

Lowering the entry barrier for novel photonic designers

Irene Olivares, Jaime Vilas, Ignacio Jauregui, Antonio Dias

*Alcyon Photonics S.L., Genova-11, 28004, Madrid-Spain
Antonio.dias@alcyonphotonics.com*

Abstract: Through our open software package and a guided process flow, barriers to entry are lowered so that a student can focus on photonic circuits design from the very beginning. © 2021 The Author(s)

1. Introduction

The integrated photonics model, although almost thirty years apart, follows in the footsteps of the electronics paradigm. As a consequence of a younger and less standardized model, circuit designers are expected to have a highly technical background with knowledge of a wide variety of fields, such as light-material interaction, programming, simulation, and optoelectronics, often acquired after a doctoral program.

In recent years, bachelor's and master's programs have started including training in simulations of photonic integrated circuits (PICs). However, software tools are still not as accessible as those used in the electronics field. Currently, there are several commercial software tools, each being improved by the day, that enable the entire design flow: From component design to PIC layout. However, these software packages are not fully compatible (if at all) between them, and their licenses are expensive, which frequently causes companies and research institutions to purchase the licenses of a single developer. This, in turn, makes it difficult for students or novel designers to manage their designs and the data collected during their training periods and careers when they lose access to said software licenses. Furthermore, having different tools is also a barrier to collaboration between institutions, even when they develop similar technologies.

At Alcyon Photonics, we provide a proprietary but open access software (python based) package to Master and PhD students with a programming environment to make it easier for the student to focus on designing PICs from the very beginning. This way, they can create, store, manage and share their designs even if they do not have access to said software licenses. Although there are some open tools, adapting them to solve photonic circuits requires arduous work and a great deal of scripting, which may frustrate the student's motivation and expectations. In this scenario, the Alcyon's PDK has the mission of lowering the first barrier of entry for new photonic circuit designers. Lowering the entry barrier means making high-performance designs available to users while providing an open tool for their daily work on creating and managing block designs and photonic circuits. The Alcyon's PDK does not provide proprietary simulation tools but facilitates the interoperability with powerful commercial software.

AlcyonPDK is a user-friendly version of a more advanced version created specifically for Alcyon Photonics' internal workflow. It was born out of the need to manage a large number of designs and to streamline the collaborative work between different designers. With those packages, all group personnel (including students) can share and use component libraries and circuit architectures, regardless of the version they have. This way, the student will be able to keep and access all the work done during their stay in the company, without compromising the rest of the IP of the company and without the need for software licenses to visualize, modify or share their designs.

2. Package content

The PDK is already available at the Alcyon's website [1] and includes **two libraries populated with Alcyon designs**. One library with basic geometries useful to build more complex structures or as connectors. A second library with SWG-based proprietary designs whose IP is protected by black boxes. New components and more complex architectures (receivers, sensors, polarization controllers...) will be added progressively. Those components are provided together with **compact models** that describe the frequency response of the components with data from both simulation and experiments to enable more realistic PIC simulations. The compact models are compatible with well-known commercial simulation tools.

The AlcyonPDK also provides tools for users just by following the AlcyonPDK programming rules, which are deeply documented:

- **Creation of new libraries** in Klayout© [2] through a GUI.

Sixteenth Conference on Education and Training in Optics and Photonics: ETOP 2021,
edited by A. Danner, A. Poulin-Girard, N. Wong, Proc. of SPIE Vol. 12297, 122971Y
© 2022 SPIE · 0277-786X · doi: 10.1117/12.2635571

- **Creation of new components** by creating Drawer classes, which are Python classes that contain the methods that build the geometries, contact points and layer specifications that conform the component.
- **Adding components to specific libraries.** Registration scripts will be automatically generated by Alcyon's Technology Generation Tool in such a way that the components become available through Klayout libraries.
- **Circuit layout.** Creating complex layouts in Klayout© by scripting can be an arduous task. However, the **Layout-Creator module** provides classes and functionalities that make this process much easier and intuitive, allowing circuits to be laid out either through scripting or by taking components from different libraries.
- **Creation of behavioural models.** Scripts that convert dataframes with S parameters into file formats compatible with some of the main commercial tools for PIC simulation.

3. An example: creating a photonic circuit

Imagine a novel designer wants to design an MZI. Said designer could create each of the components and register them in Klayout libraries using the classes and methods already defined within the AlcyonPDK or, in this case, use the components from the libraries already available withing the AlcyonPDK:

As shown in Fig. 1.a, the available components are displayed in drop-down libraries, whose properties are described in the documentation. Each of the library components is defined by a Python file with the same name as the class that defines it, for example, a waveguide with an arbitrary number of curves and straight lines is built from the *PipeDrawer* class in the PipeDrawer.py file (see Fig.1.b). To build the MZI, the designer just needs to import the classes of the elements available in the libraries and connect them using a node- and branch-based approach. In Fig. 1.c, it is shown how through a few lines of code the MZI can be built and added to a Klayout design by indicating the netlist of pins to be connected. In addition, Fig. 1a. demonstrates that gratings from other PDKs can be also added through scripting.

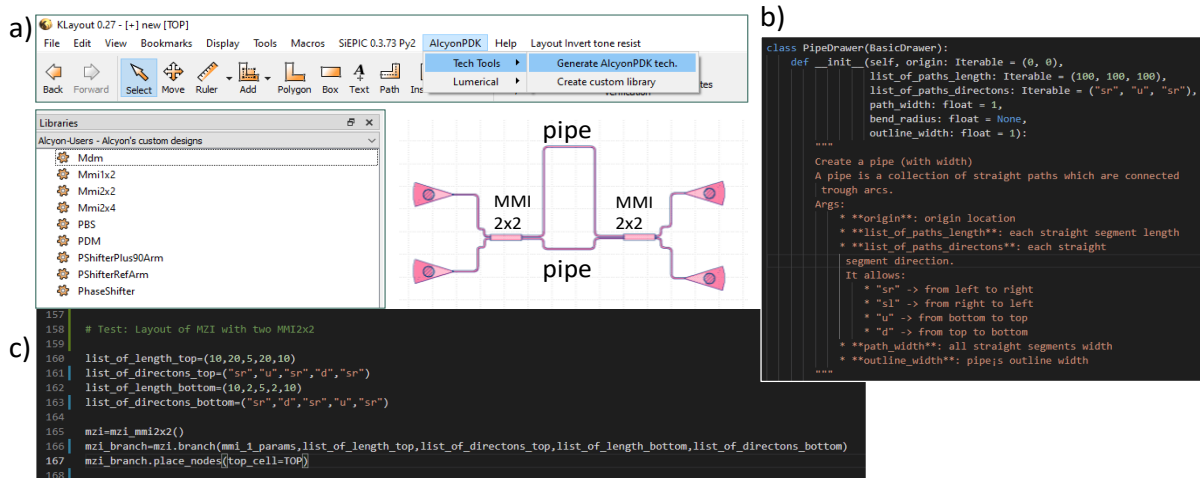


Fig. 1. Screen capture of: a) Klayout with the AlcyonPDK tools and libraries. b) Code excerpt taken from the PipeDrawer class that defines the arms of the MZI. c) Code excerpt that builds the MZI by using MMI2x2 and pipes from the Alcyon libraries.

In conclusion, the software package together with a guided process flow allows students to get started in photonic design in a more efficient way.

3. References

- [1] Alcyon Photonics Website. <https://alcyonphotonics.com> (Last visited: 07/06/2021)
- [2] "Klayout". <http://www.klayout.de> (Last visited: 07/06/2021)