert a Switch szekvencia diagramjában átírtam dB:DigitalBoard és sw:Switch-re az objektumokat, de ezt az előbb se vették észre, és csak ezért nem fogom ide beletenni, hogy észrevegyék. Illetve ketté lett szedve a Build-nél a DigitalObjectek életvonalai.*

Egy konkrét hálózatra történő metódushívások dialógusai.

A hálózat egy jelgenerátorból, egy kapcsolóból, egy VAGY kapuból és egy LED-ből áll, valamint az ezeket az elemeket összekötő 3 darab vezetékből.

Feltételezzük, hogy a hálózatot már felépítettük, és a Tesztelő ezen végzi el a metódusok hívását.

StepComponent:



Toggle:



Generator:



Visszacsatolást tartartalmazó, stabil hálózat:

Egy kapcsolóból, egy ÉS kapuból, egy LED-ből és az ezeket összekötő két vezetékből épül fel.

StepComponent:



Instabil hálózat:

Tartalmaz egy kapcsolót, egy AND kaput, egy invertert, egy LED-et, és az ezeket összekötő három darab vezetéket.



*Mit fog tudni csinálni a tesztelő (milyen metódusokat fog tudni hívni)?*

*StepComponents, GetElementById, SetFrequency, SetSequence, Toggle, Run (Stop, Pause)*

[A szkeletonban implementált szekvenciadiagramok. Tipikusan egy use-case egy diagram. Ezek megegyezhetnek a korábban specifikált diagramokkal, de az egyes életvonalakat (lifeline) egyértelműen a szkeletonban példányosított objektumokhoz kell tudni kötni. Azt kell megjeleníteni, hogy a szkeletonban létrehozott objektumok egymással hogyan fognak kommunikálni.]