Setting up a Retrieval System for Design Reuse  
- Experiences and Acceptance -

Gerhard H. Büttner  
CAE/CAD Support Department  
Rohde & Schwarz GmbH & Co. KG  
Mühlhofstr. 15, D-81671 München

Abstract

Within the EU funded ESPRIT project 8369 "CONSOLE", ROHDE & SCHWARZ developed a demonstration software version of a retrieval system for design reuse. The system stores the design classifications in a database and is connected to the EDA systems from Zuken-Redac and Viewlogic for access (read or copy) to the schematic and layout design data. This software version is part of the user requirement definition of a "subcircuit browser", which will be developed as industrial prototype within the "CONSOLE" project by Zuken-Redac. ROHDE & SCHWARZ engineers are using this software since early 1994.

The presentation will cover the following main issues:
- Objectives of the system
- R&S definition of reusable designs (subcircuits, functional modules)
- Brief overview on the current system functionality
- Necessary in house rules for reusable designs
- Definition of families for design classification (i.e. power units, amplifiers)
- Introduction of "reusability" in the current design flow
- Work practice and acceptance experiences

1. Introduction

The objective of the CONSOLE project is to build an industrial prototype for a hierarchical design system for PCB design. Hierarchical elements could be of different types, such as
- daughter boards
- MCMs
- ASICs/FPGAs
- logical (schematic) subcircuits
- physical (layout) subcircuits (same or different layout system e.g. RF layout)
- mechanical subassemblies
- software modules

Designing a system in a hierarchical way is done for two main issues:
- Reducing the time to market by using resources (e.g. designers) concurrently.
- Reducing cost (and probably time to market) by using finished and proved "building blocks", like the mentioned hierarchical elements, which are called "functional modules" at ROHDE & SCHWARZ.

ROHDE & SCHWARZ concentrated within the CONSOLE project on the area of reuse of functional modules and integrated the reuse of functional modules into its design flow, including demonstration software of a retrieval system for functional modules. The design flow for reuse is shown in figure 1.

Fig. 1: Design Flow for Reuse

2. Basics and System Overview

A functional module has to have clearly defined interfaces and a specific function. A design project comprises one or several functional modules. In
project management of the R&S CAE/CAD system, a PCB, ASIC or software design corresponds to a project. A project is made up of at least one functional module. Functional-module management is linked to project management. A functional module itself is described by:

- project name,
- modification status of project,
- module name (of circuit schematic block for electrical functional modules).

For the ability of a clear description of a functional module within the retrieval system, the logical (schematic) hierarchy was introduced for all new PCB and ASIC design projects.

The highest level in a hierarchical circuit schematic of a project is the TOP block (figure 2). It contains either the circuit schematic of the functional module (if the project consists of one module only) or a block diagram (each block is given a name), describing the constituent functional modules of the project.

![Circuit schematic hierarchy](image)

**Fig. 2: Circuit schematic hierarchy**

This hierarchical structure is employed for each project and organises the circuit diagrams logically. It does not represent a physical hierarchy across projects (e.g. instrument -> PCB -> IC). This physical hierarchy is managed in the project administration.

### 3. Classification

A functional module is characterised first by the type of system it is designed with (e.g. electrical CAD, RF CAD, software), second a family classification and third specific set of up to 20 different attributes for each family. This attributes are standardised over all families; for example is the attribute "clock speed" the same for CPU modules or "DIGITAL INTERFACE" modules and has the same unit for both families.

Engineers from all ROHDE & SCHWARZ business units defined families and family specific attributes for 26 different hardware and software functions. The families are currently:

```
DC/DC-CONVERTER
A/D-CONVERTER
D/A-CONVERTER
DEMODULATOR
EICHLEITUNG
FILTER ANALOG
HF-AMPLIFIER
INTERFACE DIGITAL
COUPLER
MISCHER
MODULATOR
POWER UNITS
QUARZ- OR DIEL.RES.OSZ
CPU
SYNTHESISER
VCO- OR YIG-OSZILLATOR
VSWR-BRIDGE
SW:BASIS-SYSTEM
SW:DB-APPLIKATION
SW: HARDWARE-DRIVER
SW:HELP-FUNKTION
SW:PROTOKOLL-DRIVER
SW:STARTUP-MODUL
SW:SYSTEM-INTERFACE
SW:TEST-FUNKTION
SW:USER-INTERFACE
```

An additional family for modules without a specific function is also available, where only comments can be assigned. The families definition, that is family name and the specific attributes are fixed for the
users and can only be modified by the standardisation department.

As an example the family specific attributes for CPU modules are shown below together with the standardised units:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor type</td>
<td>DSP56002</td>
</tr>
<tr>
<td>Processor clock</td>
<td>40 MHz</td>
</tr>
<tr>
<td>RAM</td>
<td>0.064 SRAM Mbyte</td>
</tr>
<tr>
<td>ROM</td>
<td>Mbyte</td>
</tr>
<tr>
<td>LCD interface</td>
<td>N</td>
</tr>
<tr>
<td>DISK interface</td>
<td>N</td>
</tr>
<tr>
<td>Floppy interface</td>
<td>N</td>
</tr>
<tr>
<td>External bus</td>
<td>24BIT-DSP-BUS</td>
</tr>
<tr>
<td>Parallel interface</td>
<td>32/8/8BIT BEM01/02/03</td>
</tr>
<tr>
<td>Serial interface</td>
<td>SYNC SSI, ASYNC 8BIT SCI</td>
</tr>
<tr>
<td>Power supply</td>
<td>5,200 V, mA</td>
</tr>
<tr>
<td>Oper. temp. range</td>
<td>Grad C</td>
</tr>
<tr>
<td>MTBF</td>
<td>h</td>
</tr>
<tr>
<td>Dimensions (LxWxH)</td>
<td>90X80X10 mm</td>
</tr>
<tr>
<td>Operating system</td>
<td></td>
</tr>
<tr>
<td>Prod. cost (HK1)</td>
<td>TDM</td>
</tr>
<tr>
<td>Dev. engineer</td>
<td>GERHARD BUETTNER</td>
</tr>
</tbody>
</table>

Fig 3: Module report for a CPU module

4. Retrieval System Functionality

The retrieval system is a database application based on INGRES. All module attributes are stored in this database. There is a "MOTIF" window application for the users access to this database. This application has four main features:

- definition/modification of attributes and families
- module listing (ordered by system, project or family)
- module search by family specific attributes
- module description

If a module is found by a search or within the listing window, all values of the attributes are shown in a module data window together with some organisational and statistical informations (e.g. name of the developer, date or number of copies taken from this module). From this window it is possible to print reports. Figure 3 shows a typical report for a CPU module, or to start viewing or copying its CAD data (schematic and probably layout). We decided, not to allow only to instance the data, because we did not want to extend our modification control system onto the module level below the level of products.
We also thought, that it will be much more reasonable to copy the data into the new project, because they are then easier to modify and adapt to the new project.

5. Experiences

The introduction of logical hierarchy into the schematic design flow was the first problem. The developers, designers, production and service were used over many years, to deal with flat schematics only. Therefore the hierarchy concept was consequently trained in classes for developers and designers. When the first hierarchical schematics came to production (especially testing departments), we had to learn, that every person in the production and also the service has to have a additional training in reading "hierarchical schematics". Meanwhile the concept of hierarchical schematics is integrated into the obligatory internal design rules. Setting up the families and family specific attributes was the next obstacle. To get the actual set of 26 families, with the necessary attributes the discussions with the representatives of the engineering departments took more than a year. The specific problem was, that we have a very big variety of products (from measurement equipment to power TV transmitters). As an example: The measurement equipment departments are speaking from "power amplifiers" in the range of "milliwatts", whereas the TV transmitter departments speak from the same in the range of "hundreds of watts".

The biggest problems arose, when we started to get module data into the database for the retrieval system. In a first step we tried to ask the engineers and feed in the data ourselves. For only 25% of all actual development projects, we got the informations. For 75% we got no information because of

- the module development was not stable enough,
- the engineer had no time,
- the engineer did not exactly know the module data (only the data for complete system were known),
- the engineers had the fear of questions to their modules

But in the most cases the schematic wasn't done by the development engineer, but by the design engineer, though we had no direct access to the developers and these developers were not directly included into the design cycle.

Another big problem is, that the development departments are divided into a lot different profit centres. We are still confronted with the demand for an internal accounting system, when copying module data from one project to another. There is also another demand for a bonus system, to give the engineers an incentive, both, when they describe a module for the retrieval system and when they use such a module.

After one year since the first data was described, a lot of the CAD data is in archive and therefore no longer under direct access. This directly breaks the design cycle for reuse. We are now thinking if it is possible to access the archive data in TIFF format instead of the CAD data for viewing the module schematic.

In parallel to this "official retrieval system", design reuse is in practice in single departments, by using the drag and drop or group mechanisms of the schematic editors and a direct "know how and module data transfer" from one engineer to another.

6. Conclusion

To achieve effective design reuse in a larger company, there is much more necessary than only a computer based retrieval system. The same or even a bigger assumption is, to get the development engineers integrated in the computer based design cycles. This means, that they have to do the schematics themselves with a schematic editor. There is a need to think in modules and hierarchy terms for the complete life cycle of a product. That means from development over design and production to service. It is also necessary, to break commercial and organisatorial obstacles between different departments and come to a company wide design reuse, starting with the reuse within one department, ending with approved modules supported by "Centers of Competence".