Regular Grammar Model for Protocol Testing

Csaba V. Rotter

This paper presents a new approach on conformance test modelling. Protocol testing is the practical way to check correctness of protocol implementation with respect to their specification. TTCN (Tree and Tabular Combined Notation) is the standardised test notation for the description of OSI conformance tests. Fundamental principles of protocol conformance testing is a regular exchange of PDU 's (Protocol Data Unit) between the peer entities and ASP 's (Abstract Service Primitives) to communicate to the neighbouring layer. The communication dialog between the tester and IUT (Implementation Under Test) is regarded as an exchange of words with help of well defined regular grammar. According to this new theory of modelling all PDU-s, ASP-s and practically all events defined on TTCN are considered as a set of alphabet on a regular language.

A language (Formal Language) is a set of sentences and a sentence is a finite sequence, or string of elements, each of which is drawn from a finite vocabulary.

A language can be defined by a particular grammar as the set of sentences it generates. The grammar is an ordered quadruple

$G = (\Sigma, V, P, S)$

where:

Σ - set of terminal symbols (PDU's, ASP's,....)

V - set of non-terminal symbols (states of the IUT, states of Tester)

P - set of the production rules on the system (Rules of the comm. dialog)

S - is the starting component of the generation (Idle state of IUT)

If we can find a set of symbols from which the participants (the tester and the IUT) compose their sent sentences and we can construct the grammar containing the rules correctly generating these sentences we can get the formal description of testing. The steps for constructing such a grammar are:

- to determine the set of symbols sent by the tester to IUT and to determine the set of symbols sent by the IUT to tester. The union of these two sets forms the set of symbols characterising the message exchange they will form the set of terminals.
- to provide the non-terminals, which can be all of the IUT 's "state" where it can arrive after changing a dialog with the tester
- to construct the set of production rules for the grammar following the sequence of the dialog

Finally we attempt to construct such as grammar for a real test case for DECT testing and we focus to the problems which can appear in these situation and the advantages and disadvantage of this method.