Extending Postsimulation by a Virtual Timebase

(Master Thesis)

LUKAS NEUBERGER

Motivation

Hardware-in-the-Loop (HiL) testing is an integral step in the development process of industrial plants. A HiL simulation tests an industrial controller, the System Under Test (SUT), by stimulating it with outputs of a simulated plant. It checks whether the output signals of the SUT can be matched with a specified reference signal and given tolerances. The output signals of the SUT can be recorded and then used in a so-called postsimulation. In a postsimulation, the same scenario is simulated again, but without the actual SUT. Instead, the recordings are played back during the simulation. This way, the output signals can be matched with different reference signals or tolerances. Usually, only the result of the postsimulation or the sequence leading up to a mismatch of the reference signal are relevant for analysis. It would be useful to skip the uninteresting parts and slow down the interesting ones. Furthermore, the postsimulation could be computed as fast as possible to show the results immediately.

State of the Art

Many simulations implement the capability to run faster or slower than real-time: physics simulations, weather simulation, even video games. Varying speed of time is often implemented using a mapping from real to a virtual time. S7-PLCSIM Advanced, a PLC emulator developed by Siemens, also utilizes this concept of virtual time to be able to emulate behaviour faster or slower than real-time.

Twistturn is a mobile Soft PLC and HiL simulation tool that uses an advanced method of signal matching for HiL simulations. It can record these simulations in order to run postsimulations with the recordings. Currently, postsimulations in Twistturn can only be executed in real time.

Objective

The objective of this thesis is to extend Twistturn's postsimulation with a virtual time base. To do that, the simulations has to decouple its clock source from the current system time and must be able to fully control it. The simulation speed should be adjustable before and during the postsimulation via an appropriate user interface, possibly similar to multimedia players. Therefore, the interface has to provide controls to run the simulation faster or slower than real-time, but also to pause it or execute it as fast as possible to obtain results quickly.

Approach

The first step is to analyze existing approaches to vary simulation speed and to determine a way to integrate such a feature into Twistturn. The current plan is to replace all methods in the simulation environment that obtain the current system time with a method that provides a virtual time, instead. Then, user controls to manipulate the virtual time, both before starting a postsimulation and during its execution, have to be implemented. At last, the correctness of this approach has to be checked by comparing real-time results to results obtained by varying the virtual time.