# Module 1-5: Processes, Threads, and Virtual Memory

## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview</td>
<td>2</td>
</tr>
<tr>
<td>Lab 1-5: Using the Remote Tools to Examine Memory and Process Layout</td>
<td>9</td>
</tr>
<tr>
<td>Review</td>
<td>10</td>
</tr>
</tbody>
</table>

---

**Module 1-5: Processes, Threads, and Virtual Memory**
Instructor Notes

This module covers the process, thread and virtual memory model of Windows CE. This is a high level coverage of the concepts adding a few layers of information onto the architecture provided earlier.

The primary learning point is the distinction between processes and threads and clarifying the terminology of those. Different RTOS and OS terminology exists including “task” which is not used in Windows CE at all. There is also the idea of Fibers which are not discussed or covered to any significant degree in this course. Although there is a lab on day 3 that demonstrates them as a means to force inclusion of and OS feature/API Set.

The memory model will continue to receive more detailed coverage throughout the course. The important point to clarify is that Windows CE is always using virtual memory and that there is only one single Virtual Memory Space for all processes.

In this module, students will learn:

- A Process is a static context within which one or more threads execute
- A thread is the unit of execution that is scheduled in Windows CE
- There is a limit of 32 Processes in Windows CE
- Each Process limited to 32M
- Single 4GB Virtual address space for all processes
- Processes are protected from accessing other processes
- Memory is divided into 32M slots; each process gets one slot;
- Current Process is mapped to Slot 0 so threads always execute in context of Slot 0

To teach this module, you need the Microsoft® PowerPoint® file Module 1-5.ppt.

To prepare for this module:

- Read all the material for this module.
- Complete the lab.
Overview

In this module, you will learn:

- A Process is a static context within which one or more threads execute
- A thread is the unit of execution that is scheduled in Windows CE
- There is a limit of 32 Processes in Windows CE
- Each Process limited to 32M
- Single 4GB Virtual address space for all processes
- Processes are protected from accessing other processes
- Memory is divided into 32M slots; each process gets one slot;
- Current Process is mapped to Slot 0 so threads always execute in context of Slot 0
Windows CE Kernel Features

- **Multiple processes**
  - Can support maximum of 32 separate processes
- **Multiple threads**
  - Supports 256 thread priorities
- **Fibers**
  - Unit of execution that must be manually scheduled by the application
- **Synchronization objects**
  - Critical Sections, Mutexes, Semaphores, Events, Message Queues
- **Memory model**
  - Virtual memory, Code sections Paged, No backing store for Data sections

Windows CE always uses virtual memory. The only place that deals with physical addresses is a small part of the start up code and the functions used for mapping virtual memory.
Processes

- **Static context within which one or more threads run**
  - Processes aren’t scheduled to run – threads are.

- **The maximum number of simultaneous processes is limited 32 processes because:**
  - It is a reasonable limit for most embedded devices, as using multi-thread is recommended over multi-processes
  - Architecture of some supported CPUs has fixed MMU mappings.

- **Windows CE uses the same loading and unloading mechanism as Windows XP**
  - (and other desktop Win32 versions of Windows)

- **Support for console applications**
  - But not the same API as desktop Win32

- **Call CreateProcess() to start a process**

Processes themselves do not run and are not scheduled they are simply a container of resources for threads.
Threads

- Unit of execution in Win32
- Scheduled by the OS based on Priority
- Higher priority threads pre-empt lower priority threads when ready to run
- Threads at the same priority are scheduled in a Round-Robin fashion.
- Default Quantum is 100ms configurable by OEM in OAL
  - Can also be programmed per thread at run time.

Threads are what the system schedules for execution. Applications should NEVER assume a default quantum as the OEM can configure the default as needed by the system as a whole.
Thread Priority

- Thread A is in the highest priority
- If at any point while thread A is running, thread B or C wake up
- If A is blocked, and both B and C are awake
- If thread A wakes up at any point during execution of any thread below its priority level

Threads are scheduled according to priority. The thread with the highest priority that is runnable is given time by the scheduler. If more than one thread exists at the same priority they are scheduled in a round robin fashion. If a thread has the quantum set to 0 then it runs until it completes or blocks (e.g. no round robin)
Memory Architecture

- **Physical Memory**
  - Actual RAM, ROM and Devices with addresses as they appear on the external (or internal) bus

- **Virtual Memory**
  - Memory system that runs addresses through a Memory Management Unit (MMU) that translates a "Virtual" address into a physical one.
  - Allows for paging code in to memory as needed
Virtual Memory

Windows CE uses a single 4G virtual address space for all processes. The lower 2G is divided into slots used by the process (one slot for each process). You’ll learn more about the details of the memory architecture in day 2 of this course.
Lab 1-5: Using the Remote Tools to Examine Memory and Process Layout

Introduction
This lab introduces the Remote tools provided with Platform Builder as a means to explore the processes, threads and memory of a running Windows CE OS Image.
1. How many processes can Windows CE load simultaneously?
   Windows CE supports a maximum of 32 processes.

2. How many threads can a process have running?
   Theoretically, infinite; practically, the system resources and 32M per process limits the number to a little more than 500 threads which is well beyond any rational design needs.

3. What is the maximum amount of RAM Windows CE supports?
   Theoretically, 512M. However, this is reduced by the amount of system statically mapped addresses so in practice is less than 512.