AduC7000 Series Development Tools

Keil Software and Analog Devices have teamed-up to provide you with a state-of-the-art development platform for the ARM powered AduC7000 Precision Analog MicroController device series.

The AduC7000 series integrate precision A/D and D/A converters, digital peripherals, in-system programmable Flash/EE memory, and on-chip debugging logic around an ARM7TDMI core.

For detailed technical specifications and further information refer to: www.analog.com/MicroConverter/ARM7.

Keil Software supports the AduC7000 series within the popular µVision IDE/Debugger (the same environment used for 8052-based AduC800 series projects). µVision3 integrates Compiler, Assembler, Editor, Project Manager, Debugger, CPU & Peripheral Simulator in a single intelligent environment. The debugger interfaces to the ARM7 embedded ICE via the ULINK USB-JTAG Adapter and offers target debugging and high-speed Flash programming.

Detailed product information is provided on www.keil.com/ARM.

Keil µVision3 integrates the development tools in a single, intelligent environment and provides full debugging support for all AduC devices.
Create a Project

The µVision3 Project Manager organizes all source files, tool configuration options, and build directions required to create a target program. A project may create several program targets from a single project (for example debug and release version).

To create projects for ADuC Devices:
1. Start µVision3 and use Project - New Project. Select an ADuC7000 series device from the Device Database. µVision adds relevant startup code to your project.
2. Use File - New to enter source code or add existing files in the Project Window via the Add Files to Group dialog.
3. Use Project - Build Target to generate application code.
4. Debug - Start/Stop Debug Session allows program testing and debugging with the CPU & Peripheral Simulation right on your desktop without target hardware.

Several example projects for the ADuC7000 series devices are available from: www.analog.com/MicroConverter/ARM7.

Choice of Tools

µVision3 allows you to select the toolchain of your choice and interfaces to the GNU GCC Compiler, ARM ADS/RealView, or Keil ARM Compiler. µVision3 even integrates your favorite utilities into the Tools menu. The SVCS menu accesses Version Control Systems such as CVS, Perforce, or MS VSS.

ULINK JTAG/USB Adapter

The Keil ULINK Adapter connects the PC USB port to the JTAG port of the ADuC7000 series device. Together with µVision, ULINK supports Flash download and program test on actual target hardware.

The embedded ICE logic of the ADuC7000 series device provides CPU register and memory access, breakpoints, and single stepping. No additional hardware or software components are required in your system.
Debugging ADuC7000 Series Programs

µVision3 has powerful simulation capabilities that give you serious benefits for rapid, reliable embedded software development.

- Simulation allows software testing on your desktop with no hardware environment.
- Early software debugging on a functional basis improves overall software reliability.
- Simulation allows breakpoints that are impossible with hardware debuggers.
- Simulation offers optimal input signals.
- Single-stepping through signal processing algorithms is possible. External signals are stopped when the CPU halts.
- Failure scenarios that would damage real hardware peripherals are easily checked.

**High-Speed Device Simulation**

The µVision3 Debugger accurately simulates the ADuC7000 series including the instruction set and on-chip peripherals. Peripheral simulation is composed of:

- Dialog boxes that allow you to view and rapidly configure peripherals.
- Virtual Registers (VTREGs) that allow you to control peripherals using breakpoints or the built-in debugging C script language.

You may use dialogs, virtual registers, and debug scripts to create high-fidelity simulations of your target hardware and environment.

**Debug Scripts**

The µVision3 Debugger incorporates a C script language you can use to create:

- Signal functions that simulate analog and digital input to the microcontroller. Signal functions run in the background while µVision3 simulates your target program.
- User functions that extend the command scope or combine repetitive operations.

Many built-in functions (like printf, memset, rand, sin, long, sqrt, and so on) are available. Debug functions may be invoked from the Command Window, from triggered Breakpoints, or from Toolbox Buttons.

```
FUNC float sine_wave (float f) {
    return __sin (f * (STATES / CLOCK) * 2 * PI);
}
ADC0 = sine_wave (2200) + sine_wave (4500);
```

Signal functions may generate analog and digital signal patterns using peripheral input pins (for example, ADC0 is Analog Input 0).

[http://www.keil.com/demo](http://www.keil.com/demo)
The ADuC7000 family of QuickStart Development Systems allows the designer to rapidly evaluate performance and functionality, and commence prototype development.


The ADuC7000 series is based on an ARM7TDMI core and integrates analog/digital features. Typical applications are: Process Control, Smart Sensors, Precision Instrumentation, Base Station Systems, Optical Networking, Automotive, and more.

<table>
<thead>
<tr>
<th>Part Number</th>
<th>ADC Channels</th>
<th>DACs</th>
<th>Flash/EE (bytes)</th>
<th>SRAM (bytes)</th>
<th>GPIO (max)</th>
<th>Comp.</th>
<th>PWM</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADuC7020</td>
<td>5</td>
<td>4</td>
<td>62K</td>
<td>8K</td>
<td>14</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>ADuC7021</td>
<td>8</td>
<td>2</td>
<td>62K</td>
<td>8K</td>
<td>13</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>ADuC7022</td>
<td>10</td>
<td>-</td>
<td>62K</td>
<td>8K</td>
<td>13</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>ADuC7024</td>
<td>10</td>
<td>2</td>
<td>62K</td>
<td>8K</td>
<td>30</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>ADuC7025</td>
<td>12</td>
<td>-</td>
<td>62K</td>
<td>8K</td>
<td>30</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>ADuC7026</td>
<td>12</td>
<td>4</td>
<td>62K+ext</td>
<td>8K</td>
<td>40</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>ADuC7027</td>
<td>16</td>
<td>-</td>
<td>62K+ext</td>
<td>8K</td>
<td>40</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

More devices to follow.