CPU_Memory.asm

```
; =====================================================================
; ASSEMBLY CODE - JARED: SYSTEM AND MEMORY INITIALIZATION
; System: 8086 10MHz + 8087 Coprocessor + 1MB RAM
; =====================================================================

.MODEL SMALL
.STACK 4096
.DATA

; === SYSTEM CONSTANTS ===
DRAM_CTRL_PORT       EQU 90h        ; DRAM control port
REFRESH_PORT         EQU 91h        ; Refresh timer port
BANK_SEL_PORT        EQU 93h        ; Memory bank selector port
DRAM_CONFIG_PORT     EQU 94h        ; DRAM configuration port

; === BASE ADDRESSES FOR TEAM ===
BASE_INTERRUPTS      EQU 0C0000h     ; Person 3
BASE_DMA             EQU 0C0100h     ; Person 3
BASE_SERIAL          EQU 0C0200h     ; Person 3
BASE_PARALLEL        EQU 0C0300h     ; Person 3
BASE_KEYBOARD        EQU 0C0400h     ; Person 2
BASE_DISPLAY         EQU 0C0500h     ; Person 2
BASE_ADC_DAC         EQU 0C0600h     ; Person 4
BASE_USB             EQU 0C0700h     ; Person 3
BASE_PRINTER         EQU 0C0800h     ; Person 2
BASE_FLOPPY          EQU 0C0900h     ; Person 4

; === SYSTEM VARIABLES ===
msg_start           DB 'Initializing 8086 System...', 0Dh, 0Ah, '$'
msg_memory_ok       DB 'Memory initialized: 1MB OK', 0Dh, 0Ah, '$'
msg_8087_ok         DB 'Coprocessor 8087 OK', 0Dh, 0Ah, '$'
msg_error_memory    DB 'ERROR: Memory failure', 0Dh, 0Ah, '$'
msg_error_8087      DB 'ERROR: Coprocessor 8087 not found', 0Dh, 0Ah, '$'
msg_system_ready    DB 'System ready for use', 0Dh, 0Ah, '$'

test_pattern_1      DW 0AA55h       ; Memory test pattern 1
test_pattern_2      DW 055AAh       ; Memory test pattern 2
memory_ok           DB 0            ; Memory status flag
coprocessor_ok      DB 0            ; 8087 status flag

.CODE

; =====================================================================
; MAIN FUNCTION - COMPLETE SYSTEM INITIALIZATION
; =====================================================================
start:
    mov ax, @data
    mov ds, ax
    mov es, ax

    mov ax, 9000h
    mov ss, ax
    mov sp, 0FFFEh
    cli

    lea dx, msg_start
    call show_message

    call configure_buses
    call initialize_memory
    call initialize_8087

    ; (Other initialization as needed)

    hlt                 ; Halt CPU (placeholder for end of main routine)

; =====================================================================
; === DUMMY PROCEDURE STUBS (for documentation, not implemented) ===
; =====================================================================
show_message PROC
    ret
show_message ENDP

configure_buses PROC
    ret
configure_buses ENDP

initialize_memory PROC
    ret
initialize_memory ENDP

initialize_8087 PROC
    ret
initialize_8087 ENDP

END start
```