Unleash the power of Intel multi-core with INtime

INtime turns a PC into a powerful industrial system equipped with a MPU and Realtime DSPs.

Introduction

In recent years, it is common that processors have more than a single core. Multi-core gives more multi-processing power, but it hasn't been widely used in industrial/embedded systems. Many industrial systems still consist of a processor running Windows or some variant of GUI front-end system and an embedded processor running real-time OS to do time critical tasks. INtime Realtime OS from TenAsys corporation has been providing a unique capability that makes it possible to run real-time tasks simultaneously with Windows on a single PC. With the introduction of INtime 4.0, now it is possible to run realtime tasks on multiple cores independently.

System on a Chip

INtime is a full-featured Realtime OS for x86 architecture, supporting virtual memory and multi-processing/multi-threading model. The most unique feature of INtime is that it can co-exist with Windows. In INtime 4.0, it can control each core. For example, a typical Intel Core 2 Duo processor with Hyper-threading can be treated as 4 core processor, with one processor assigned to Windows and the rest of the cores dedicated to INtime (these 'cores' are called 'Nodes' in INtime terminology). All the four cores run independently, and each Node has its own memory space. So practically, this is a system with a Windows front-end and three real-time embedded processors.

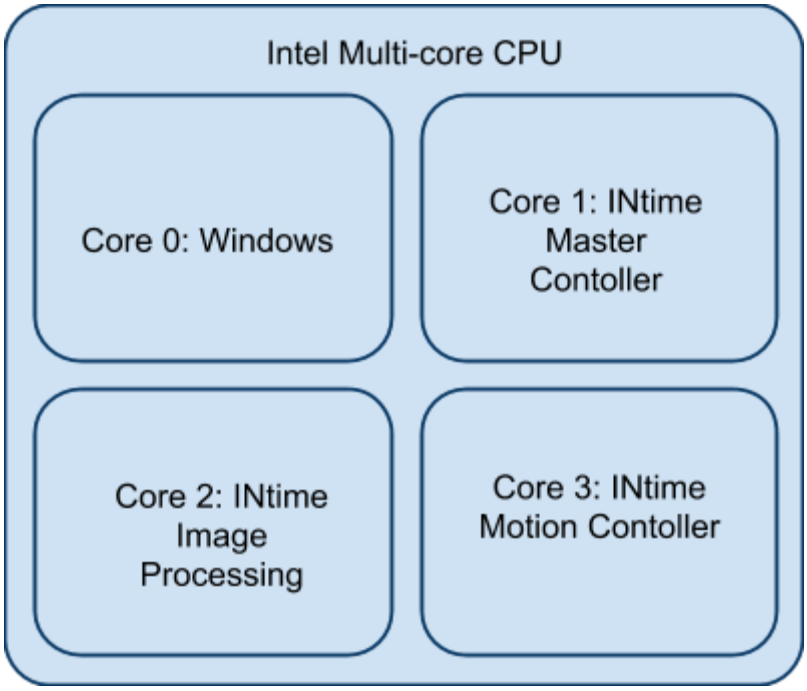

Figure 1: Motion Controller Example

Figure 1 shows a possible Motion Control system using INtime running on an Intel multi-core processor. Windows provides GUI to the operator. Core 1 through 3 run INtime OS, giving them real-time, deterministic operations; Core 1 acts as the master controller, taking the data from the Image Processing unit on Core 2 and giving commands to the Motion Controller system running on Core 3.

Power of the Multi-core

This is practically a tightly coupled multi-processor system built on a single architecture. The advantages are:

- Based on a single architecture: No need to have multiple compilers/tools to develop/debug system.
- Ease of development: Microsoft Visual Studio is used for both Windows and INtime development.
- High Performance: Intel x86 processors are much faster than typical MPU and its SSE instructions fares well against typical DSPs.
- Efficient communication between cores: INtime provides efficient synchronization mechanism, such as shared memory and mailbox.
- Coherent memory system: Unlike most of the embedded system, cache coherency is maintained by hardware, eliminating the possibility of difficult to find cache related bugs.

Demonstration System

INtime 4.0 with its multi-core support opens a lots of new possibilities to developers; one

possibility is to replace existing hetero-processor system with a single multi-core system. Another possibility is achieving maximum parallel performance in a deterministic way. To demonstrate the multi-core capability, TenAsys developed a simple image processing system (Figure 2).

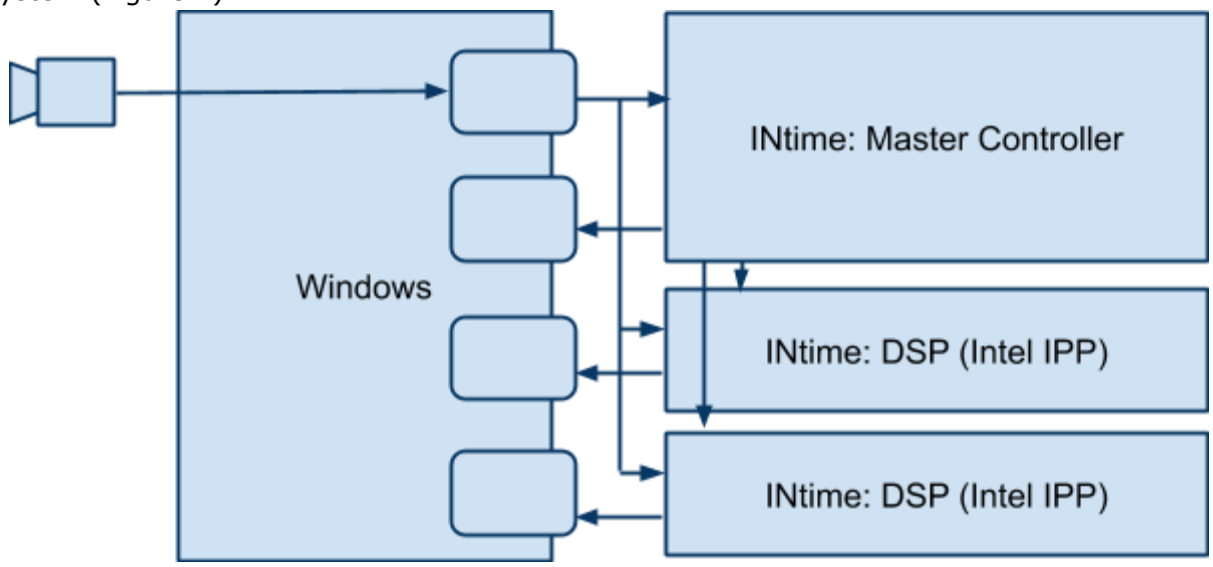

Figure 2: Demo system block diagram

This system produces 3 filtered image from a single source image in parallel using 3 INtime nodes. It consists of a 4 core Intel CPU:

- Core 0: Windows -- runs an image capture application(*1). Also displays Image Processing results from INtime nodes.
- Core 1: INtime -- acts as a master controller. Communicates with Windows GUI application, sends commands to other 2 nodes, process image by itself.
- **Core 2 and 3: INtime -- performs Image Processing task using Intel IPP (Integrated Performance Primitives. Please see the article "Utilizing SIMD Instructions on the INtime RTOS with the Intel IPP Library" on how to use Intel IPP in INtime environment).**

(*1): For real-time, deterministic operation, the image acquisition should be done on an INtime node with a PCI image capture board or a GigE camera. To use these devices, device driver need to be ported to INtime OS.

The image acquired on Windows is copied to a shared memory created by INtime. The image is also shared among all INtime nodes. The master node signals other nodes to start processing using mailboxes. Each node processes the image and produces filtered image into another shared memory, and signals its peer Windows application to display the image. Figure 3 shows the running application.

In practical system, multi-core may be used for processing in ping-pong buffering to maximize the number of image to be processed per a unit time.

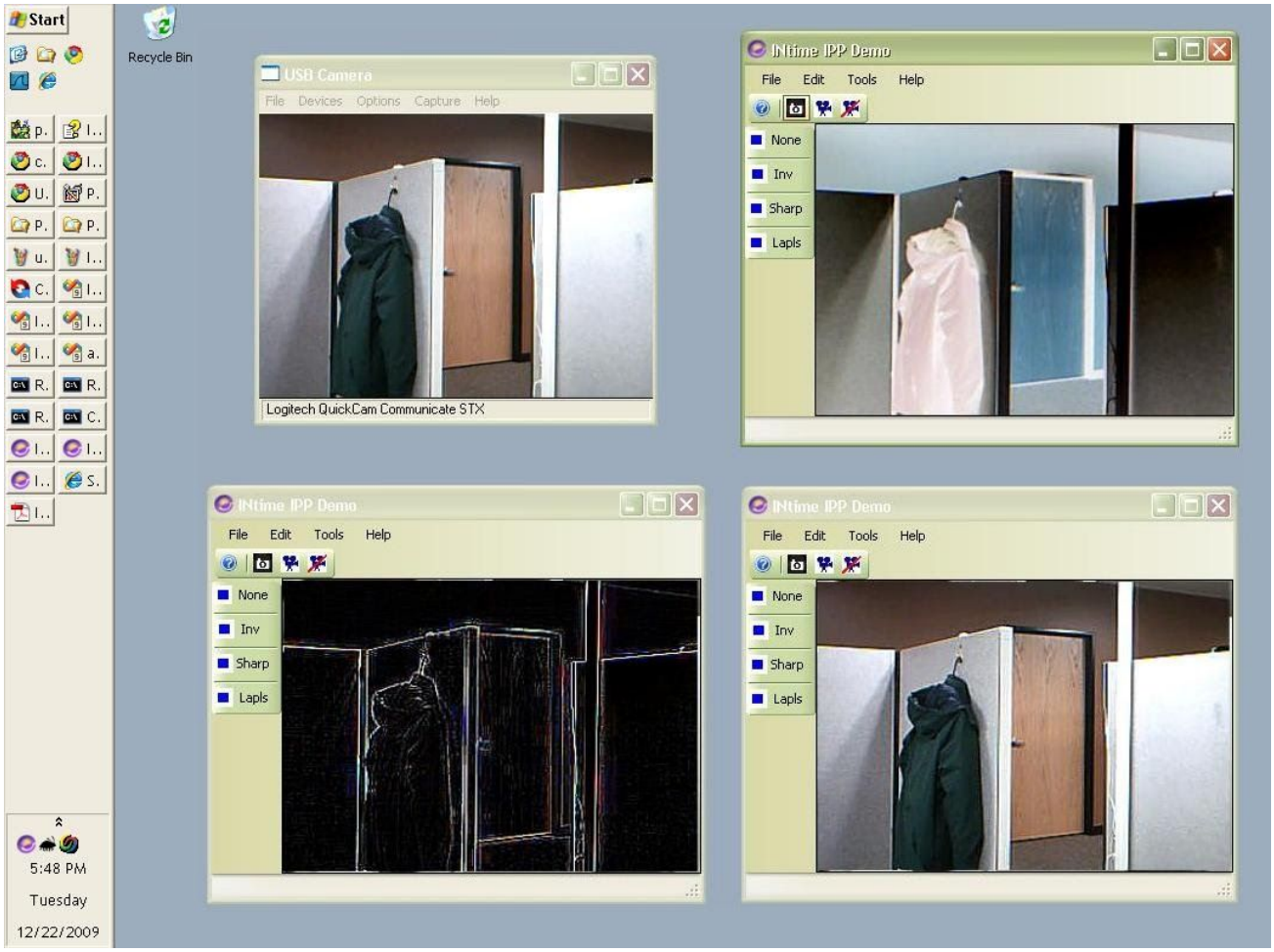

Figure 3: Demo; upper-left is input image, others display processed images from INtime nodes

Conclusion

INtime Realtime OS, with its multi-core support and determinism, opens a new door to industrial/embedded systems using Intel x86 architecture. Especially high-performance, real-time applications, such as motion control, image processing and speech recognition, can benefit from INtime enabled Intel's multi-core processors. For more information, please contact info@tenasys.com.