



Course Title:

MATLAB to C with MATLAB Coder

Course Purpose

This two-day course focuses on generating C code from MATLAB code using the MATLAB Coder product. The focus is on developing MATLAB code that is ready for code generation, generating C code that meets optimization requirements, and integrating generated code into parent projects and external modules. This course is intended for intermediate to advanced MATLAB users.

Pre-requisites

MATLAB® Fundamentals course and knowledge of C programming language



- ✓ 2 training days
- ✓ Hours: 09:00-17:00
- ✓ Total training hours: 16

Teaching method

The course combines lectures, demonstrations and practical exercises in MATLAB, using original training books from MathWorks. The course is in Hebrew but the training materials are in English.

עמוד מס' 1

Training Center Systematics - Contact information:

Phone number: 03-7660111 Ext: 5 **Email:** training@systematics.co.il

Website: <http://www.systematics.co.il/mathworks>



Course Objective:

Code Generation with MATLAB Coder

Objective: Become familiar with MATLAB Coder and its applications.

- MATLAB Coder overview
- Workflow for generating C code from MATLAB code
- Setting up your C compiler
- Generating C code
- Navigating generated code
- Generated code modules

Preparing MATLAB Code for Code Generation

Objective: Use MATLAB Coder coding standards to write MATLAB code that is ready for code generation.

- Translating MATLAB code into C code
- Calling unsupported MATLAB functions
- Preparing existing MATLAB code
- Code preparation workflows

Working with Fixed-Size Data

Objective: Generate C code from MATLAB code that has fixed-size or constant inputs.

- Data characteristics overview
- Specifying fixed-size top-level inputs
- Specifying constant top-level inputs

Working with Variable-Size Data

Objective: Generate C code from MATLAB code that has variable-size inputs or local data.

- Specifying variable-size top-level inputs
- Specifying variable-size local data
- Reusing variable

עמוד מס' 2

Training Center Systematics - Contact information:

Phone number: 03-7660111 Ext: 5 **Email:** training@systematics.co.il

Website: <http://www.systematics.co.il/mathworks>



Working with Global Data and Structures

Objective: Generate C code from MATLAB code that contains persistent data, global variables, or input structures.

- Persistent variables
- Global variables
- Working with structures
- Passing variables by reference

Integrating with External Code

Objective: Integrate generated C code from MATLAB Coder with external C code.

- Code integration overview
- Entry points to generated code
- Integrating external C code using MATLAB Coder interface
- Integrating external C code using an external IDE
- Calling external C routines
- Integrating with a MATLAB wrapper function

Optimizing Generated Code

Objective: Use various options and techniques to optimize generated code.

- Specifying variable numbers of arguments
- Loop unrolling in generated code
- Memory preallocation without initialization
- Function inlining
- File partitioning
- Configuration objects
- Naming conventions in generated code
- Optimizing for computational complexity

Appendix A: Fixed Point Design

- Fixed-Point Designer™ overview
- Fixed-point workflow – manual conversion
- Preparing for fixed-point conversion
- Converting to fixed point
- Fixed-point workflow – automated conversion
- Code coverage
- Workflow comparison

עמוד מס' 3

Training Center Systematics - Contact information:

Phone number: 03-7660111 Ext: 5 **Email:** training@systematics.co.il

Website: <http://www.systematics.co.il/mathworks>