

MATLAB COM Integration In SAW/BAW Modeling

Dr. Alexander Rukhlenko, SAW/BAW Consultant

1. Introduction

Widely used by scientists and engineers for numerical analysis, modeling, and algorithm development, the computing environment [MATLAB®](#) (Mathworks) can be integrated with external C/C++ and Fortran applications through [MATLAB COM \(Component Object Model\) Integration](#) and mixed-language programming techniques. This enables communication and data exchange between MATLAB models and external software applications through COM Automation technology.

This mixed-language MATLAB COM approach improves computational efficiency in many practical cases and extends the capabilities of SAW/BAW models. It also provides access to external libraries and software components that are not directly available in MATLAB.

Therefore, MATLAB mixed language COM programming provides a powerful technique for integrating MATLAB-based SAW and BAW models with external software applications and simulation environments. For example, Dr. Alexander Rukhlenko successfully applied this approach to integrate SAW and BAW MATLAB models with commercial electronic circuit simulators such as Keysight® Pathwave Advanced Design System (ADS).

2. Mixed Language Programming

The basic idea is straightforward: the user calls custom C/C++ or Fortran functions from MATLAB as if they were built-in functions. Developers implement these functions as MEX functions, and MATLAB executes them directly by their file names.

However, MEX functions are not suitable for all applications, since MATLAB is

a high-productivity environment designed to reduce low-level programming and improve computational efficiency. Nevertheless, in many cases, MEX programming in C/C++ or Fortran can significantly reduce computational time compared to pure MATLAB code. This is especially useful when nested loops are unavoidable or when vectorization of SAW/BAW models is not efficient.

To create MEX functions, developers write gateway and computational routines in C/C++ or Fortran and compile them using the MATLAB-neutral Application Programming Interface (API). These API functions enable data transfer between MEX functions and the MATLAB workspace.

3. C/C++ MATLAB API Types for MATLAB COM Integration

MATLAB provides two API types for external C/C++ applications: the MATLAB C++ Data API and the MATLAB C Matrix API.

The MATLAB C++ Data API supports modern C++ features, semantics, and design patterns. Developers who prefer C language can use the simpler MATLAB C Matrix API.

The author implemented both MATLAB interfaces to call SAW/BAW models from C/C++ applications, particularly as ADS User-Compiled Models (UCMs).

4. SAW/BAW Model MATLAB COM Integration

In addition to mixed-language programming, the author developed and implemented an alternative approach for interfacing MATLAB SAW/BAW models with external C/C++ applications. This approach is based on MATLAB COM Integration. In general, COM stands for Microsoft[®] Component Object Model, a framework for integrating reusable software components into external applications.

This implementation requires MATLAB COM Automation. It allows one application (the client, or controller) to control COM objects exported by another application (the server). In this setup, MATLAB acts as the server and provides modeling functionality through COM objects. As a rule, the MATLAB server runs as a standalone application.

5. COM Data Exchange with SAW/BAW Models

Communication between the client and the server relies on COM Automation technology, which provides a structured data exchange infrastructure. As a result, COM automation clients (C/C++ applications acting as controllers) can access and manipulate COM objects exposed by Automation COM servers.

In this architecture, the MATLAB COM server represents a set of MATLAB functions implementing SAW/BAW models.

Practical system-level MATLAB integration with external simulation tools such as ADS requires an additional communication layer for data exchange. For this purpose, Dr. Alexander Rukhlenko proposed using standard network-based data transfer mechanisms. In particular, he implemented a generic TCP/IP interface that combines the Transmission Control Protocol (TCP) with the Internet Protocol (IP). This approach provides a standard communication channel for MATLAB-based data exchange. As a result, COM-based components can exchange numerical input and output data efficiently.

MATLAB supports TCP/IP communication through network sockets, which allow an external application (TCP/IP client) to connect to a MATLAB server. In general, the client sends input parameters in binary or ASCII format and receives computed data from MATLAB. In SAW/BAW modeling applications, the client (ADS) transmits SAW/BAW model parameters to the MATLAB server via the TCP/IP interface and receives the corresponding simulation results calculated by MATLAB SAW/BAW model functions.

Overall, this combined MATLAB COM and TCP/IP-based architecture provides a flexible and scalable framework for integrating MATLAB-based SAW/BAW models with external simulation environments.

6. MATLAB COM Compilation and Software Upgrade

The MATLAB COM approach with a TCP/IP interface enables efficient data exchange between a client application (ADS) and a MATLAB server running SAW/BAW models during simulation. Although this method is more complex to implement, it offers several important advantages.

COM components and applications remain independent and can be compiled

separately. As a result, software updates become easier because developers only need to replace modified components without recompiling the entire system. In addition, component location remains transparent to the application. Users can move components to another process or even to a remote system

Furthermore, the ADS client, including User-Compiled Models (UCMs) calling MATLAB SAW/BAW models, and the MATLAB server remain isolated because they do not share DLLs through compilation and linking. Consequently, this modeling architecture is largely version-independent and eliminates potential compatibility issues between ADS and MATLAB releases.

In summary, the proposed MATLAB COM TCP/IP approach provides an efficient and practical solution for integrating MATLAB SAW/BAW models with external simulation and measurement software.

Web version: [MATLAB COM Integration In SAW/BAW Modeling](#)