# Lab 8

## Processors: Function That Can Be Called From C

## Purpose

To learn how to write an assembly language function that can be called from a C program.

## Background Reading And Preparation

Read Chapter 8 to learn about subroutine calls in assembly languages, and read the C and assembler reference manuals to determine the conventions that C uses to call a function on your local computer.

## Overview

Write an assembly language function that can be called from a C program to perform the *exclusive or* of two integer values.

## Procedure And Details (checkmark as each is completed)

1. Write a C program that calls function *xor* with two integer arguments and displays the result of the function.
2. Create an assembly language function, *xor*, that takes two integer values as arguments, computes the *exclusive or* of the two arguments, and returns the result as the value of the function.
3. Add a *printf* call to the *xor* function to verify that the function correctly receives the two values that the C program passes as arguments (i.e., argument passing works correctly).
4. Add a *printf* call to the C function to verify that the *xor* code returns the correct value.

## Optional Extensions (checkmark as each is completed)

- [ ] 5. Modify the C program and the *xor* function so the C program passes a single structure as an argument instead of two integers. Arrange for the structure to contain two integer values.

## Notes