



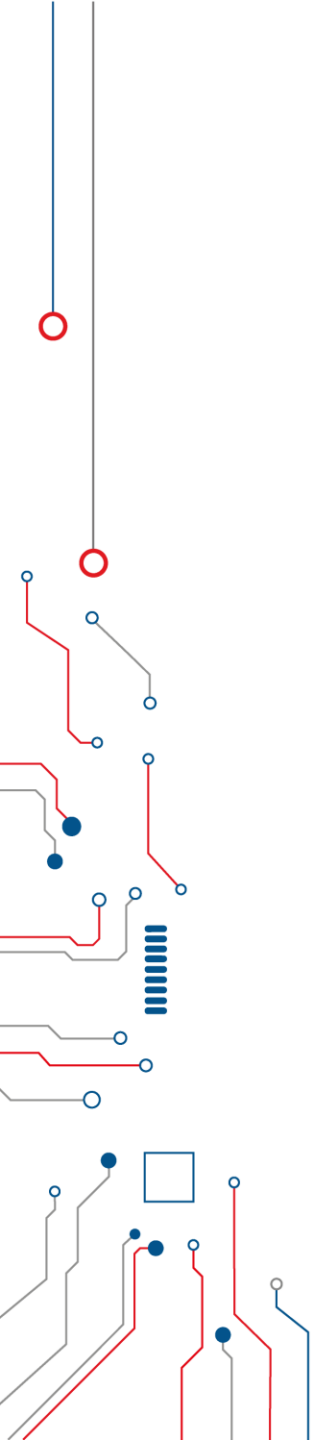
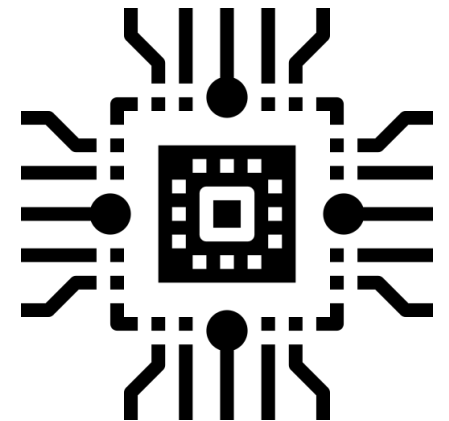
Leibniz Institute
for high
performance
microelectronics

LibMan - an easy way to manage your open source design flow

Dr. Anton Datsuk - Scientist Technology / Research & Prototyping Service

Networking Workshop FMD-QNC 'OpenPDK, OpenTooling and Open Source Design – An Initiative to Push Development'

27 June 2023



Outline



1. Motivation
2. Goals
3. Main interface of library manager
4. IHP OpenPDK data structure
5. LibMan functionality
6. Documentation generator and static analysis tool
7. Summary



Motivation



- The commercial software like Cadence Virtuoso, Keysight ADS, Mentor Tanner offer Library Manager
- Library Manager is used to
 - store design data in libraries
 - navigate libraries, cells, views, and files in the directory structure by opening a cell views
 - manipulate design data in a more friendly way
 - execute view specific editors
- There's no Library Manager in Open Source community to unite various EDA tools in a flow (at least known to us)

Goals

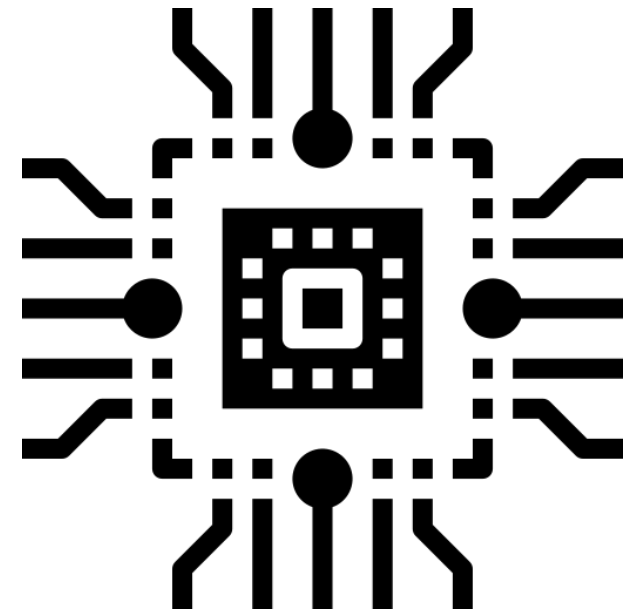


- Develop a Library Manager to unite Open Source EDA tools into a flow
- Provide similar Library Manager functionality to the ones provided by commercial companies
- Design User Interface in a way the users got used to
- Make it easily customizable and extendable based on user requests/preferences
- Integrate documentation generator and static analysis tool for software source trees (Doxygen)
- Provide support for different QT versions (4.8.6 upwards)
- Provide compilation compatibility for Linux and Windows OS

Functionality Goals



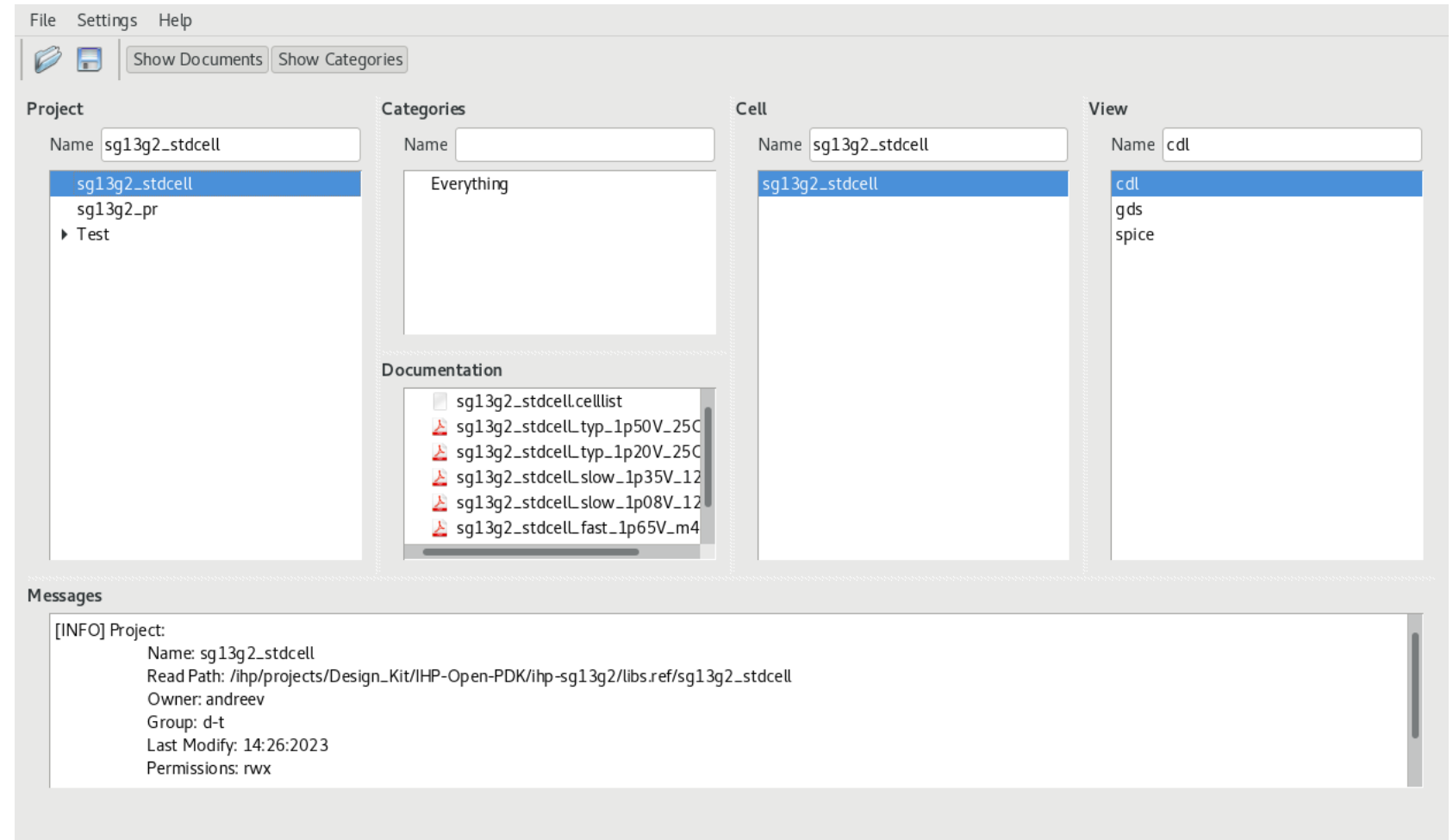
- LibMan helps to organize libraries and cell views in a design project
 - Create new libraries in your directories
 - Copy data into libraries
 - Delete libraries
 - Rename libraries, cells, views
 - Edit libraries, cells, and views
 - Organize cells into categories
 - Unite libraries into groups
 - Display libraries, cells, views information
 - Navigate through libraries, cells, views



Main Interface



- Opens Source IHP PDK data management
- Easy access to design data and documentation
- Launching view related editors
- Customization for user needs



Project File



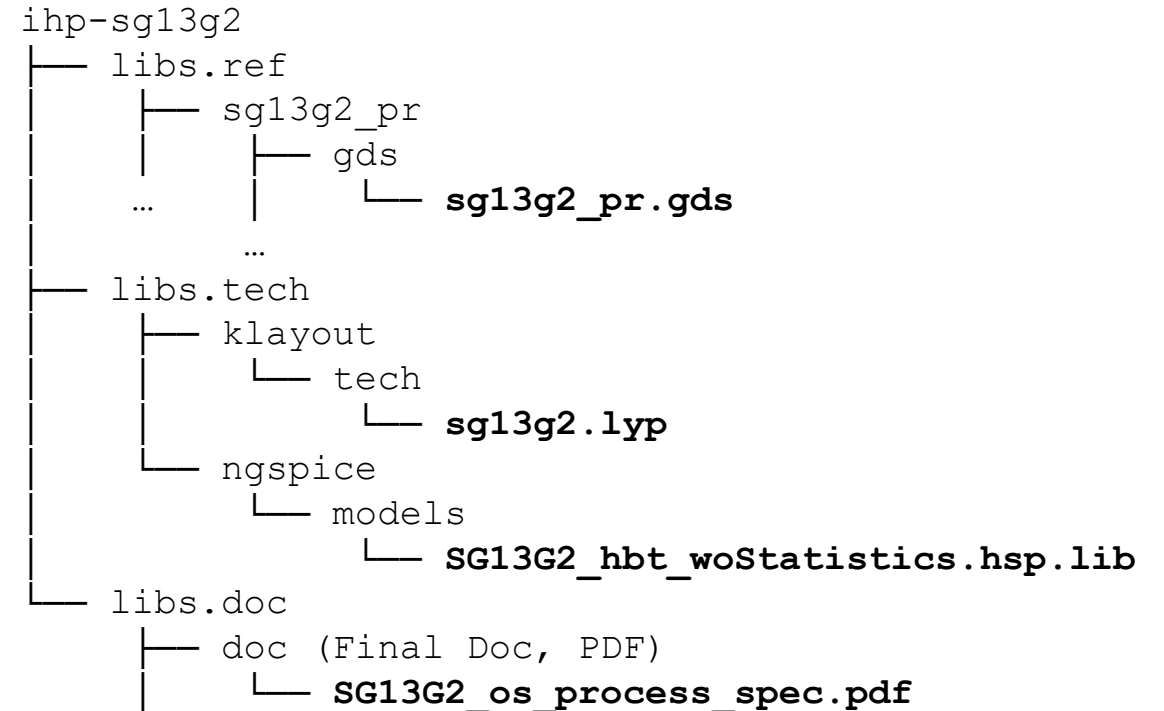
- Project file is used to specify and store the location of the reference and design libraries
- LibMan displays the libraries, categories, cells, and views specified in project file
- Project file can be loaded by
 - Explicitly specifying it as command line argument
 - Running LibMan from a folder where project file is saved
 - Loading from the recent file menu

```
sg13g2.projects - /ihp/projects/Design_Kit/datsuk/src/libman/libman-build-Desktop-Debug/
File Edit Search Preferences Shell Macro Windows Help
GROUP Test testcase2 testcase1
PROJECT sg13g2_pr /ihp/projects/Design_Kit/IHP-Open-PDK/ihp-sg13g2/libs.ref/sg13g2_pr
PROJECT sg13g2_stdcell /ihp/projects/Design_Kit/IHP-Open-PDK/ihp-sg13g2/libs.ref/sg13g2_stdcell
PROJECT testcase1 /home/datsuk/release/testcase1
PROJECT testcase2 /home/datsuk/release/testcase2
```

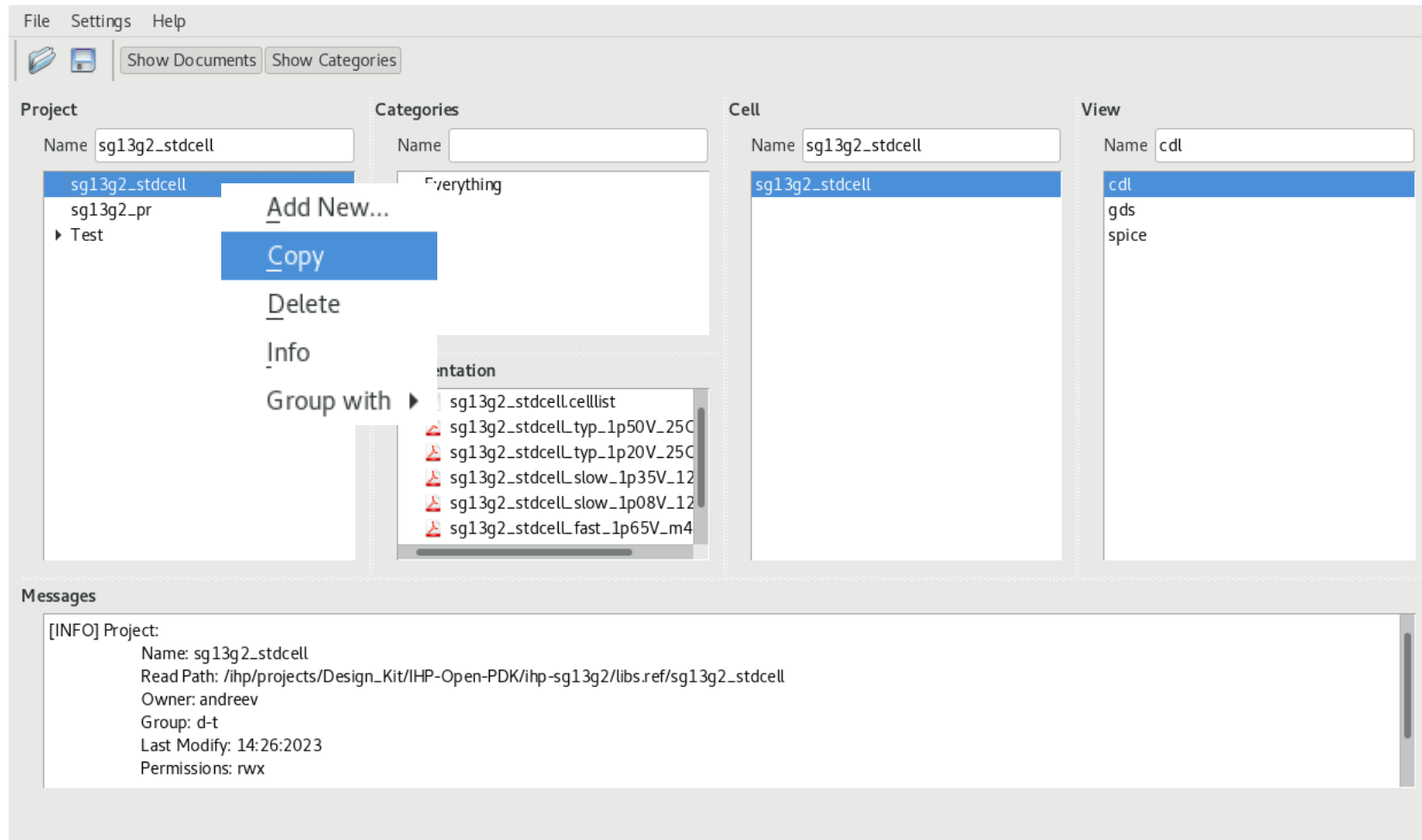
IHP OpenPDK Data Structure



- LibMan expects the following PDK data structure for correct loading of libraries:
 - GDSII view of primitive devices
 - KLayout layer property file
 - HSPICE Models of HBT devices
 - SG13G2 Process specification
- The PDK data can be accessed through LibMan

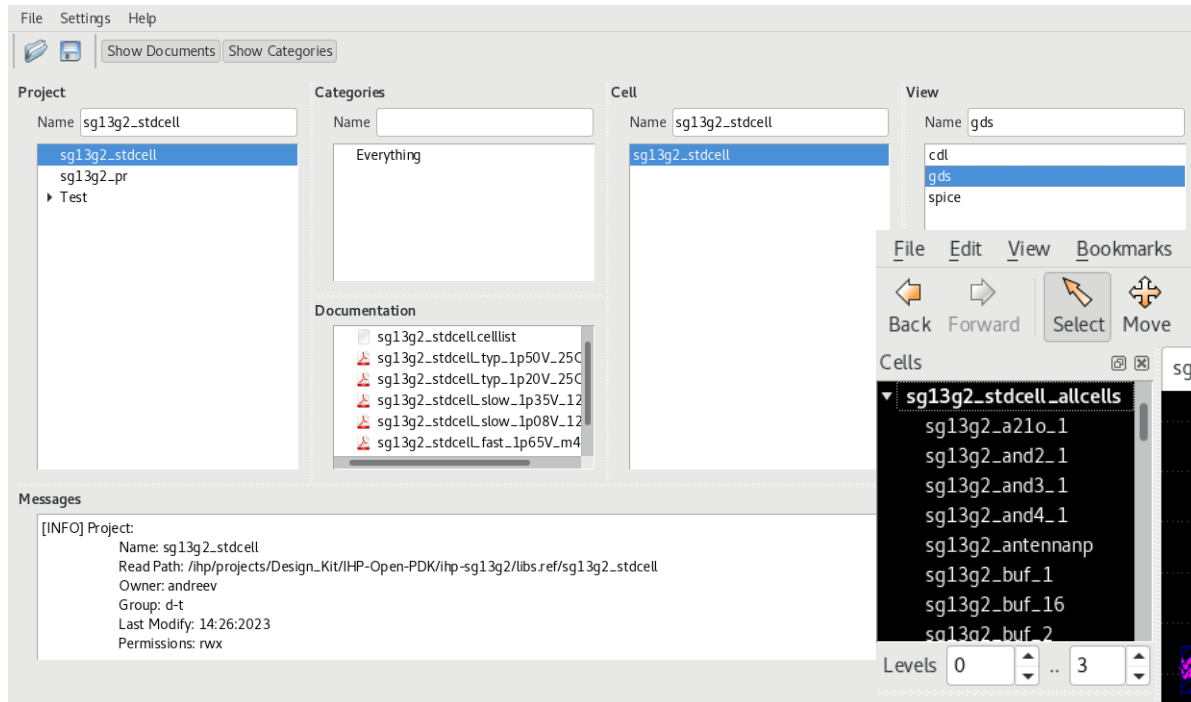


Basic Data Operations

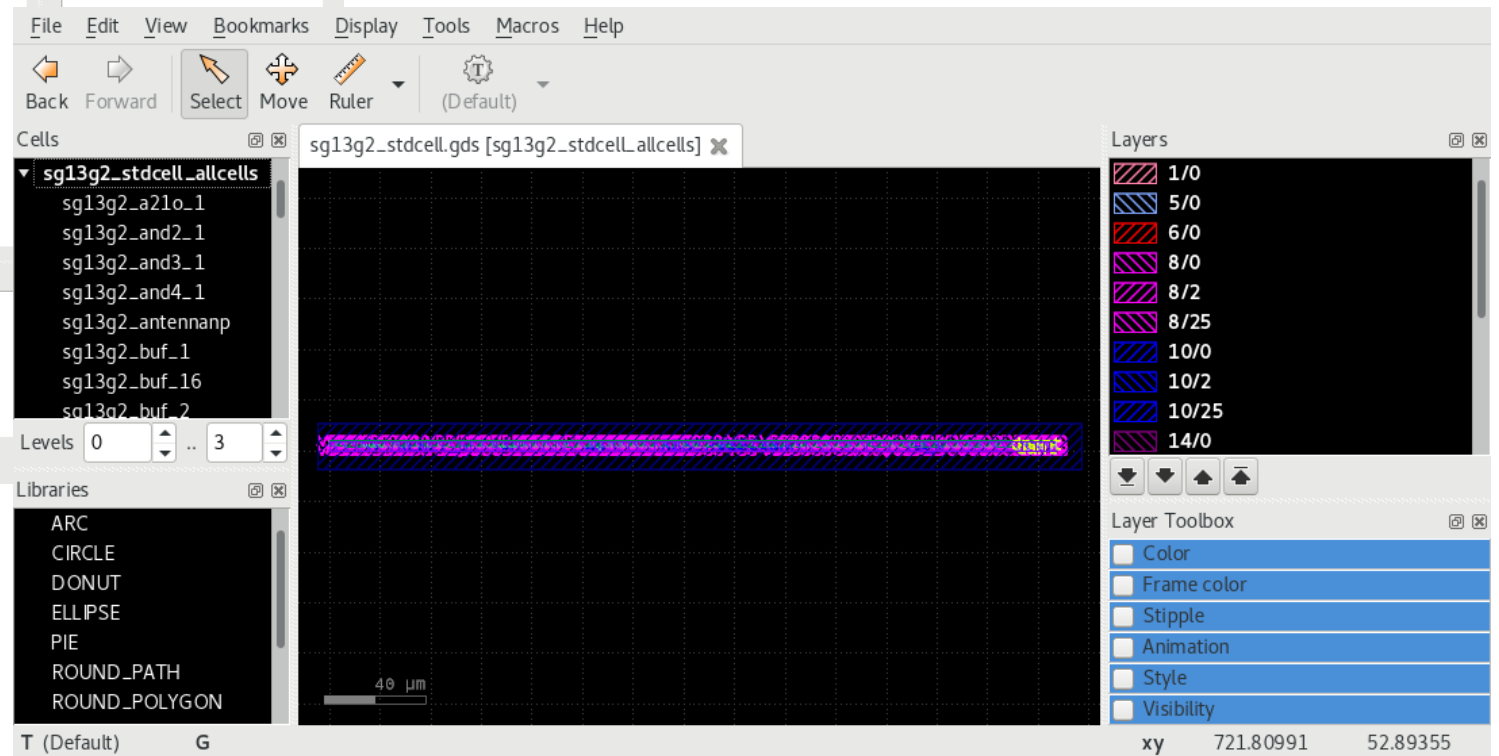


- Unlike commercial tools LibMan allows to perform the following actions directly in main UI:
 - Managing project data files
 - Managing group data (libraries)

Launching View Editors



—○ LibMan executes view editor based on user settings

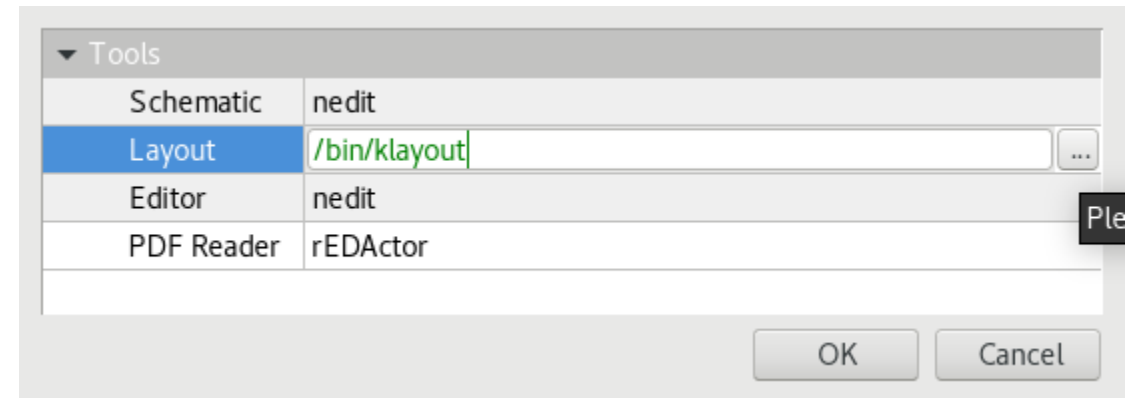
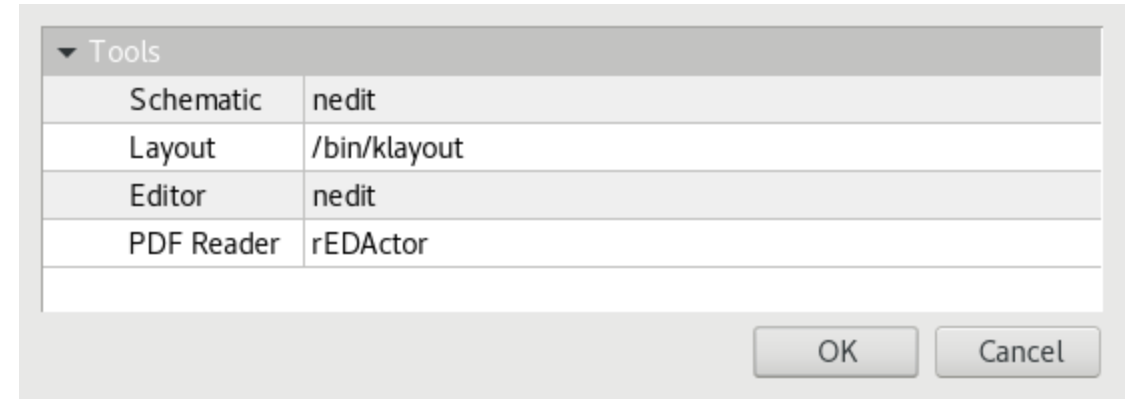


—○ LibMan creates an empty GDS for a new cell

Customization



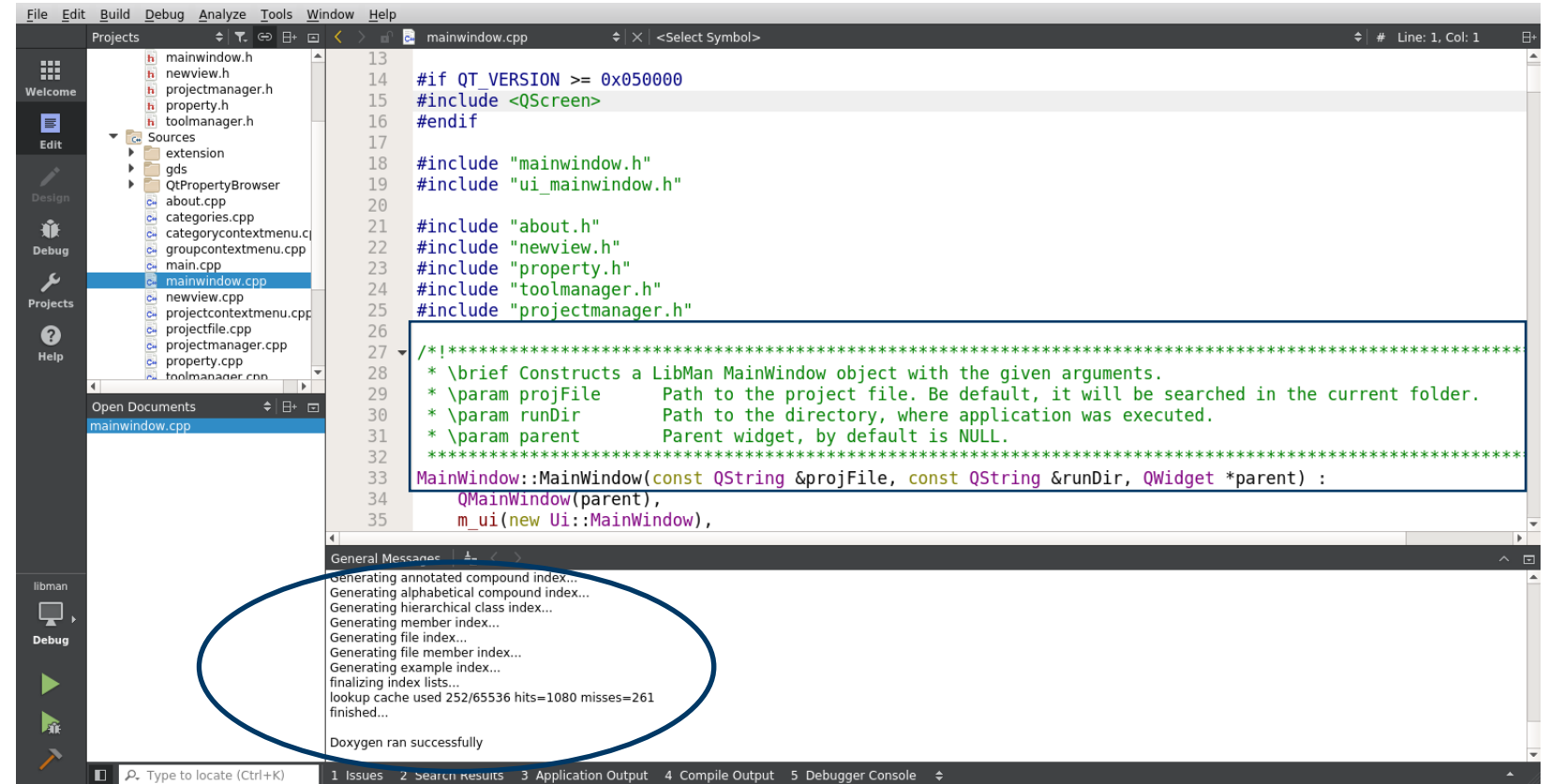
- LibMan allows to let user choosing view editors to be used in the flow
- The default tools:
 - Layout – KLayout (GDS view)
 - Schematic – Nedit (CDL view)
 - Other views – Nedit
 - Documentation – evince (PDF files)



Documentation generator and static analysis tool



- Doxygen was used to parse the source code and generate documentation
 - supports C/C++ natively
 - Can be easily integrated in QtCreator
 - Generates both latex and HTML documentation



LibMan Source Code Documentation



The screenshot displays the LibMan source code documentation interface. The top navigation bar includes 'Main Page', 'Classes', and 'Files'. The 'Classes' section is active, showing a list of class members. The 'Detailed Description' section for the `MainWindow` class is visible, stating: 'The `MainWindow` class is responsible for creating main framework of LibMan and controls all slots and signals.' Below this, the 'Constructor & Destructor Documentation' section shows the constructor signature: `MainWindow::MainWindow (const QString & projFile, const QString & runDir, QWidget * parent = 0)`. The parameters are listed as: `projFile` (Path to the project file), `runDir` (Path to the directory), and `parent` (Parent widget).

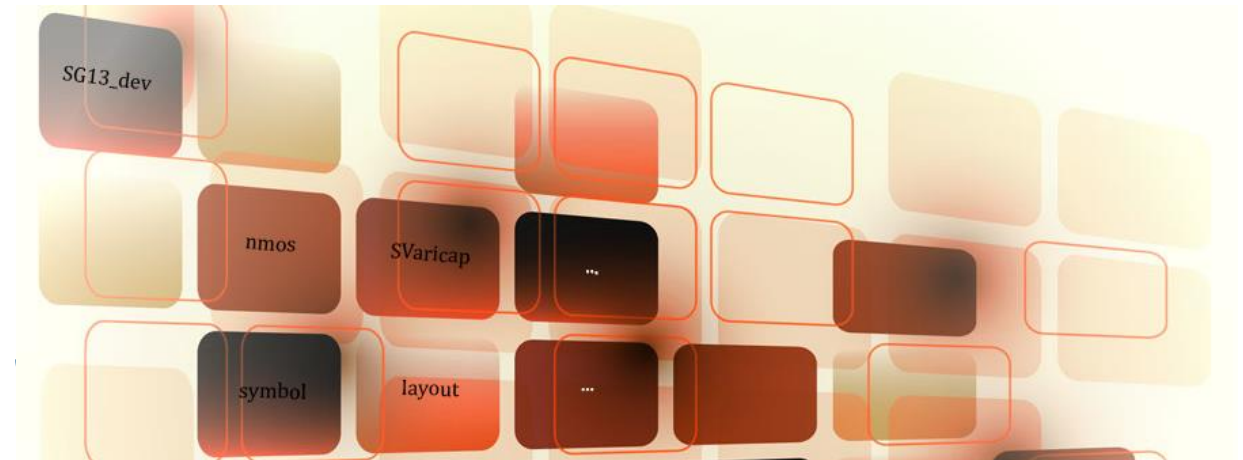
- Easy generation and maintenance of the source code documentation with Doxygen
- All classes, class member and functions have been documented appropriately
- Visualization of relationships between various elements (include dependency graphs, inheritance diagrams, collaboration diagrams etc.)

Road Map



- GitHub submit (July, 23)
- Compilation guide (July, 23)
- Provide Editor Arguments (Q4 2023)
- Git support for design data in LibMan (2024)
- Data base for Open Source EDA?

Open Source EDA Tools



Summary



- The first version of Library Manager has been developed
- Library Manager offers basic functionality to manage project data
- Provided support of IHP OpenPDK data structure
- User is capable to customize views editors
- Doxygen support was applied to the LibMan project
- API documentation for the current source code version was generated
- LibMan has been compiled for QT4.8.6 and QT5.15.1 for both Linux (RedHat, Ubuntu) and Windows OS
- Any feedback and support to make LibMan more design friendly are highly appreciated

Acknowledgment



- Thanks to my colleagues at IHP
- Thanks to FMD QNC (16ME0831)
<https://www.elektronikforschung.de/projekte/fmd-qnc>

SPONSORED BY THE



Federal Ministry
of Education
and Research





Thank you for your attention!

IHP – Leibniz-Institut for High Performance Microelectronics

Im Technologiepark 25

15236 Frankfurt (Oder)

Tel.: +49 (0) 335 5625 133

E-Mail: datsuk@ihp-microelectronics.com

