Matrox

Genesis Native Library Developer's Toolkit

Development toolkit for native programming of the TI C80™ DSP or Motorola® G4 PowerPC™ microprocessor.

Harness the full power of the Matrox Genesis or Genesis*Plus* vision processor

The Developer's Toolkit (DTK) complements the Genesis Native Library (GNL) by providing additional programming levels. These levels allow developers to fully extract the power of the Matrox Genesis and Genesis*Plus* vision processor boards. Developers can increase system determinism by running GNL control code directly on the on-board processor, further optimize an algorithm by merging GNL functions, or extend a GNL-based application with custom image processing and analysis functions implemented directly on the TI C80 DSP or Motorola G4 PowerPC microprocessor.

Increase system determinism

With the DTK, developers can easily move the control portion of an application from the host to the on-board processor to offload this time-critical task. As a result, real-time operations are guaranteed, even when the host processor is running a non real-time OS and is busy with other system tasks (i.e., MMI, disk, network, I/O). Porting all or just a portion of the application from the host to the on-board processor requires no specialized knowledge of the C80 or G4 PowerPC. DTK users still have access to all GNL functions and programming is performed in standard C/C++[3].

Optimize a GNL-based algorithm

DTK users can merge a sequence of GNL functions into a single new function to speed-up execution by reducing I/O with on-board main memory. Merged functions can run several times faster than the original sequence of GNL functions since intermediate results are stored in the processor's cache rather than in main memory. This kind of optimization is particularly easy to implement on the G4 PowerPC, since the DTK provides a large set of optimized low-level processing primitives that can be combined together to build more complex functions without writing any custom AltiVec™ code.

Write custom functions

While the GNL offers an extensive list of functions, there are some cases where a developer will need to write custom C80 or G4 PowerPC code to perform a specialized operation. For this purpose, the DTK allows for programming of the C80's Parallel Processors (PPs) or the G4 PowerPC's AltiVec unit directly.

Includes

- Matrox Genesis Native Library Developer's Guide
- Matrox Genesis Native Library in object code form (not linked)
- programming examples
- Matrox compiler and linker utilities
- advanced technical support

Requires

- Matrox Genesis or Genesis*Plus* vision processor board
- Matrox Genesis Native Library[1]
- Texas Instruments' TMS320C8x software development tools and documentation (for use with Matrox Genesis)
- Metrowerks® CodeWarrior for PowerPC Embedded Systems and Motorola documentation[2] (for use with Matrox Genesis*Plus*)

MATROX IMAGING

Write custom functions (continued)

Programming the C80 requires some knowledge of the internal architecture of the C80, including the workings of the Transfer Controller and the PPs. It also usually requires programming in assembly language in order to achieve maximum performance. For the G4 PowerPC, programming is done exclusively in C/C++ using language extensions and intrinsic functions (see insert below).

The DTK provides the framework to facilitate the writing of custom functions and dramatically reduce software development time. For example, when programming for the G4 PowerPC directly, DTK users only need to write a simple inner loop to process whole aligned vectors. Variable image sizes, unaligned images, and mixed data types are all automatically supported. Moreover, memory bandwidth is maximized since pre-fetching is also handled automatically.

AltiVec programming

The first loop is standard C code, which adds two blocks of 8-bit pixels together (one pixel at a time).

```
unsigned char *Src1, *Src2, *Dst;
...
for (X = 0; X < SizeX; X++)
Dst[X] = Src1[X] + Src2[X];
```

The next version uses the AltiVec programming model to perform the same operation much faster, because it works on one vector (sixteen 8-bit pixels) at a time.

```
vector unsigned char *vSrc1, *vSrc2, *vDst;
...
/* Get the size in vectors, not pixels */
SizeX = SizeX / vec_step(vector unsigned char);
for (X = 0; X < SizeX; X++)
    vDst[X] = vec_add(vSrc1[X], vSrc2[X]);
```

The only differences are that the pointers are declared to be a vector type, the loop counter is divided by 16 (because 16 pixels are processed at once), and the addition is specified by the AltiVec intrinsic function vec_add() rather than with the standard "+" operator.

Development platform

The hardware platform used to create, compile and debug custom code is a host PC with a Matrox Genesis or Genesis*Plus* vision processor board. Code is generated on the host PC and then downloaded to the C80 DSP or G4 PowerPC microprocessor for testing and debugging.

For the Matrox Genesis platform, users will need Texas Instruments' C80 software development tools for code generation and debugging. These development tools consist of the C80 PC Software Toolkit and C80 hardware emulator. TI's C8x PC Software Toolkit includes compilers, assemblers, a debugger and linker, as well as a register allocator and code compactor to simplify application development. The hardware emulator enables system-level integration and debugging, and interfaces to the Matrox Genesis via the JTAG connection.

For the Matrox Genesis*Plus* platform, users will need Metrowerks CodeWarrior for PowerPC Embedded Systems. CodeWarrior consists of a compiler, linker and debugger all working within an integrated development environment. CodeWarrior makes embedded code development as easy as developing applications for the PC. Moreover, with CodeWarrior, there is no need for a hardware emulator since the debugger interacts directly with the Matrox Genesis*Plus* over the PCI bus.

Comes with preferred access

DTK customers have preferential access to our experienced staff of developers specializing in coding imaging functions for the Matrox Genesis or Genesis*Plus* architecture.

Ordering Information

Part number	Description
Hardware	
Genesis/DTK	Matrox Genesis/Genesis*Plus* Developer's Guide and access to the Native Library (C80 and G4 PowerPC versions) in object code form.

Matrox Genesis users will also need to purchase TI's C8x software development tools (version 2 or later) directly from TI:

Software

Part number	Description
TMDX3248855-07	C8x PC Software Toolkit.
TMDX3240180	C8x Emulator[4].
XDS510	Hardware Emulator[4].

Matrox Genesis*Plus* users will also need to purchase Metrowerks CodeWarrior for PowerPC Embedded Systems (version 5 or later) directly from Metrowerks:

Software

Part number	Description
CE-PPCW5	CodeWarrior for PowerPC™ Embedded Systems, hosted on Windows®, Release5

Notes:
1. Version 2.2 or later required for Matrox Genesis*Plus*.
2. AltiVec Technology Programming Environments and Interface manuals, MPC7400 RISC Microprocessor User's manual.
3. C++ programming is only available with the Matrox Genesis*Plus*.
4. Optional, but required for debugging.

matrox

For more information, please call: 1-800-804-6243 (toll free in North America)
or (514) 822-6020 or e-mail: imaging.info@matrox.com or http://www.matrox.com/imaging