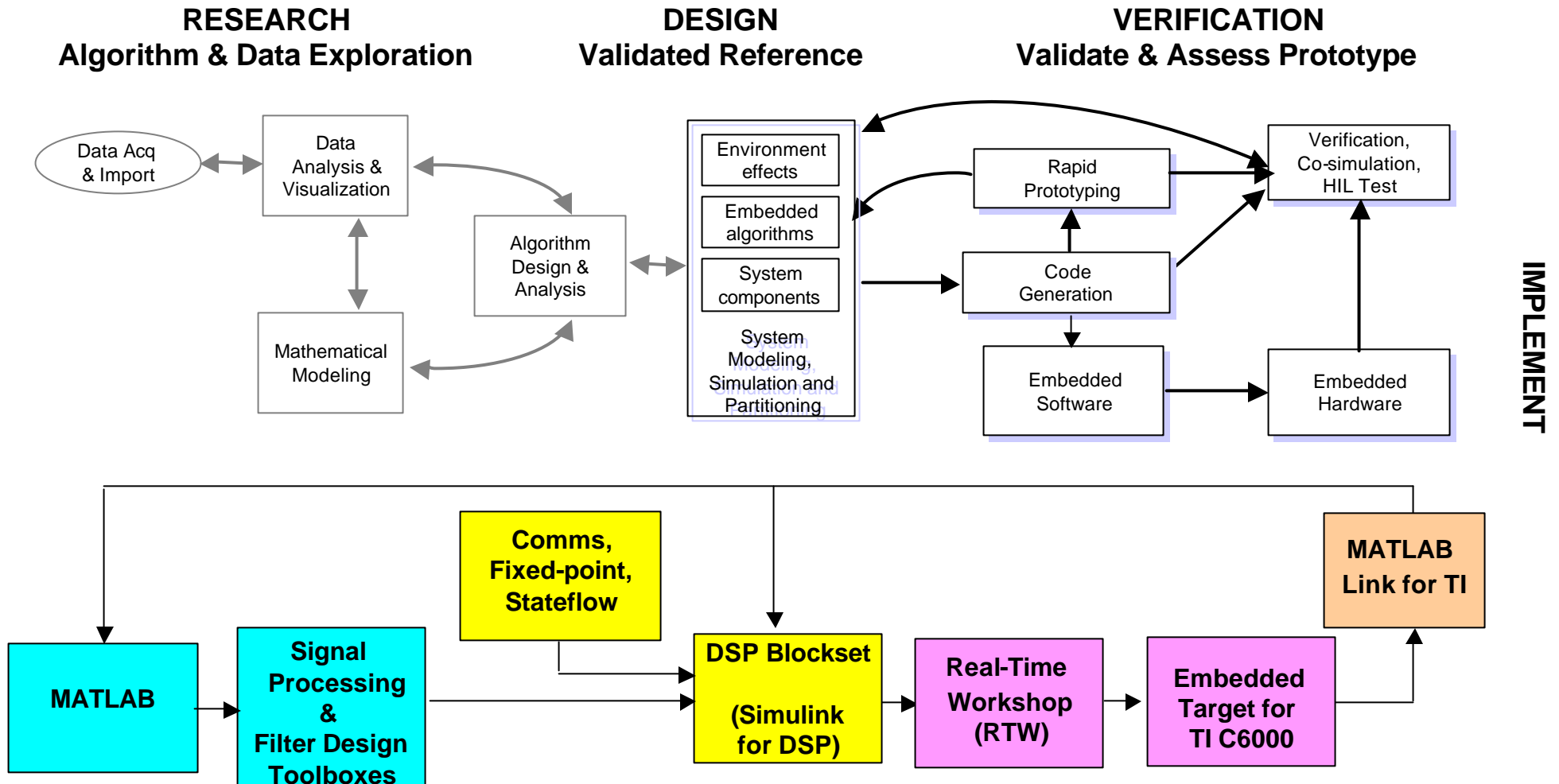


# Application: Surveillance Data-Stream Compression

- ✍ **Need:** Continuous monitoring of scene with video camera
    - ✍ Security (ATM booth, parking lot), Intelligent Highways, Bio/Pharma, etc.
  - ✍ **Problem:** Generates large volumes of data to record, archive, and review
    - ✍ Preferable to reduce data stream at the source (embedded compression)
  - ✍ **Solution:** Shrink data storage requirements
    - ✍ Reduce **size of each** video frame to record, and/or
    - ✍ Reduce **total number** of video frames to record
  - ✍ **Simple Idea:** **Motion** within camera's field of view **triggers storage** of "interesting" frames
- Video Application:** Identify and record only these "interesting" video frames



# Embedded System Strategy: Model-Based Design

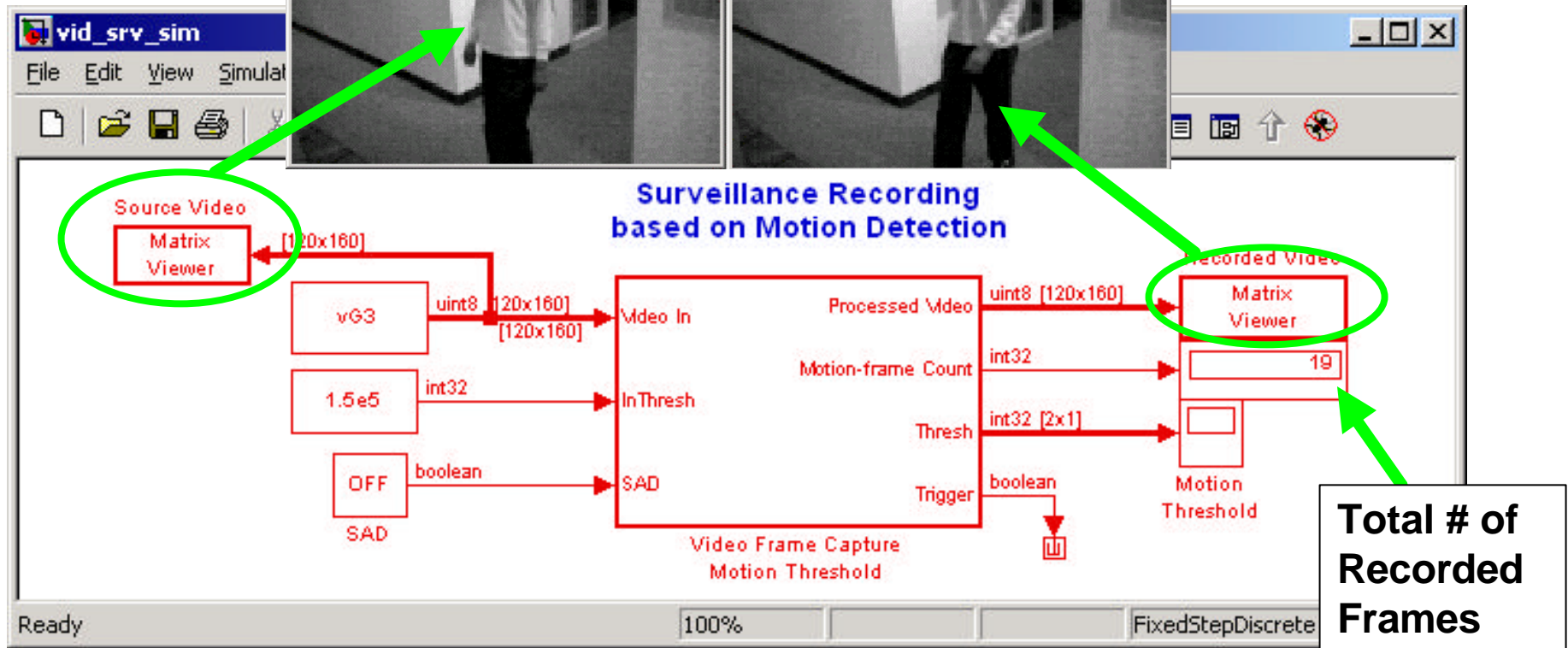


# Surveillance Compression

Source Frames  
High rate: 30 fps



Recorded Frames  
Aperiodic rate  
( $\ll$  30 fps)



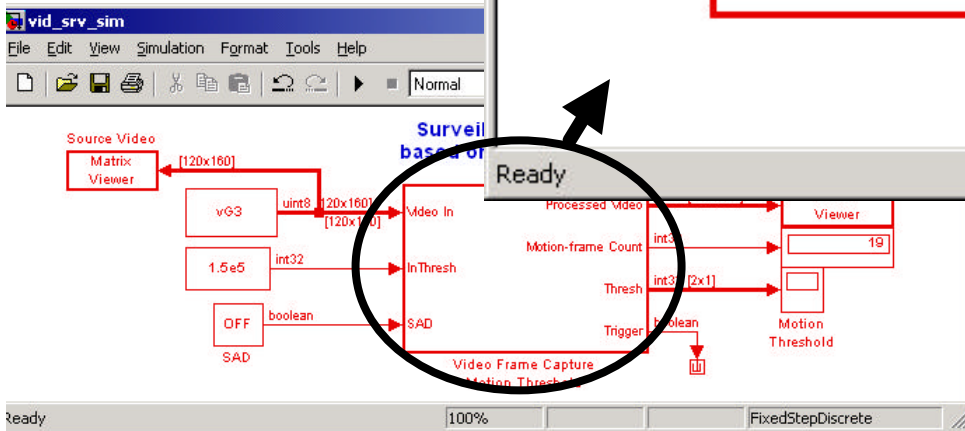
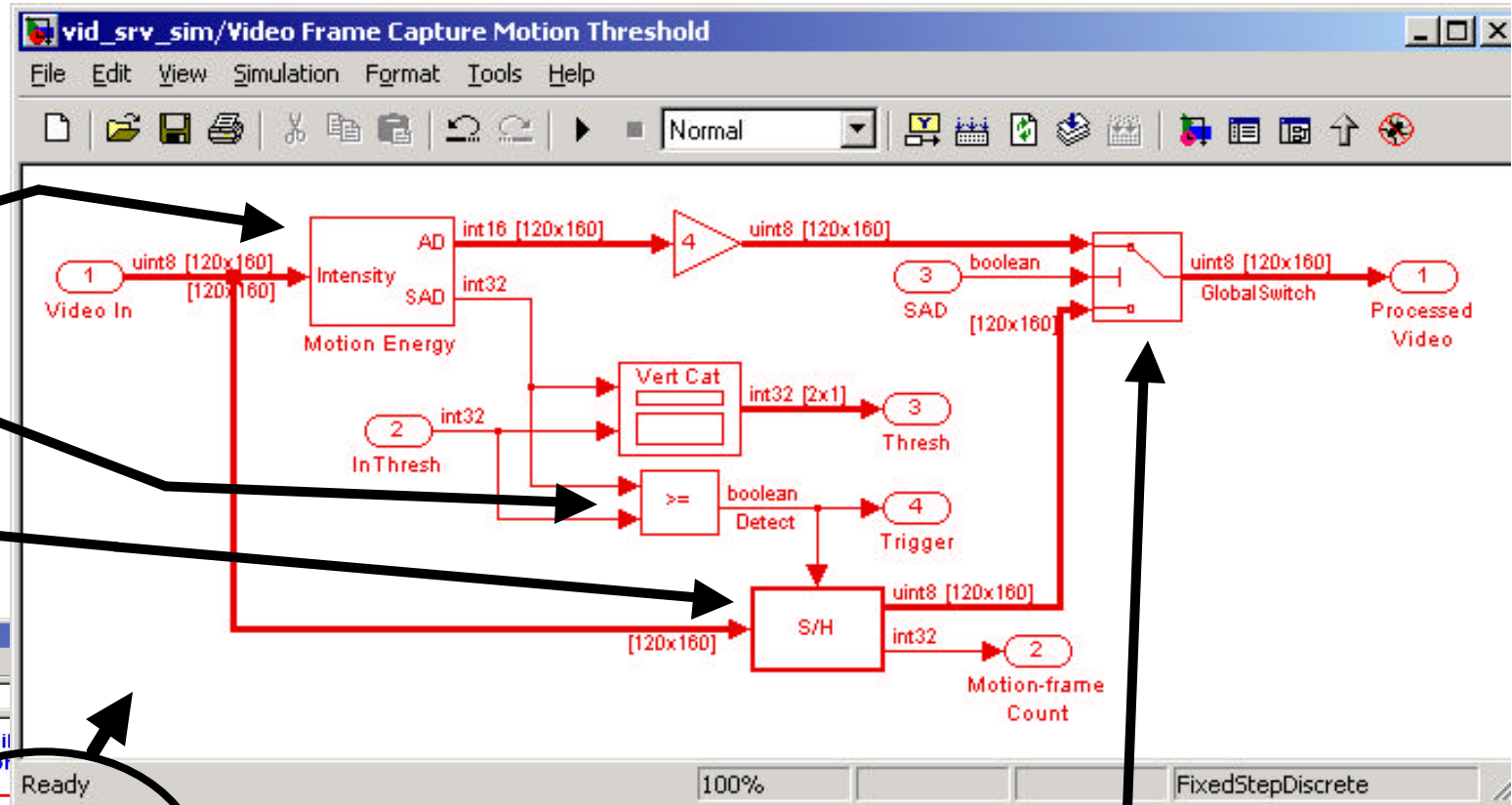
# Subsystem Hierarchy

## Model Overview

Estimate motion

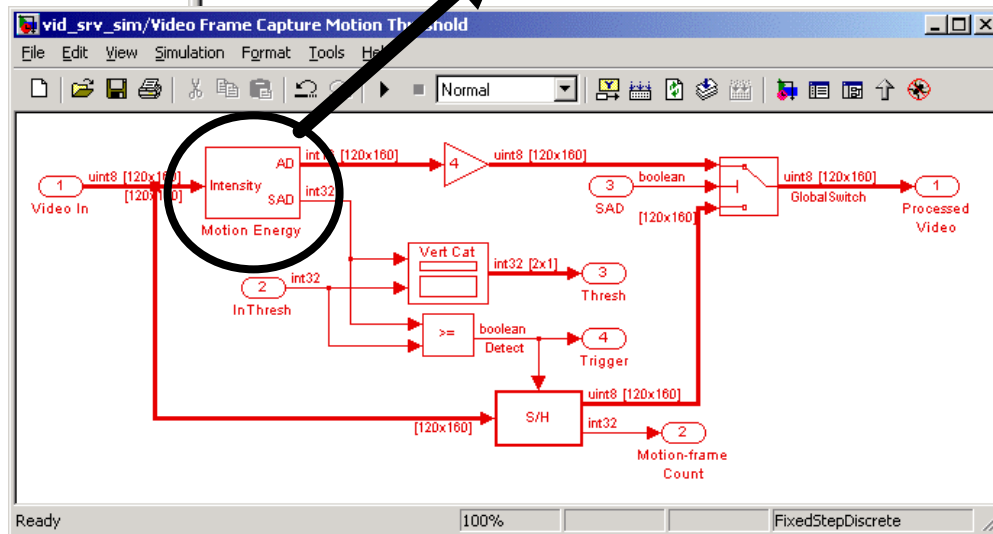
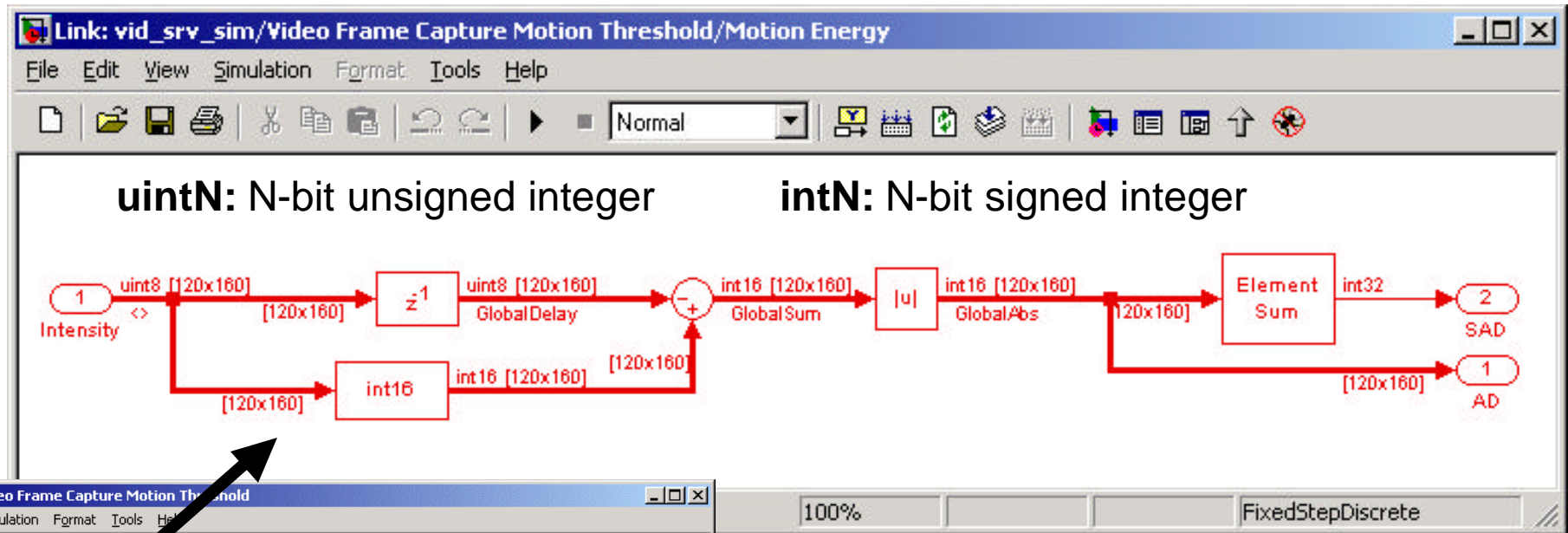
Compare threshold

Record frames & update count



Allows switching between **compressed** and **absolute differences** video streams on-the-fly, for enhanced testing

# SAD Algorithm for Motion Estimation



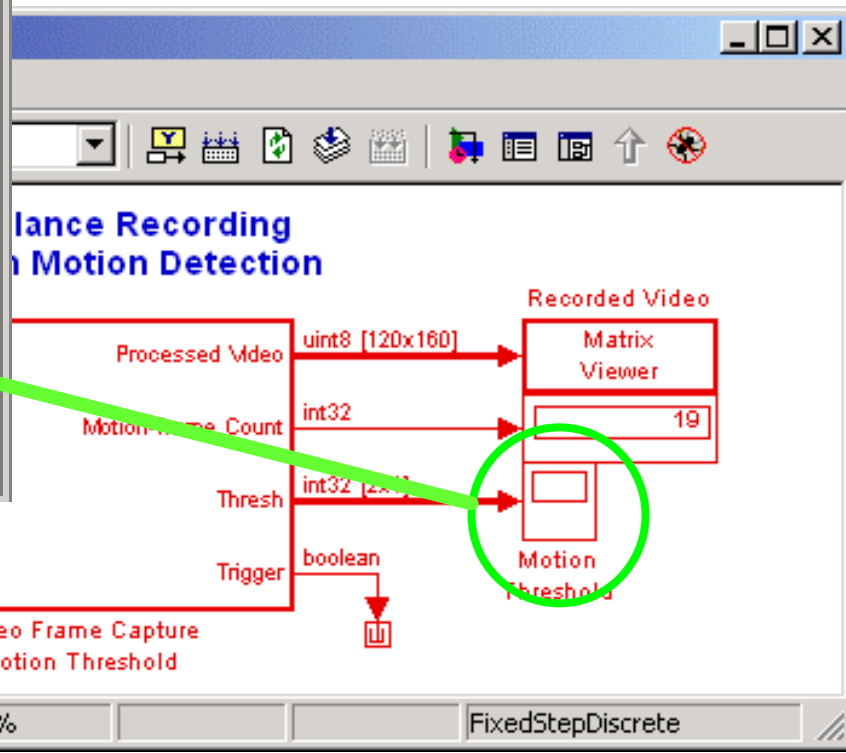
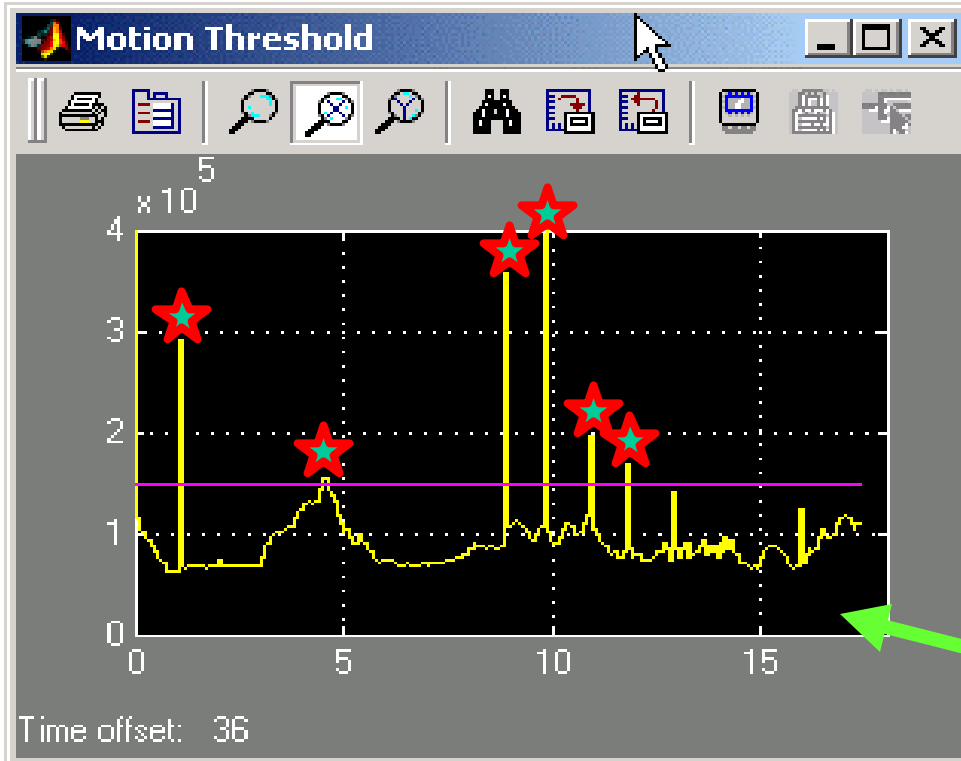
## Sum of Absolute Differences

- Very simple method of inter-frame video motion detection
- Has efficient fixed-point (integer) implementation in hardware

# Motion Detection via Thresholding

**Motion levels**

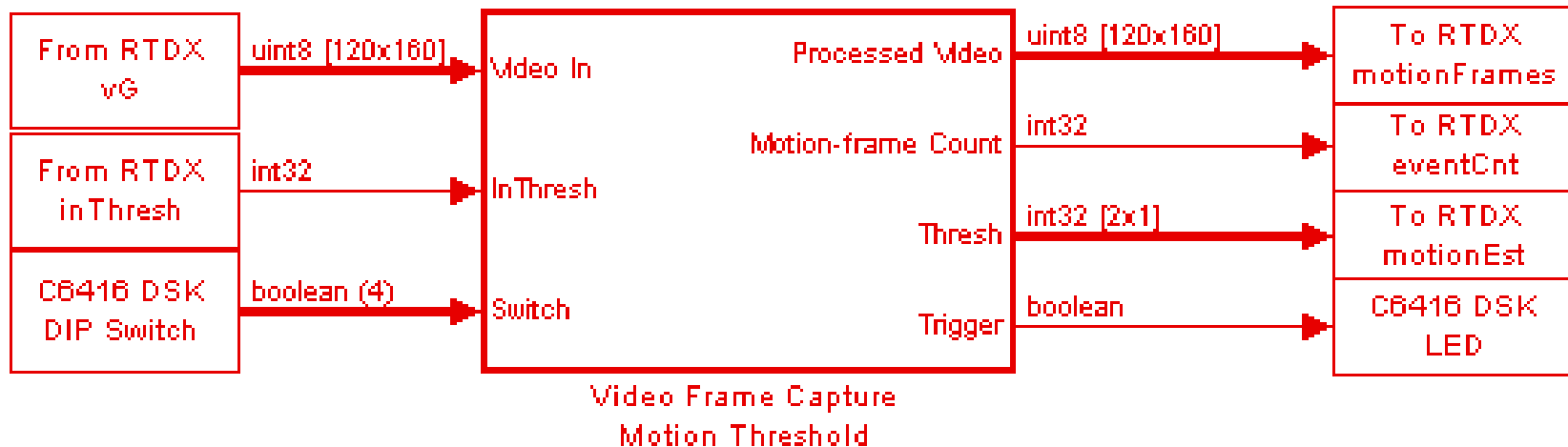
-  Video Motion Estimate
-  Detection Threshold
-  Captured Video Frames



# Steps to Target the TI C6416 DSK

## 1. Utilize I/O blocks to prepare for on-target system verification

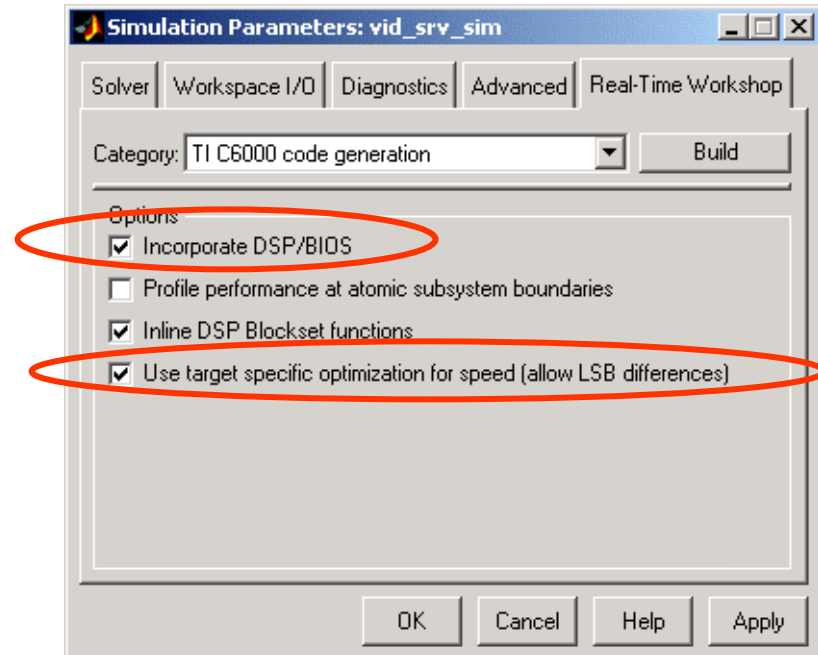
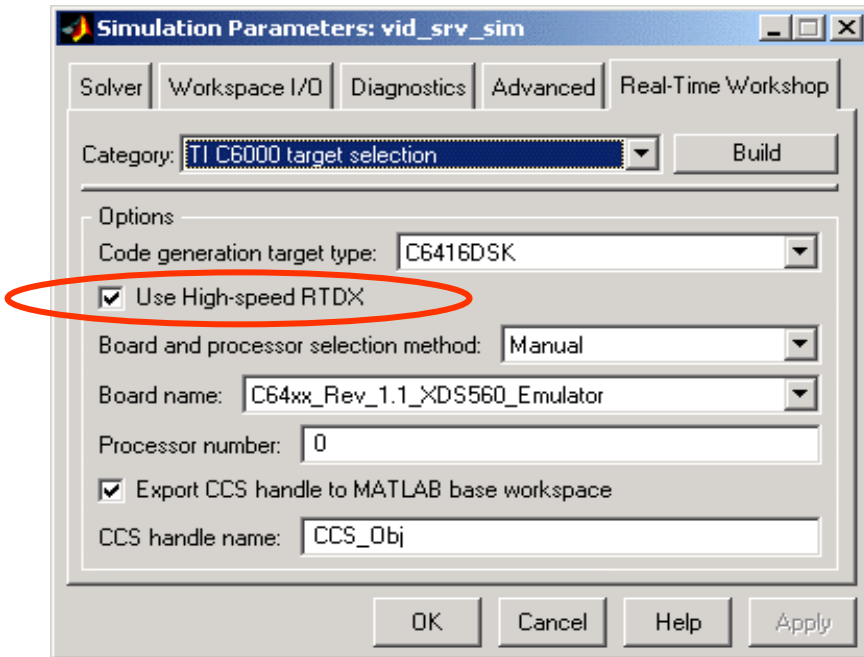
- Use board support library components, such as RTDX Input/Output, codecs, LEDs, switches, etc



**RTDX:** TI protocol for Real-Time Data eXchange  
 High-Speed RTDX offers **~2 MB/sec** bandwidth

# Steps to Target the TI C6416 DSP

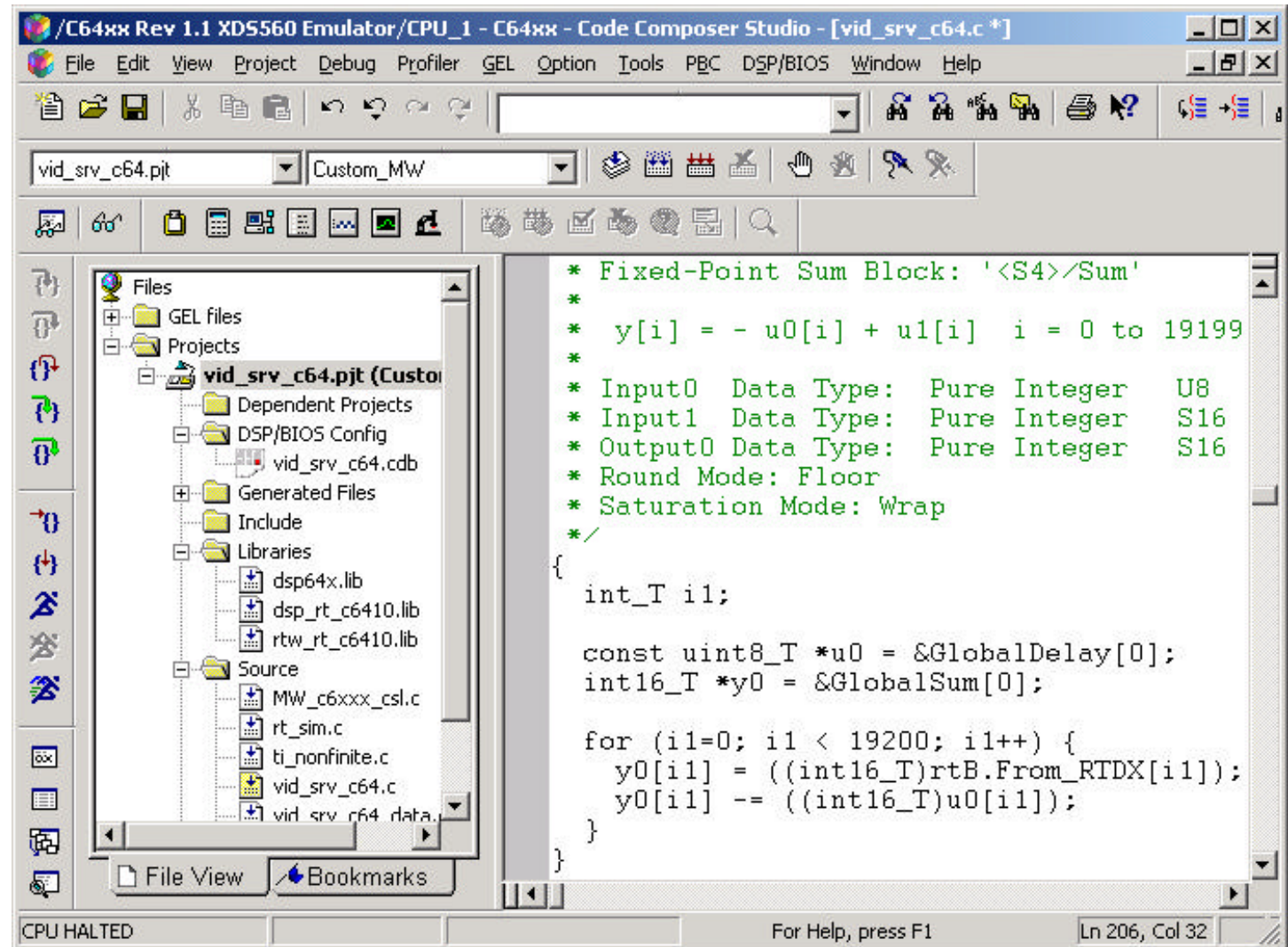
## 2. Select target options (DSP/BIOS, compiler settings, etc)



# Steps to Target the TI C6416 DSP

## 3. Build process

- Auto-generate ANSI C and ASM code
- Integration of RTOS and scheduler
- Create full CCS project in IDE
- Invoke compiler, linker, and download code
- Run target



# Steps to Target the TI C6416 DSP

## 4. Automatic profiling of program executing on DSP

System profiling  
 Includes entire DSP application code

Subsystem profiling

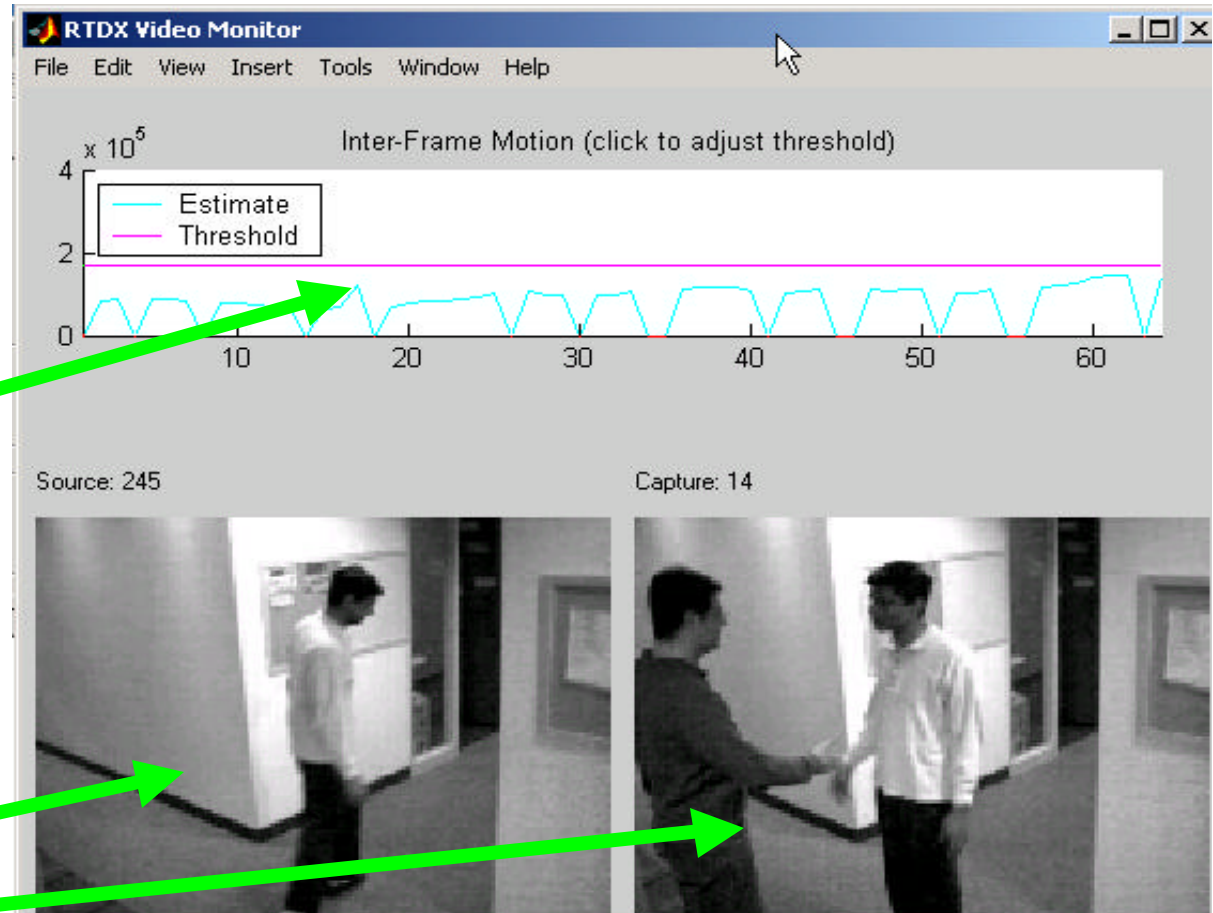
The screenshot shows the MATLAB Profiler interface with two profiling reports. The top report is for the system 'vid\_srv\_c64' and the bottom report is for the subsystem 'vid\_srv\_c64/Video Frame Capture Motion Threshold'. Both reports show the 'Max percent of interrupt interval' value circled in red.

System name	Max percent of interrupt interval
vid_srv_c64	20%
vid_srv_c64/Video Frame Capture Motion Threshold	4.3%

# Design Verification: Real-time Visualization

## MATLAB Application

- ✎ **Host-side visualization**
- ✎ **MATLAB M-file GUI**
  - ✎ Using **Link for TI CCS** and standard Handle Graphics functionality
- ✎ **Motion Estimates**
  - ✎ Log and plot estimates over time (scrolling data)
  - ✎ Adjust detection threshold (click mouse on graph)
- ✎ **Monitor video capture**
  - ✎ Input video frames
  - ✎ Captured frames



# Design Verification

## *Automating embedded software verification cycle*

- ✎ **MATLAB Host Application uses *Link for TI CCS IDE* for communication with Code Composer Studio**
  - ✎ **Extends MATLAB language to interact with CCS**
  - ✎ **Provides project load/run/stop management, debug points, data manipulation, HIL, and co-simulation support**
  - ✎ **Supports real-time data exchange (RTDX)**
- ✎ **Use Link to visualize & verify correct behavior of code running on any TI DSP (C2000, C5000, C6000, OMAP, TMS470)**
- ✎ **Equips MATLAB with interactive DSP verification and test-bench validation capabilities**

## **Example CCS Link M-file**

```
% Connect to CCS & DSP:
CCS_Obj = ccspdsp;
r = CCS_Obj.rtdx;

% Configure RTDX channels
r.open('vidIn', 'w', ...
       'vidOut', 'r', ...
       'motionEst','r');
CCS_Obj.run; % Run target app
r.enable;    % Enable RTDX chans

% Write data to DSP (real-time)
r.writemsg('vidIn', nextFrame);

% Read data from DSP
outFrame = r.readmsg('vidOut',
                    'uint8',[120 160]);
```

## Summary: Simulink for Video Processing

- ✍ **Graphical block-diagram design environment**
  - ✍ Time-oriented, stream-based signal processing
  - ✍ Matrix-based, multi-rate, multi-tasking, frame-aware
  - ✍ Floating-point, fixed-point, and integer data types
  - ✍ Automatic code generation via Real-Time Workshop
  - ✍ Interoperability with MATLAB

## Next Steps: FPGA Implementation

- Third-Party FPGA Design Products
  - Xilinx System Generator
  - Altera DSP Builder
- MathWorks HDL Products
  - Link for ModelSim™ (New: Oct'03)
  - HDL Filter Designer (Beta: Oct'03)



## Next Steps: High Performance Video

- ✍ Fixed point enhancements to Simulink and MATLAB
- ✍ Image Acquisition Toolbox
- ✍ Video processing components (Beta: Jan'04)

## Next Steps: For More Information

- ✍ **Upcoming Simulink video processing webinar**
  - ✍ [www.mathworks.com/webinar\\_tid](http://www.mathworks.com/webinar_tid)
- ✍ **The MathWorks products**
  - ✍ [www.mathworks.com/products](http://www.mathworks.com/products)
- ✍ **Additional Simulink video processing examples**
  - ✍ [www.matlabcentral.com](http://www.matlabcentral.com)

# Simulation and Real-time Verification of Video Algorithms on the TI C6400 DSP

## Synopsis of Poster C.10

**Poster C.10** - Thu 25 Sep 2003

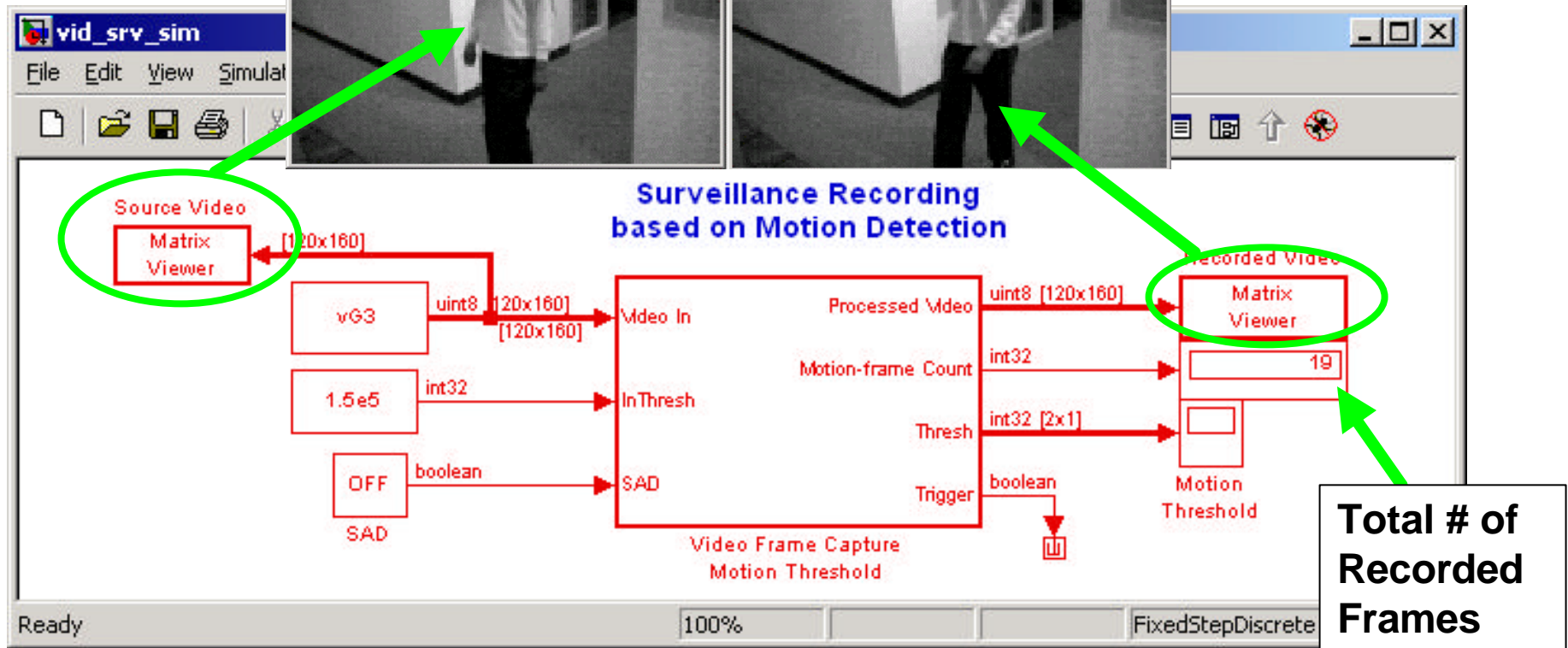
Don Orofino  
Manager, DSP Development  
The MathWorks, Inc.

# Surveillance Compression

**Source Frames**  
High rate: 30 fps

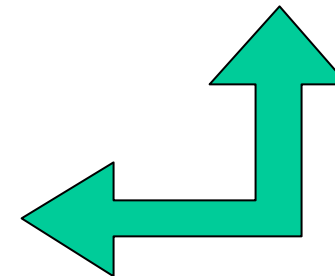
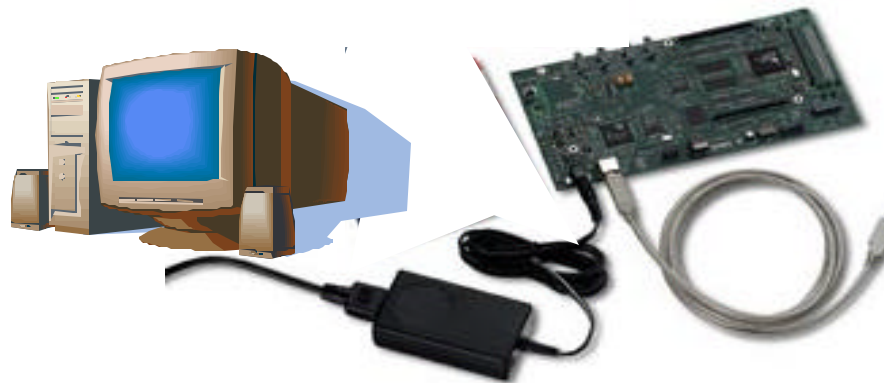
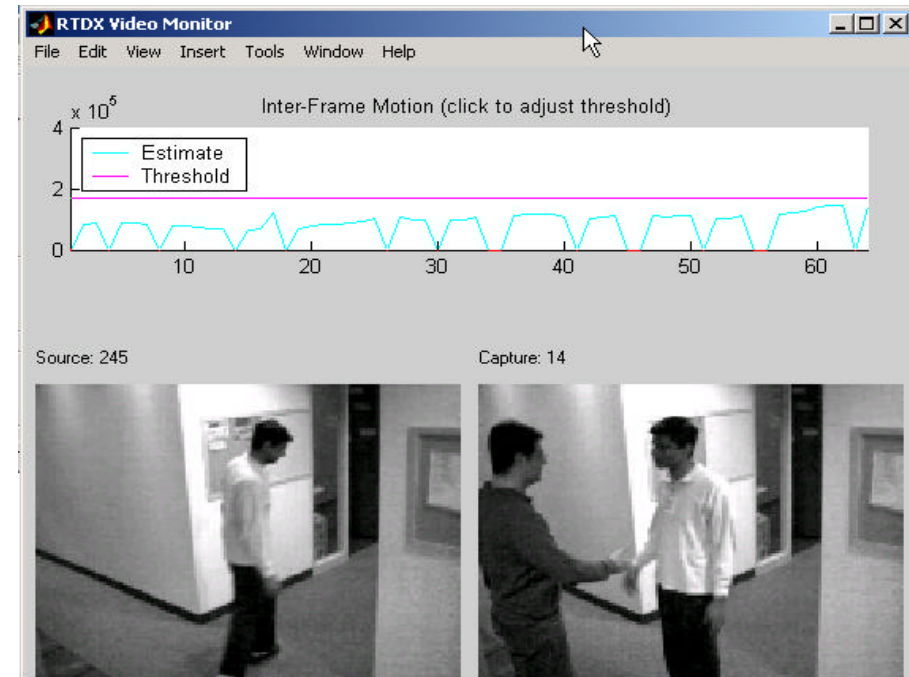


**Recorded Frames**  
Aperiodic rate  
( $\ll$  30 fps)



# Video system demonstration

- **DSP running video application**
  - “On-chip Rapid Prototyping”
  - **TI C6416 DSK:** High-performance fixed-point DSP targeted by Simulink
  
- **MATLAB Host-side GUI**
  - Link for TI Code Composer Studio utilized for quick programming
  - Bi-directional data transfer for full-speed testing/visualization



# Steps to Target the TI C6416 DSP

## 4. Automatic profiling of program executing on DSP

System profiling  
 Includes entire DSP application code

Subsystem profiling

The screenshot shows a 'Profile Report' window with two entries. The first entry is for the system 'vid\_srv\_c64' and the second is for the subsystem 'vid\_srv\_c64/Video Frame Capture Motion Threshold'. Both entries show their respective 'Max percent of interrupt interval' values circled in red.

System name	Max percent of interrupt interval
vid_srv_c64	20%
vid_srv_c64/Video Frame Capture Motion Threshold	4.3%

# Simulation and Real-time Verification of Video Algorithms on the TI C6400 DSP

**Poster C.10** - Thu 25 Sep 2003

Don Orofino  
Manager, DSP Development  
The MathWorks, Inc.

# Poster Panel Layout

