MathWorks

MATLAB&SIMULINK

Design and Verify Embedded Signal Processing Systems Using MATLAB and Simulink

Giorgia Zucchelli, Application Engineer, MathWorks

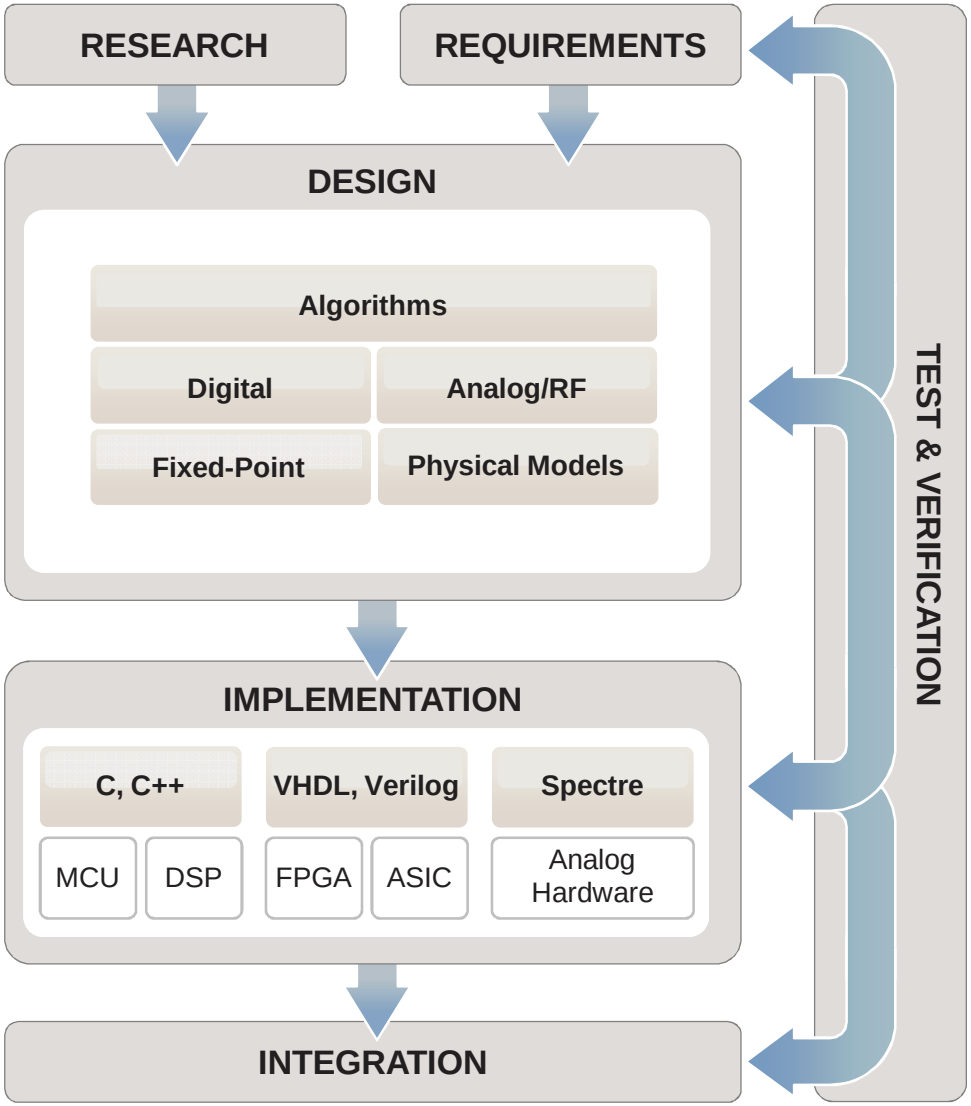
17 December 2010, Technical University Eindhoven

Introduction to Model Based Design

- Methodology to design complex systems
  - Using models and simulation
  - Using tools that help automation
- Find errors early
- Reduce costly prototypes
- Increase productivity

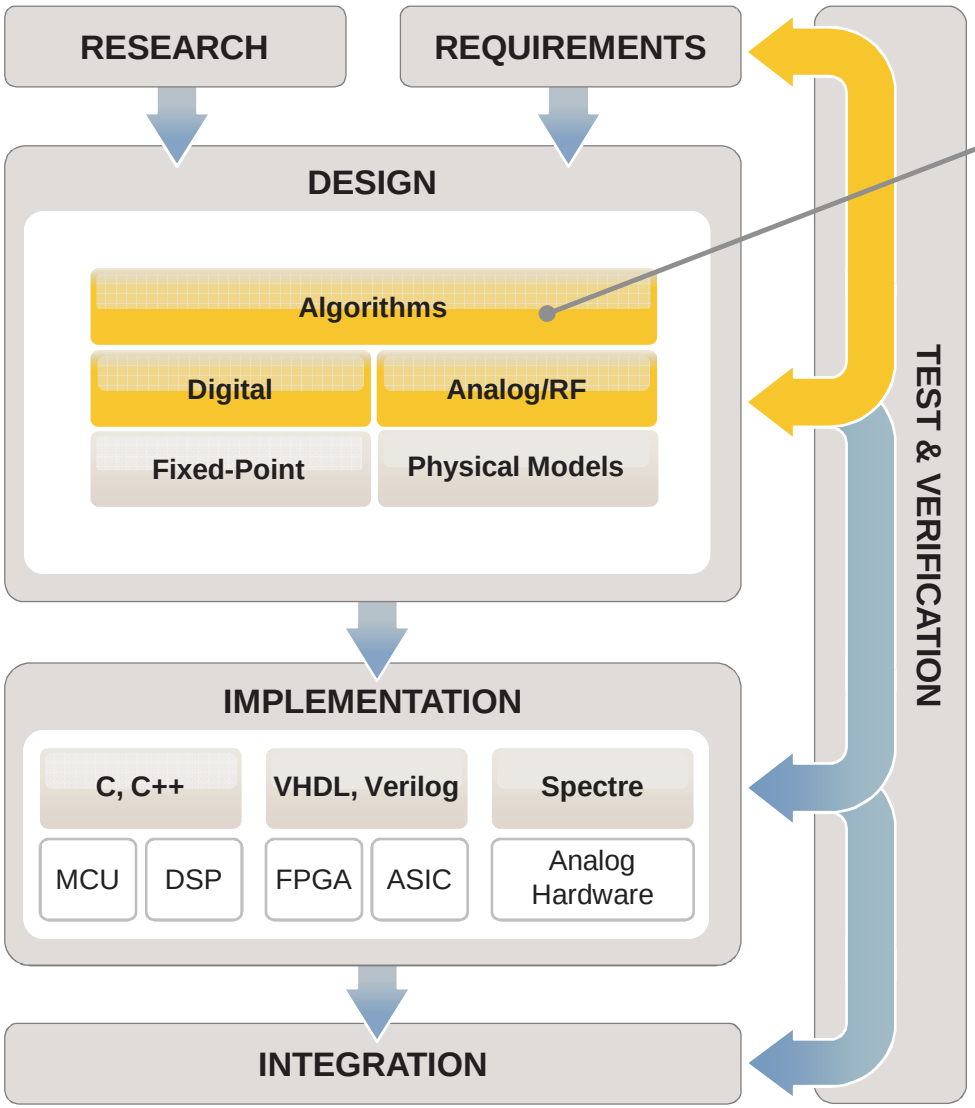
From Idea to Implementation

Not only coding, but also verifying, debugging, documenting, reusing



- Design Flows
- Design Methodologies
- EDA Tools
- Languages
- Simulators
- Models
- Prototypes
- …

Algorithm Design

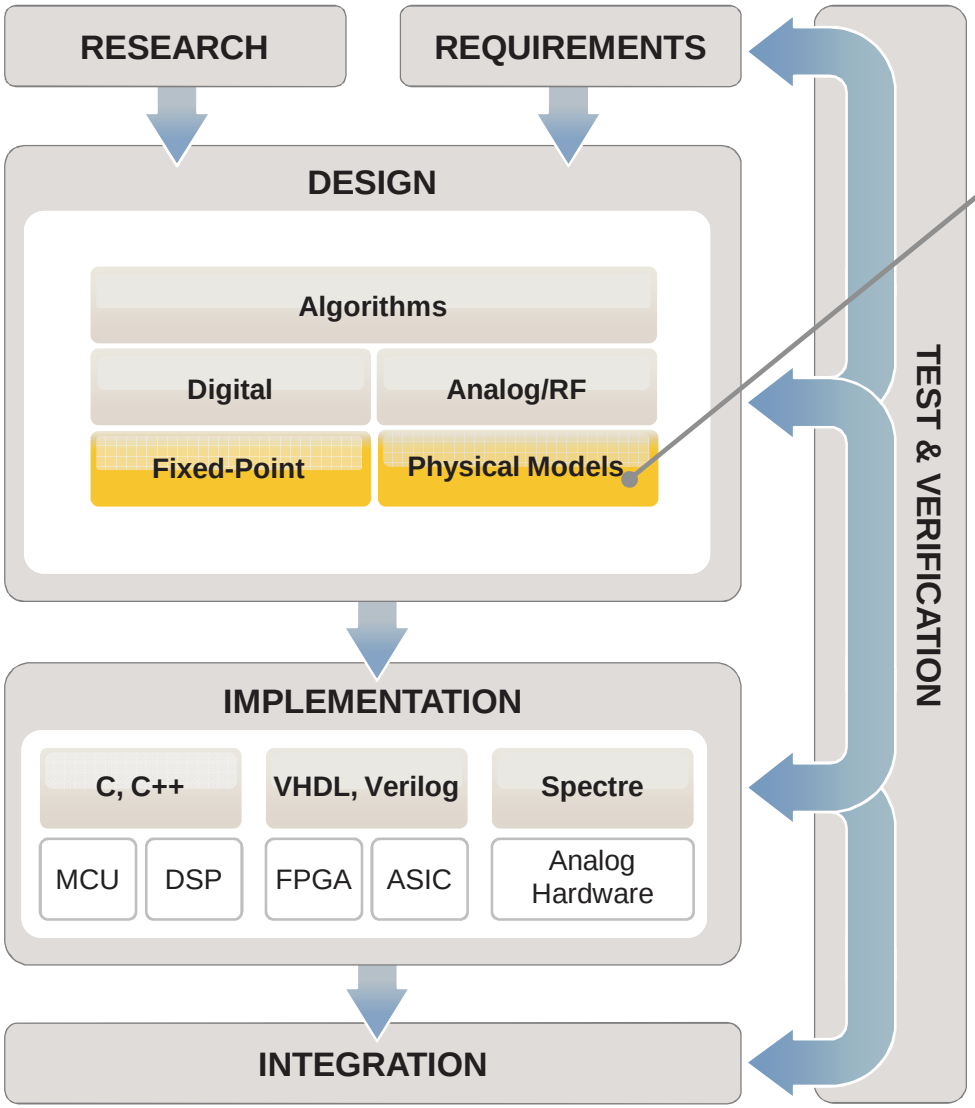


- Executable specs
- Multi-domain: one model
- Many trusted functions

Advantages:

- **Share the same specs**
- **Reuse the same model**
- **Fast design iterations**

Algorithm Refinement

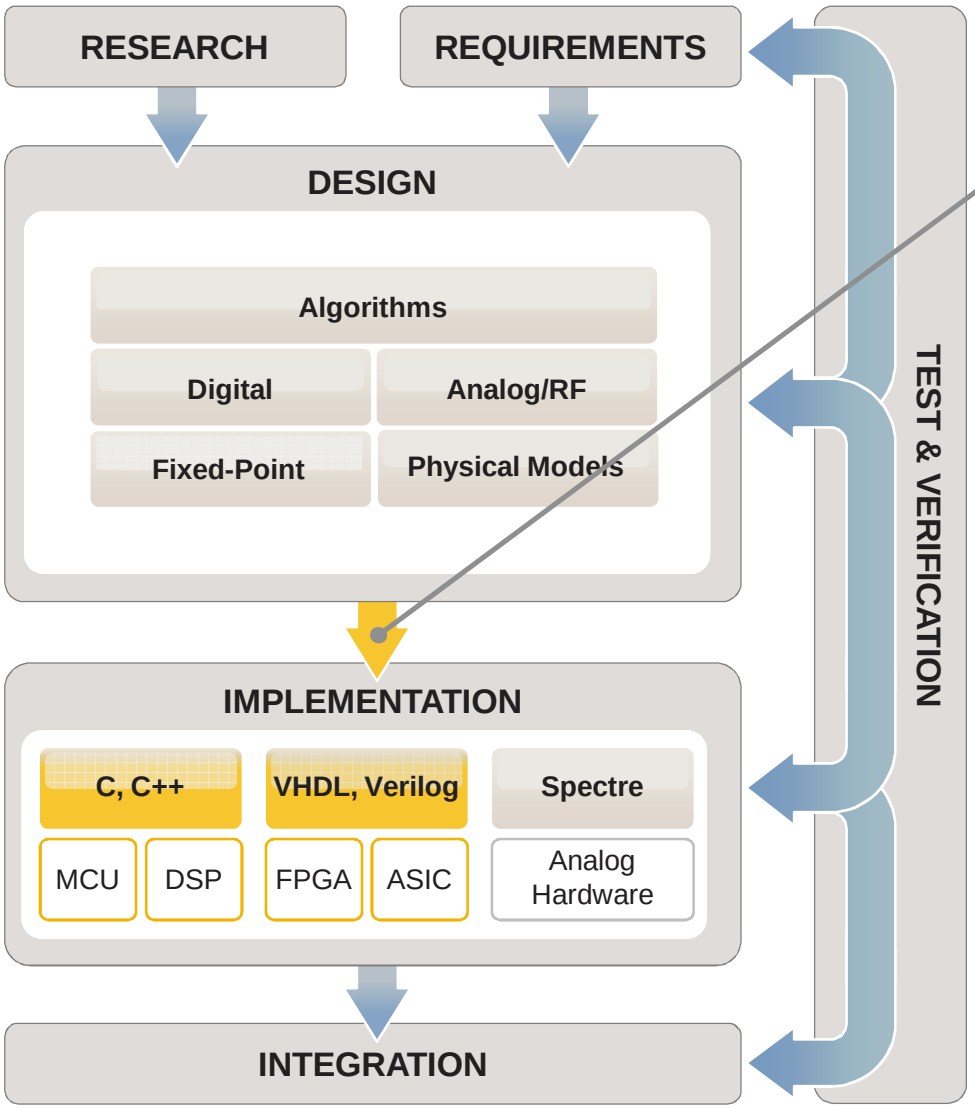


- Refined behavioral models
- Bit-true simulation
- Multi-domain physics

Advantages:

- **Anticipate implementation**
- **Early verification**

Algorithm Implementation

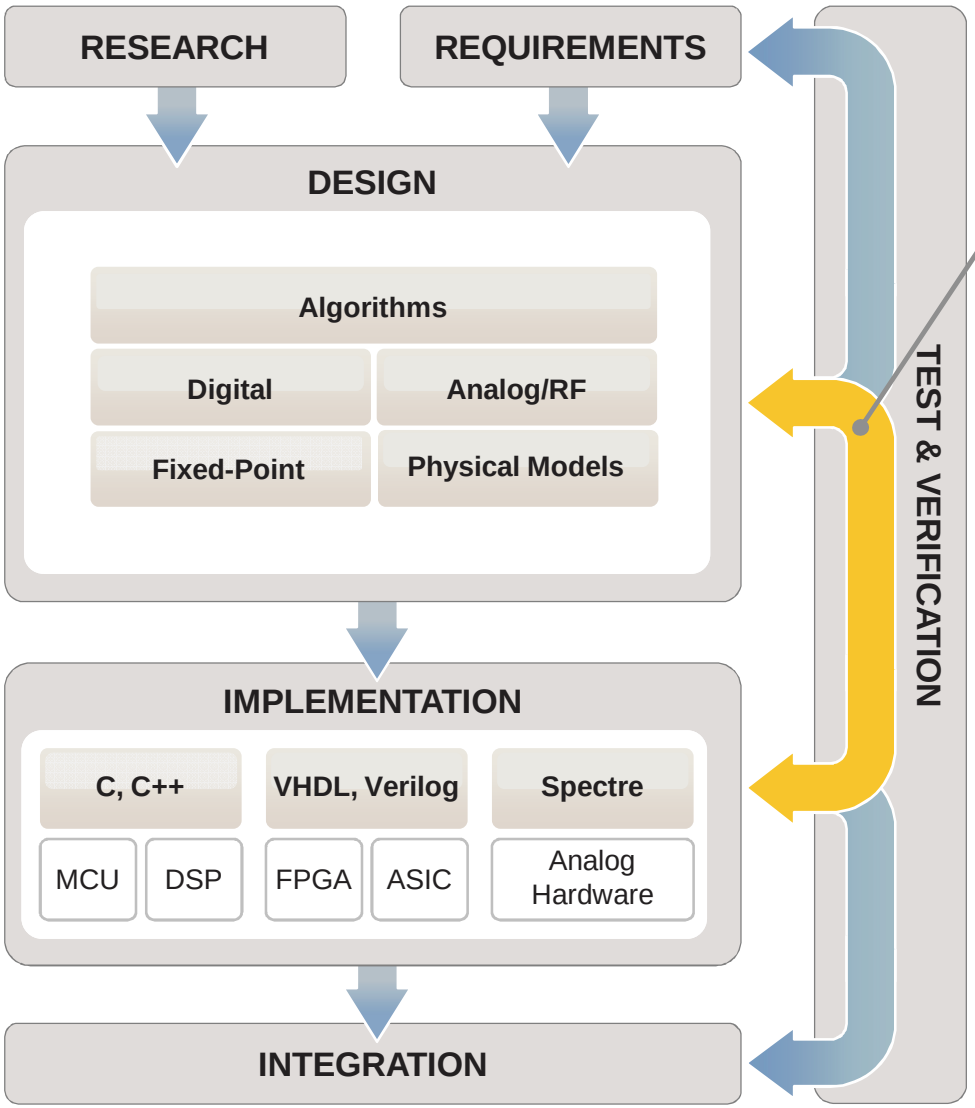


- Automatic code generation
- C / C++
- Synthesizable HDL

Advantages:

- **Rapid prototyping**
- **Less debugging, better design**

Algorithm Verification

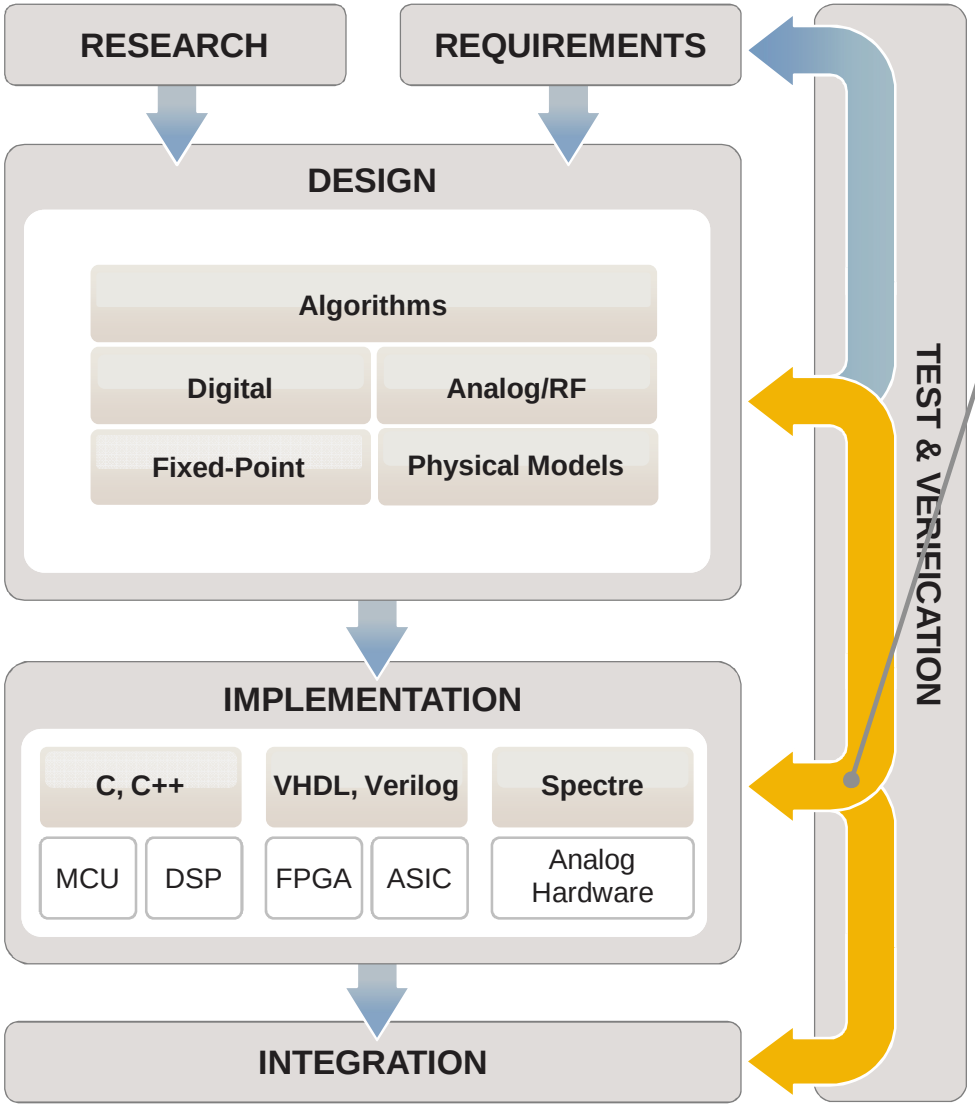


- Co-simulation with EDA tools
- Integration with IDEs

Advantages:

- **Verify implementation at system-level**
- **Design reuse**

Testing



- "Hardware in the loop" verification
- Test at system-level

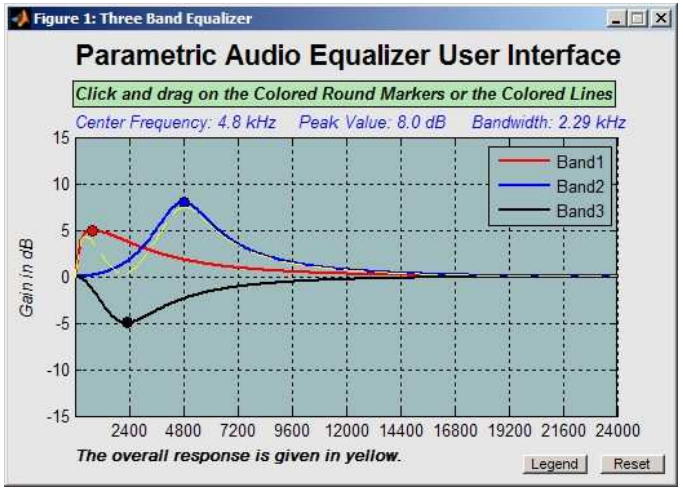
Advantages:

- **Unambiguous, fast verification**
- **One testbench fits all**

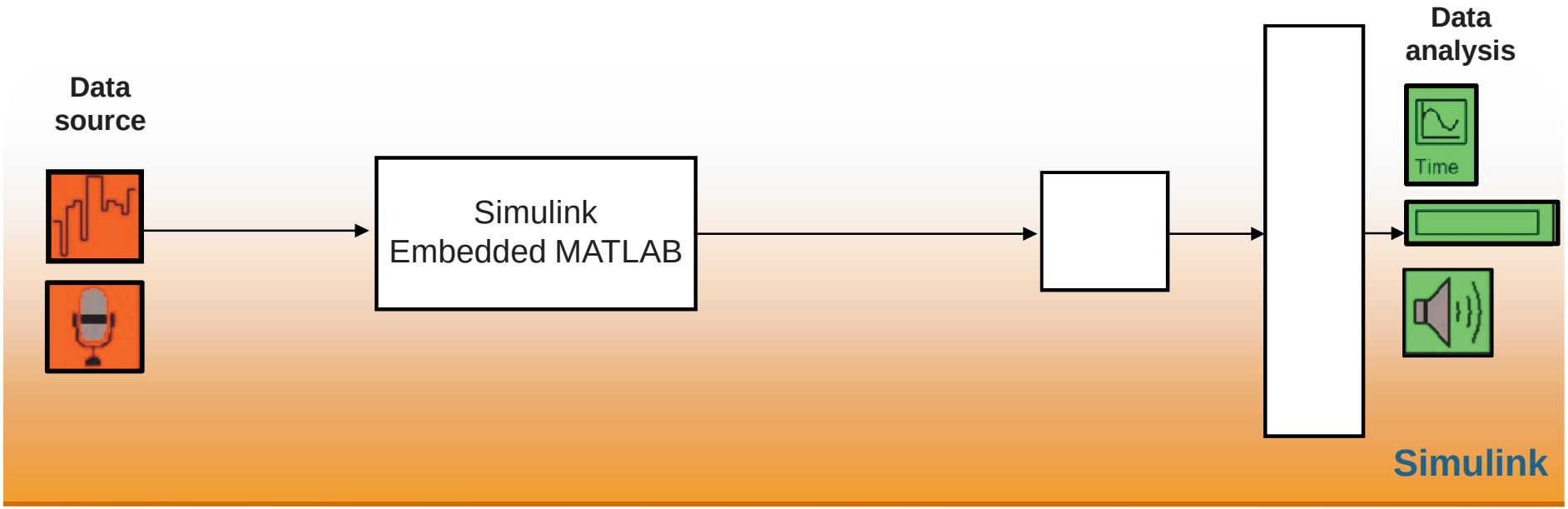
Demo: Parametric Audio Equalizer

Digital filters used to adjust the frequency content of an audio signal

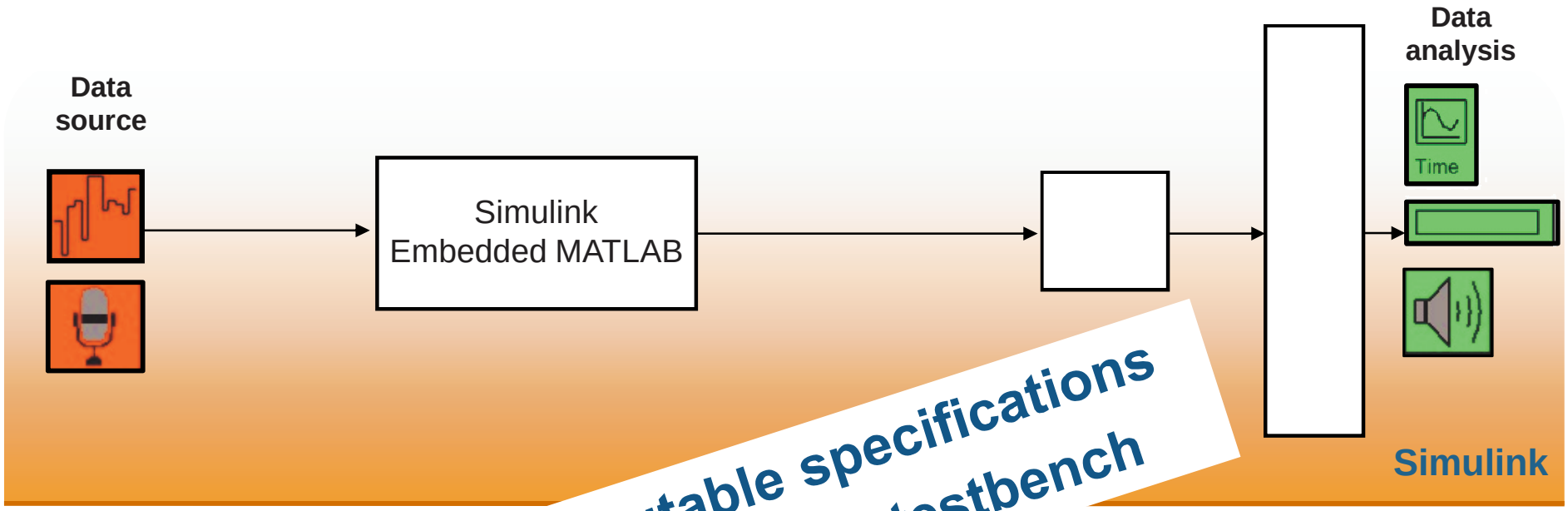
- Parametric response that can be run-time controlled
- Three band equalizer
  - Low Band: 60 to 1500 Hz
  - Mid Range: 1200 to 4800 Hz
  - High Range: 4800 to 12 kHz
  - Amplitude range: -8 to +8 dB
- Target processor: TI C6437 DSP



Algorithm Design: PC Based Prototyping

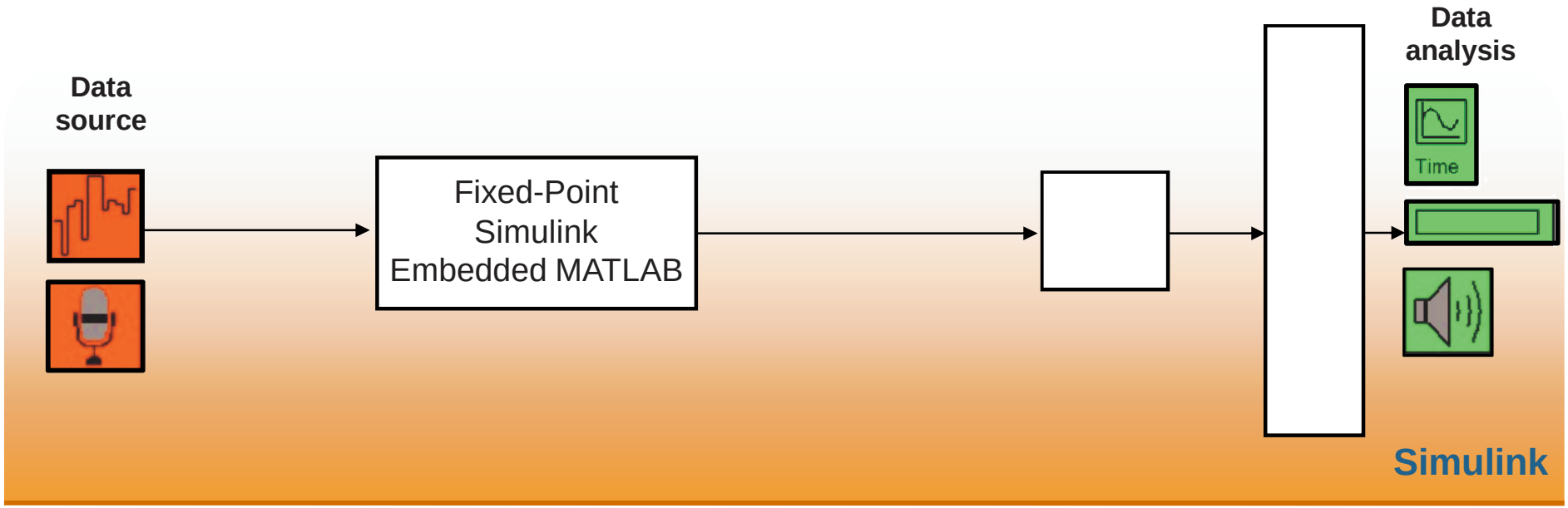


Algorithm Design: PC Based Prototyping

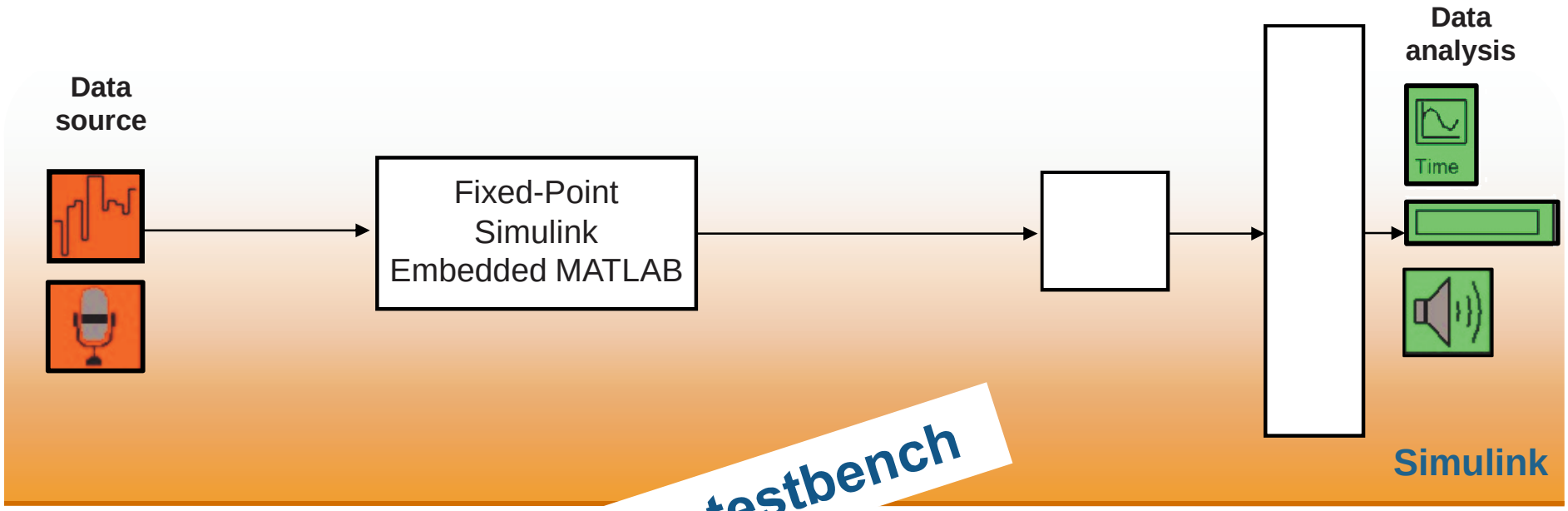


- Implement the executable specifications
- Separate the model from the testbench

Algorithm Refinement: Fixed-Point

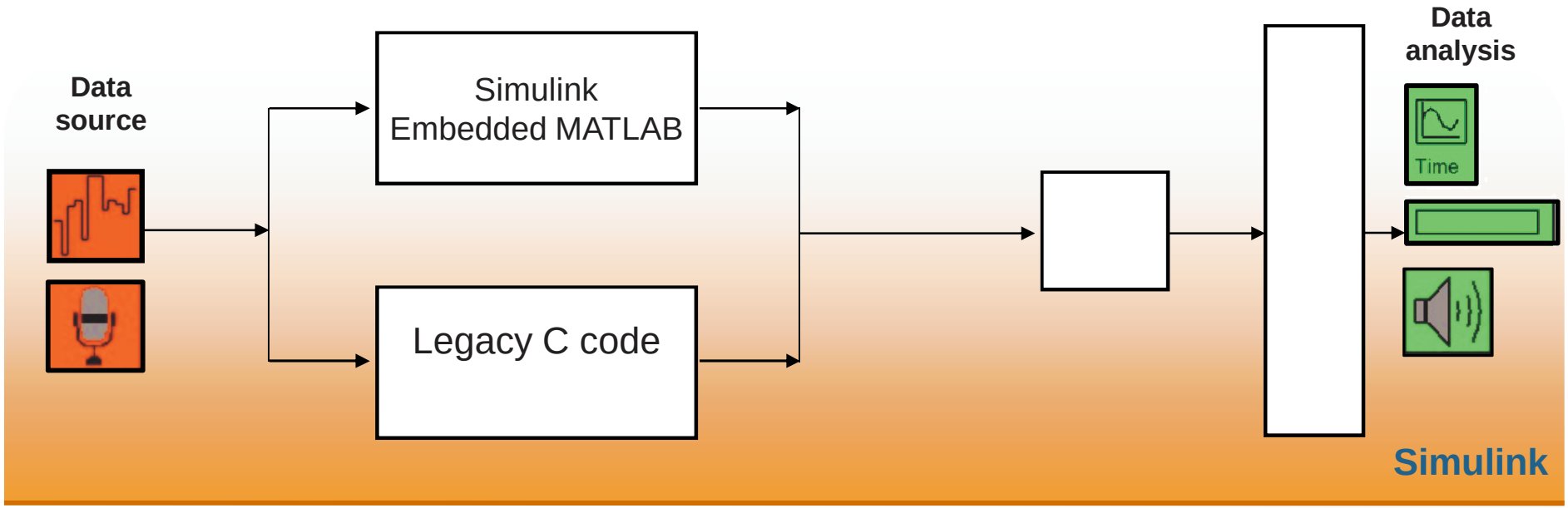


Algorithm Refinement: Fixed-Point

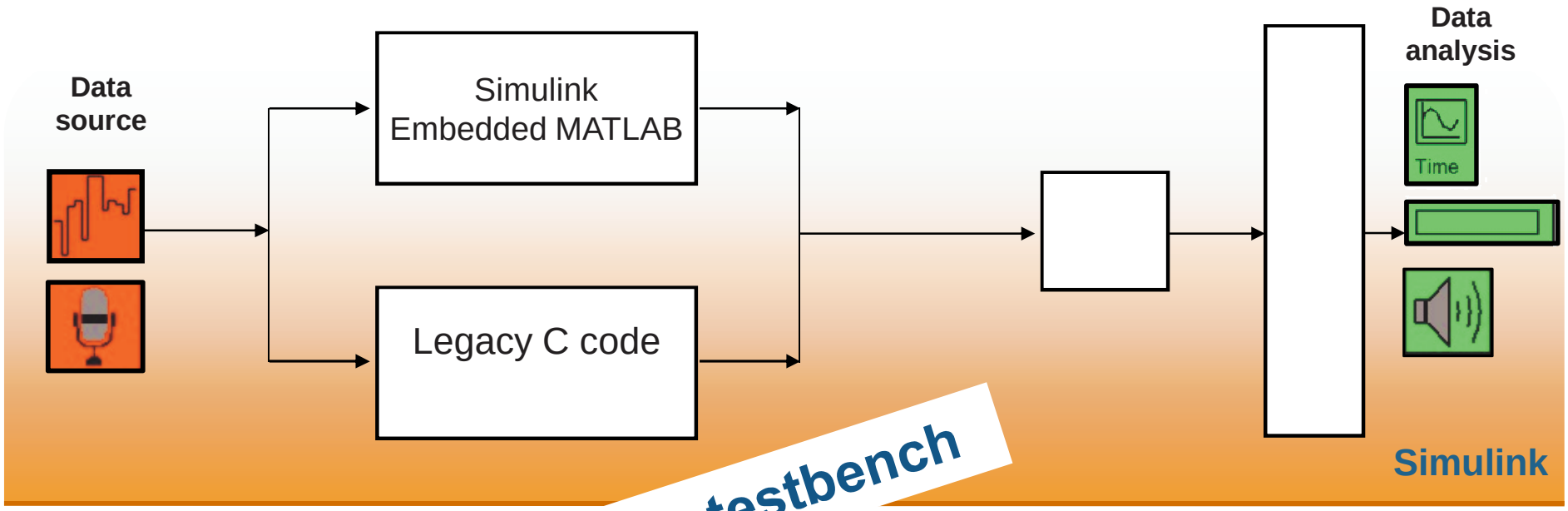


- Reuse the same testbench

Software-in-the-Loop (SIL) Verification

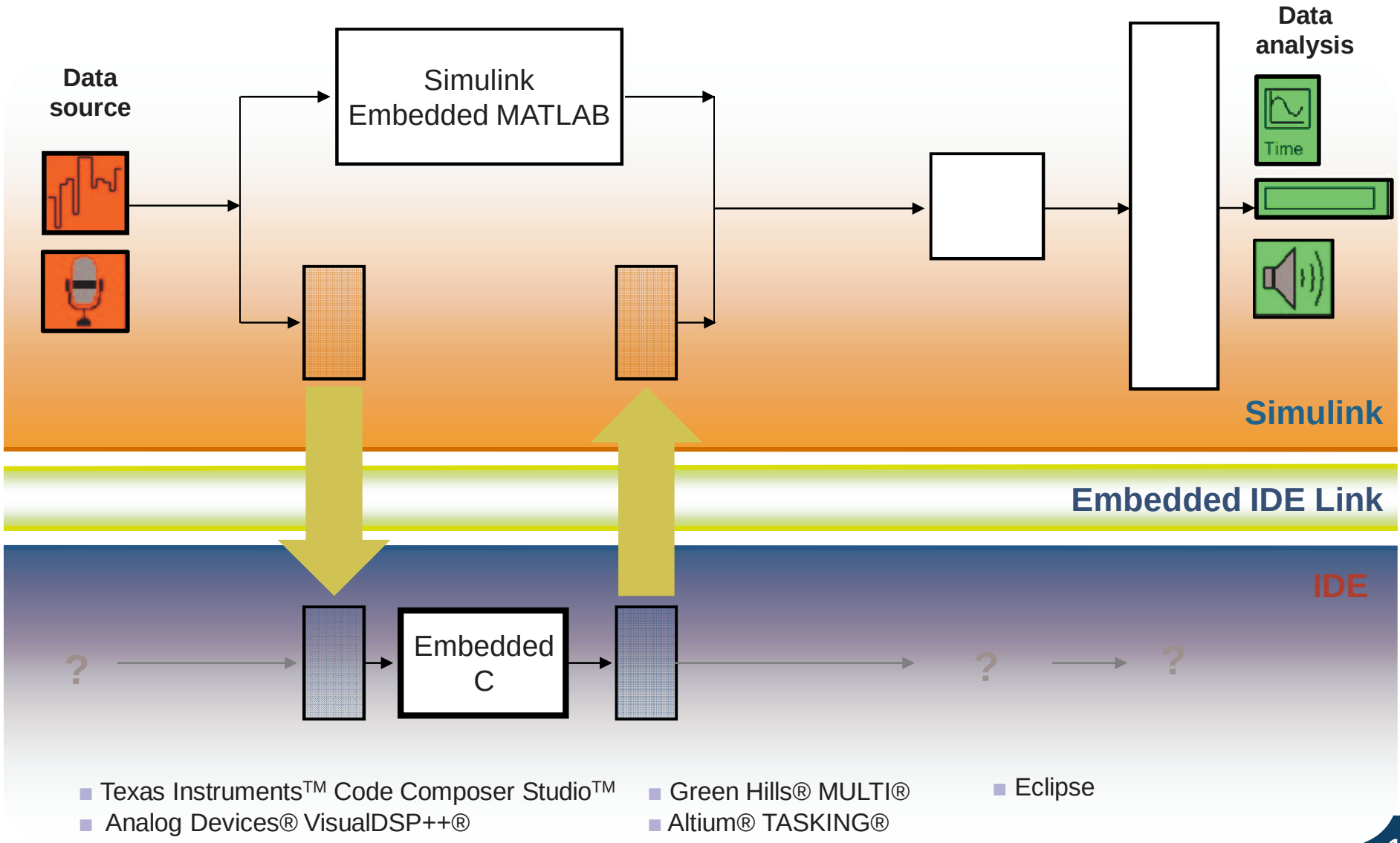


Software-in-the-Loop (SIL) Verification



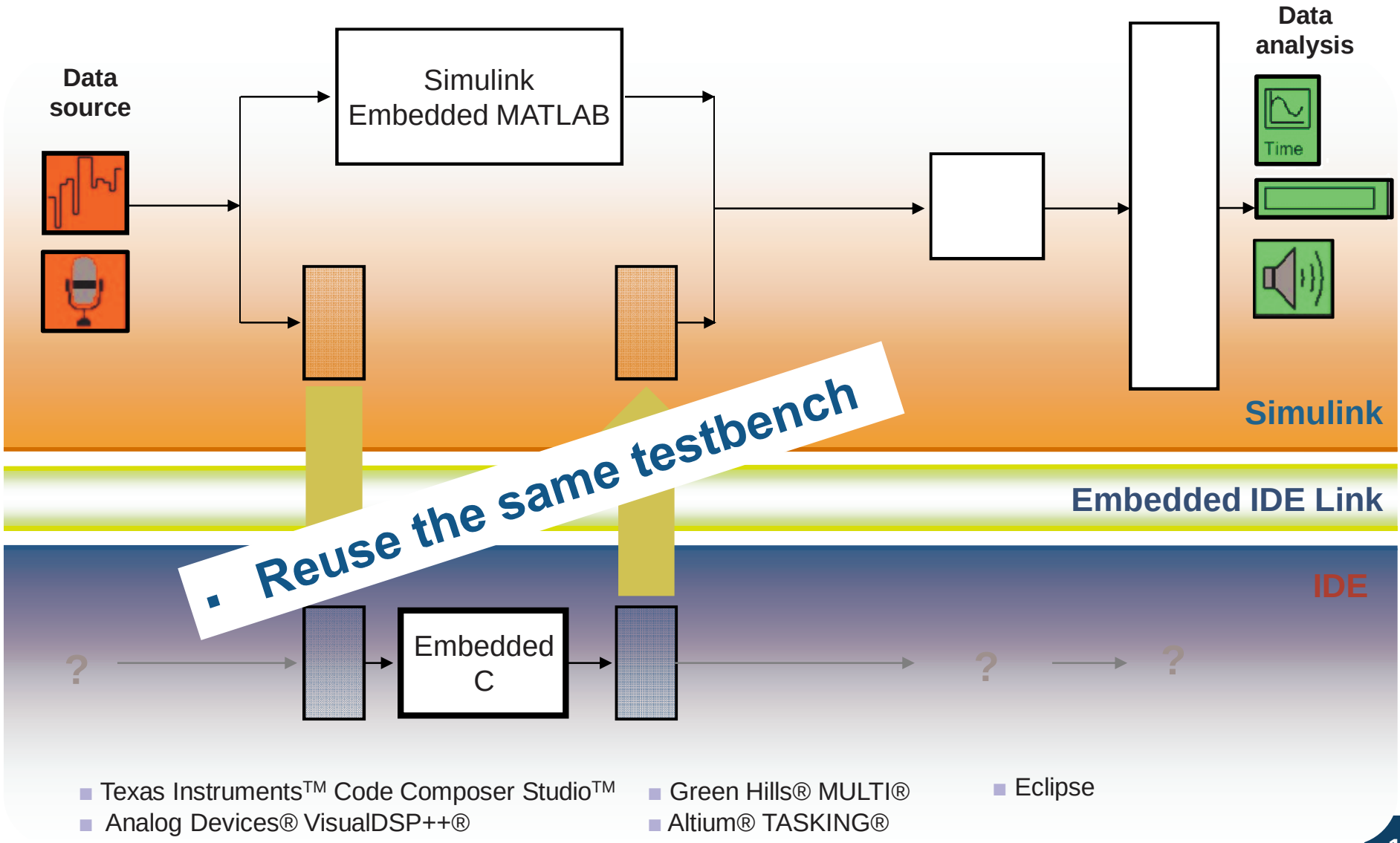
- Reuse the same testbench

Processor-in-the-Loop (PIL) Verification



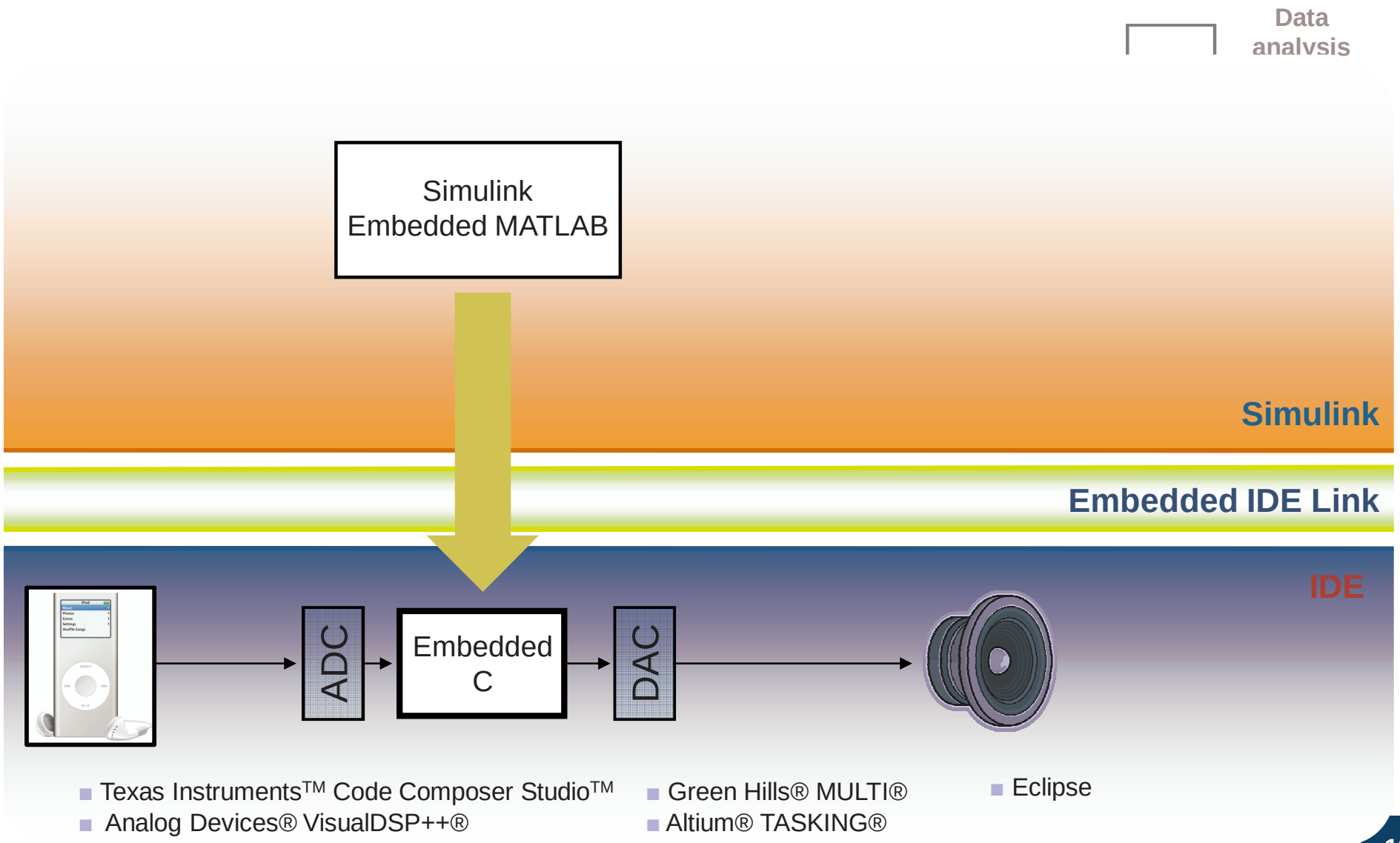
- Texas Instruments™ Code Composer Studio™
- Analog Devices® VisualDSP++®
- Green Hills® MULTI®
- Altium® TASKING®
- Eclipse

Processor-in-the-Loop (PIL) Verification

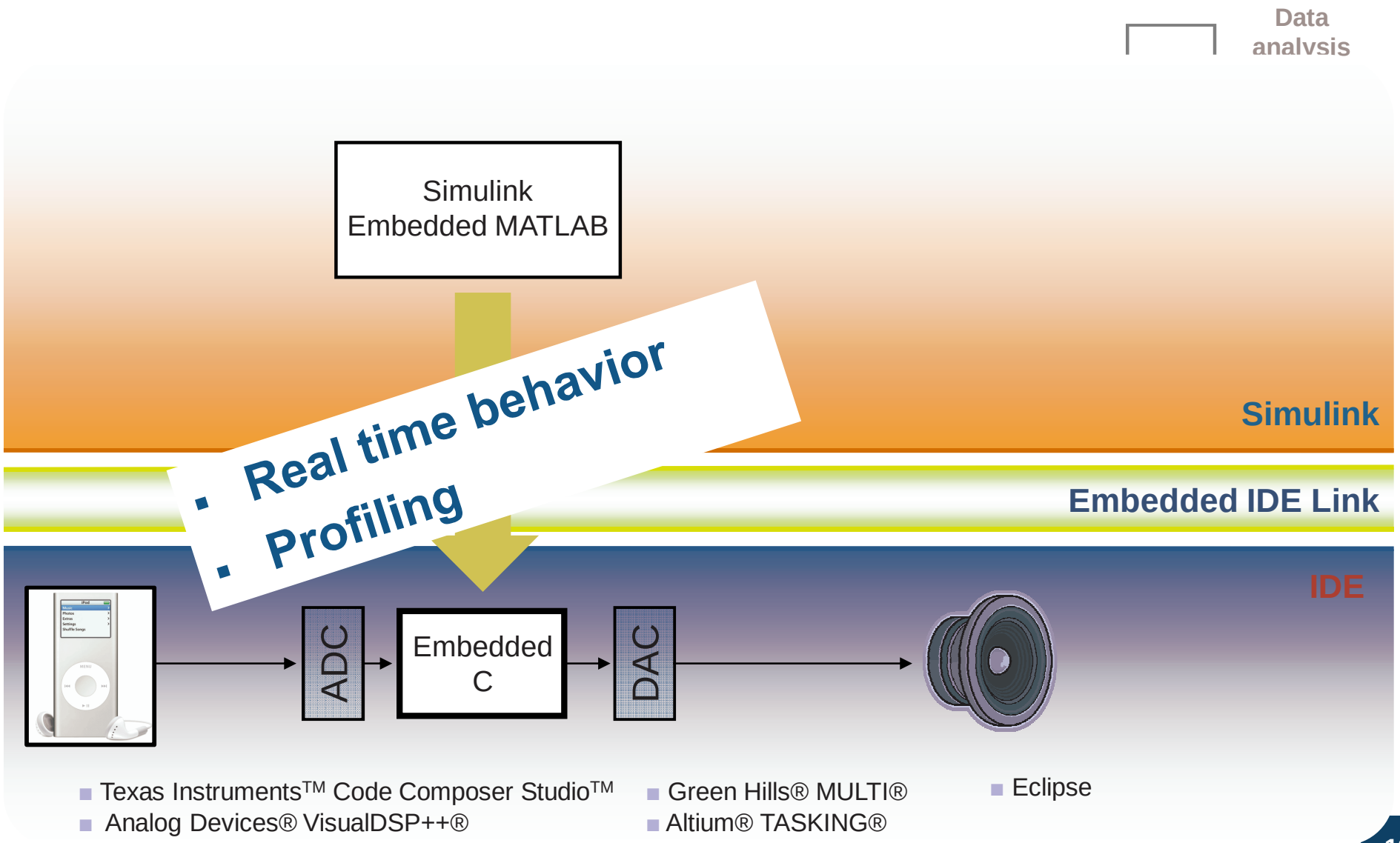


- Texas Instruments™ Code Composer Studio™
- Analog Devices® VisualDSP++®
- Green Hills® MULTI®
- Altium® TASKING®
- Eclipse

On-Target Rapid Prototyping



On-Target Rapid Prototyping



Quickly Iterate between Idea and Prototype

- ✓ First prototype is functionally correct with automatic C code generation
- ✓ Spend your time in optimizing rather than debugging the code
- ✓ Find errors reusing the same testbench at each design step