```
                //
                // RAM
                // RAM:0000-RAM:0fff
                //

                ***************************************************************
                * MC68705U3 - 8-Bit EPROM Microcontroller Unit               *
                *                                                            *
                * The MC68705U3 (HMOS) Microcontroller Unit (MCU) is an E... *
                * The user programmable EPROM allows 'program changes and... *
                * This low cost MCU has parallel I/O capability with pins... *
                *                                                            *
                * • Internal 8-Bit Timer with 7-Bit Programmable Prescaler    *
                * • On-chip Oscillator                                        *
                * • Memory Mapped 1/0                                         *
                * • Versatile Interrupt Handling                              *
                * • Bit Manipulation                                          *
                * • Bit Test and Branch Instruction                           *
                * • Vectored Interrupts                                       *
                * • Bootstrap Program in ROM                                  *
                * • 112 Bytes of RAM                                          *
                * • 3776 Bytes of EPROM                                       *
                * • 24110 Pins                                                *
                ***************************************************************
                PORT A I/O Lines

                Connected to bus PA(7:0) for READ/WRITE.
                PA(7:0) will output signals to IDB(7:0) via CHIP 3B if EPANS …

                PORTA                                           XREF[48]:    Strobe_WMM_Read_From_FIFO:09bf(R…
                                                                             Output_Response_To_ND120_IDB:09c…
                                                                             Output_Status_Code_1_To_ND120:09…
                                                                             Output_Status_Code_1_To_ND120:09…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Output_Status_Code_1_To_ND120:0a…
                                                                             Initialize_MM58274_RTC_Chip:0ac6…
                                                                             [more]
0000 00                db          0h
```

```
                PORT B I/O Lines

                PB0 = /WMM (clock signal to latch PA(7:0) data on chip 32B / …
                PB1 = /WRCLK (connected to MM58247 RTC pin 3 /WD)
                PB2 = /ROCLK (connected to MM58247 RTC pin 2 /RD)
                PB3 = /RMM (signal to somewhere(?) on the CPU board)
                PB4 = STAT3
                PB5 = STAT4
                PB6 = READ (IDB:13)
                PB7 = STAT7 => INVERTED TO PRES (IDB:15) <= PRESENT SIGNAL if…



                PORTB                                            XREF[77]:    Wait_For_FIFO_Commands_And_Proce…
                                                                              Wait_For_FIFO_Commands_And_Proce…
                                                                              Wait_For_FIFO_Commands_And_Proce…
                                                                              Wait_For_FIFO_Commands_And_Proce…
                                                                              Output_RTC_Time_Data_To_ND120:06…
                                                                              Output_RTC_Time_Data_To_ND120:06…
                                                                              Output_RTC_Time_Data_To_ND120:06…
                                                                              Output_RTC_Time_Data_To_ND120:06…
                                                                              Output_RTC_Time_Data_To_ND120:07…
                                                                              Strobe_WMM_Read_From_FIFO:09bd(R…
                                                                              Strobe_WMM_Read_From_FIFO:09c1(R…
                                                                              Output_Response_To_ND120_IDB:09c…
                                                                              Output_Response_To_ND120_IDB:09c…
                                                                              Output_Status_Code_1_To_ND120:09…
                                                                              Output_Status_Code_1_To_ND120:09…
                                                                              Output_Status_Code_1_To_ND120:09…
                                                                              Output_Status_Code_1_To_ND120:0a…
                                                                              Output_Status_Code_1_To_ND120:0a…
                                                                              Output_Status_Code_1_To_ND120:0a…
                                                                              Output_Status_Code_1_To_ND120:0a…
                                                                              [more]
0001 00              db         0h

                PORT C I/O Lines

                STAT 0-4 goew to IDB(8:11)

                PC0 = STAT0
                PC1 = STAT1
                PC2 = STAT2

                DISP1-5 is to control display ? Goes to bus /DP(5:1) via inve…

                PC3 = DISP1
                PC4 = DISP2
                PC5 = DISP3
                PC6 = DISP4
                PC7 = DISP5


                PORTC                                            XREF[13]:    Wait_For_FIFO_Commands_And_Proce…
                                                                              Delay_And_Sample_Port_C:04a7(R),
                                                                              Delay_And_Sample_Port_C:04b0(W),
                                                                              Delay_And_Sample_Port_C:04b2(W),
                                                                              Delay_And_Sample_Port_C:04b6(W),
                                                                              Delay_And_Sample_Port_C:04b8(RW),
                                                                              Delay_And_Sample_Port_C:04be(W),
                                                                              Delay_And_Sample_Port_C:04c0(RW),
                                                                              Delay_And_Sample_Port_C:04cc(W),
                                                                              Delay_And_Sample_Port_C:04ce(RW),
                                                                              Delay_And_Sample_Port_C:04d5(W),
                                                                              Output_RTC_Time_Data_To_ND120:06…
                                                                              Output_RTC_Time_Data_To_ND120:06…
0002 00              db         0h
```

```
                 PORT D *INPUT* lines
                 PIN PD6 can be /INT2


                 PD0 = PCR0
                 PD1 = PCR1
                 PD2 = PONI
                 PD3 = IONI
                 PD4 = LHIT
                 PD5 = LEV0 (Level 0. Meaning CPU inactive)
                 PD6 = HIGH (To avoid external interrupt to be triggered)
                 PD7 = /EMP
                 PORTD                                           XREF[5]:     Wait_For_FIFO_Commands_And_Proce…
                                                                              Wait_For_FIFO_Commands_And_Proce…
                                                                              Monitor_LEV0_LHIT_Signals_Update…
                                                                              Monitor_LEV0_LHIT_Signals_Update…
                                                                              Sample_ND120_CPU_Status_Signals:…
0003 00              db          0h

                 DDRA                                            XREF[8]:     Wait_For_FIFO_Commands_And_Proce…
                                                                              Process_PANC_Command_From_FIFO:0…
                                                                              Output_Response_To_ND120_IDB:09c…
                                                                              Output_Response_To_ND120_IDB:09d…
                                                                              Output_Status_Code_1_To_ND120:09…
                                                                              Output_Status_Code_1_To_ND120:0a…
                                                                              Initialize_MM58274_RTC_Chip:0ac2…
                                                                              Initialize_MM58274_RTC_Chip:0bdb…
0004 00              db          0h

                 DDRB                                            XREF[2]:     Wait_For_FIFO_Commands_And_Proce…
                                                                              Process_PANC_Command_From_FIFO:0…
0005 00              db          0h

                 DDRC                                            XREF[2]:     Wait_For_FIFO_Commands_And_Proce…
                                                                              Process_PANC_Command_From_FIFO:0…
0006 00              db          0h

                 DDRD                                            XREF[1]:     Wait_For_FIFO_Commands_And_Proce…
0007 00              db          0h

                 TDR_Timer_Data_Register                         XREF[3]:     Wait_For_FIFO_Commands_And_Proce…
                                                                              Wait_For_FIFO_Commands_And_Proce…
                                                                              Timer_1200Hz_CPU_Performance_Mon…
0008 00              db          0h

                 Timer Control Register Bits

                 b7 - TIR - Timer Interrupt Request Status (Set to 1 when time…
                 b6 - TIM - Timer Interrupt MASK (1=Interrupt inhibited, 0=Int…
                 b5 - TIN - Timer Input Select (1=External clock, 0=Internal c…
                 b4 - TIE - Timer External Input Enable (1=Enable external tim…
                 b3- PSC - Prescaler Clear (Write only. Writing 1 resets the p…
                 b2 - PS2 - Prescaler select 2
                 b1 - PS1 - Prescaler select 1
                 b0 - PS0 - Prescaler select 0

                 PS2  PS1  PS0  - Divide by
                 0    0    0       1
                 0    0    1       2
                 0    1    0       4
                 0    1    1       8
                 1    0    0       16
                 1    0    1       32
                 1    1    0       64
                 1    1    1       128

                 TCR_Timer_Control_Register                      XREF[3]:     Wait_For_FIFO_Commands_And_Proce…
                                                                              Process_PANC_Command_From_FIFO:0…
                                                                              Timer_1200Hz_CPU_Performance_Mon…
0009 00              db          0h

                 MR_Misc_register                                XREF[2]:     Wait_For_FIFO_Commands_And_Proce…
                                                                              Process_PANC_Command_From_FIFO:0…
000a 00              db          0h

                 PCR_Program_Control_Register                    XREF[1]:     Process_PANC_Command_From_FIFO:0…
000b 00              db          0h
                 NOT USED MEMORY 0x00C-0x00F

000c 00              db          0h
000d 00              db          0h
000e 00              db          0h
000f 00              db          0h
```

```
                 0x010 RAM START (112 bytes)
                 0x07F RAM END
                 RAM_0010                                     XREF[5]:     Load_Display_Character_Data_From
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Monitor_LEV0_LHIT_Signals_Update
        0010 00            db          0h

                 RAM_0011                                     XREF[6]:     Load_Display_Character_Data_From
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Monitor_LEV0_LHIT_Signals_Update
        0011 00            db          0h

                 RAM_0012                                     XREF[7]:     Load_Display_Character_Data_From
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
        0012 00            db          0h

                 RAM_0013                                     XREF[7]:     Load_Display_Character_Data_From
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
        0013 00            db          0h

                 RAM_0014                                     XREF[6]:     Format_7Segment_Display_Pattern
                                                                           Format_7Segment_Display_Pattern
                                                                           Format_7Segment_Display_Pattern
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
        0014 00            db          0h

                 RAM_0015                                     XREF[8]:     Format_7Segment_Display_Pattern
                                                                           Format_7Segment_Display_Pattern
                                                                           Format_7Segment_Display_Pattern
                                                                           Format_7Segment_Display_Pattern
                                                                           Format_7Segment_Display_Pattern
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
        0015 00            db          0h

                 RAM_0016                                     XREF[11]:    Format_7Segment_Display_Pattern
                                                                           Format_7Segment_Display_Pattern
                                                                           Format_7Segment_Display_Pattern
                                                                           Format_7Segment_Display_Pattern
                                                                           Process_Display_Data_Alternative
                                                                           Process_Display_Data_Alternative
                                                                           Process_Display_Data_Alternative
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Calculate_CPU_Utilization_Percen
        0016 00            db          0h

                 RAM_0017                                     XREF[13]:    Process_Display_Data_Alternative
                                                                           Process_Display_Data_Alternative
                                                                           Process_Display_Data_Alternative
                                                                           Process_Display_Data_Alternative
                                                                           Process_Display_Data_Alternative
                                                                           Process_Display_Data_Alternative
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Process_PANC_Command_From_FIFO:0
                                                                           Calculate_CPU_Utilization_Percen
                                                                           Calculate_CPU_Utilization_Percen
                                                                           Calculate_CPU_Utilization_Percen
        0017 00            db          0h
```

```
                RAM_0018                                       XREF[12]:    Process_Display_Data_Alternative..
                                                                            Process_Display_Data_Alternative..
                                                                            Process_Display_Data_Alternative..
                                                                            Process_Display_Data_Alternative..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Calculate_CPU_Utilization_Percen..
                                                                            Calculate_CPU_Utilization_Percen..
                                                                            Calculate_CPU_Utilization_Percen..
                                                                            Calculate_CPU_Utilization_Percen..
0018 00             db         0h

                ND120_CPU_Status_Ring_PONI_IONI                XREF[7]:     Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Update_CPU_Ring_PONI_IONI_Statis..
                                                                            Update_CPU_Ring_PONI_IONI_Statis..
                                                                            Sample_ND120_CPU_Status_Signals:..
                                                                            Sample_ND120_CPU_Status_Signals:..
                                                                            Sample_ND120_CPU_Status_Signals:..
0019 00             db         0h

                System_Configuration_Flags                     XREF[5]:     Wait_For_FIFO_Commands_And_Proce..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
001a 00             db         0h

                Display_Control_Mode_Flags                     XREF[22]:    Format_7Segment_Display_Patterns..
                                                                            Format_7Segment_Display_Patterns..
                                                                            Format_7Segment_Display_Patterns..
                                                                            Format_7Segment_Display_Patterns..
                                                                            Process_Display_Data_Alternative..
                                                                            Process_Display_Data_Alternative..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            [more]
001b 00             db         0h

                Last_PANC_Command_Byte                         XREF[10]:    Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Output_RTC_Time_Data_To_ND120:06..
                                                                            Output_RTC_Time_Data_To_ND120:06..
                                                                            Output_RTC_Time_Data_To_ND120:06..
                                                                            Output_RTC_Time_Data_To_ND120:06..
                                                                            Output_RTC_Time_Data_To_ND120:06..
                                                                            Output_RTC_Time_Data_To_ND120:06..
                                                                            Output_RTC_Time_Data_To_ND120:07..
001c 00             db         0h

                CPU_Utilization_Parameter_1                    XREF[10]:    Format_7Segment_Display_Patterns..
                                                                            Process_Display_Data_Alternative..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_PANC_Command_From_FIFO:0..
                                                                            Process_Performance_Statistics:0..
                                                                            Process_Performance_Statistics:0..
001d 00             db         0h
```

```
                 CPU_Utilization_Parameter_2              XREF[7]:     Process_PANC_Command_From_FIFO:0…
                                                                       Process_PANC_Command_From_FIFO:0…
                                                                       Process_PANC_Command_From_FIFO:0…
                                                                       Process_PANC_Command_From_FIFO:0…
                                                                       Process_PANC_Command_From_FIFO:0…
                                                                       Process_PANC_Command_From_FIFO:0…
                                                                       Calculate_CPU_Utilization_Percen…
001e 00              db          0h

                 FIFO_Empty_Timeout_Counter               XREF[2]:     Wait_For_FIFO_Commands_And_Proce…
                                                                       Wait_For_FIFO_Commands_And_Proce…
001f 00              db          0h

                 Software_RTC_Tenths_Seconds              XREF[15]:    Format_7Segment_Display_Patterns…
                                                                       Format_7Segment_Display_Patterns…
                                                                       Convert_Time_Offset_To_Software_…
                                                                       Convert_Time_Offset_To_Software_…
                                                                       Convert_Time_Offset_To_Software_…
                                                                       Calculate_RTC_Date_Time_Offset:0…
                                                                       Calculate_RTC_Date_Time_Offset:0…
                                                                       Calculate_RTC_Date_Time_Offset:0…
                                                                       Calculate_RTC_Date_Time_Offset:0…
                                                                       Calculate_RTC_Date_Time_Offset:0…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Output_Status_Code_1_To_ND120:0a…
                                                                       Initialize_MM58274_RTC_Chip:0ae8…
                                                                       Initialize_MM58274_RTC_Chip:0aff…
                                                                       Initialize_MM58274_RTC_Chip:0b01…
0020 00              db          0h

                 Software_RTC_Units_Seconds               XREF[14]:    Format_7Segment_Display_Patterns…
                                                                       Format_7Segment_Display_Patterns…
                                                                       Convert_Time_Offset_To_Software_…
                                                                       Convert_Time_Offset_To_Software_…
                                                                       Convert_Time_Offset_To_Software_…
                                                                       Calculate_RTC_Date_Time_Offset:0…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Output_Status_Code_1_To_ND120:0a…
                                                                       Initialize_MM58274_RTC_Chip:0b0f…
                                                                       Initialize_MM58274_RTC_Chip:0b26…
                                                                       Initialize_MM58274_RTC_Chip:0b28…
0021 00              db          0h

                 Software_RTC_Tens_Seconds                XREF[9]:     Convert_Time_Offset_To_Software_…
                                                                       Calculate_RTC_Date_Time_Offset:0…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Output_Status_Code_1_To_ND120:0a…
                                                                       Initialize_MM58274_RTC_Chip:0b36…
                                                                       Initialize_MM58274_RTC_Chip:0b4d…
                                                                       Initialize_MM58274_RTC_Chip:0b4f…
0022 00              db          0h

                 Software_RTC_Units_Minutes               XREF[10]:    Convert_Time_Offset_To_Software_…
                                                                       Convert_Time_Offset_To_Software_…
                                                                       Calculate_RTC_Date_Time_Offset:0…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Output_Status_Code_1_To_ND120:0a…
                                                                       Initialize_MM58274_RTC_Chip:0b5d…
                                                                       Initialize_MM58274_RTC_Chip:0b74…
                                                                       Initialize_MM58274_RTC_Chip:0b76…
0023 00              db          0h

                 Software_RTC_Tens_Minutes                XREF[12]:    Format_7Segment_Display_Patterns…
                                                                       Format_7Segment_Display_Patterns…
                                                                       Convert_Time_Offset_To_Software_…
                                                                       Convert_Time_Offset_To_Software_…
                                                                       Calculate_RTC_Date_Time_Offset:0…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Increment_Software_RTC_Counters:…
                                                                       Output_Status_Code_1_To_ND120:0a…
                                                                       Initialize_MM58274_RTC_Chip:0b84…
                                                                       Initialize_MM58274_RTC_Chip:0b9b…
                                                                       Initialize_MM58274_RTC_Chip:0b9d…
0024 00              db          0h
```

```
                Software_RTC_Units_Hours                    XREF[13]:    Format_7Segment_Display_Patterns…
                                                                         Format_7Segment_Display_Patterns…
                                                                         Convert_Time_Offset_To_Software_…
                                                                         Convert_Time_Offset_To_Software_…
                                                                         Calculate_RTC_Date_Time_Offset:0…
                                                                         Increment_Software_RTC_Counters:…
                                                                         Increment_Software_RTC_Counters:…
                                                                         Increment_Software_RTC_Counters:…
                                                                         Output_Status_Code_1_To_ND120:0a…
                                                                         Output_Status_Code_1_To_ND120:0a…
                                                                         Initialize_MM58274_RTC_Chip:0bab…
                                                                         Initialize_MM58274_RTC_Chip:0bc2…
                                                                         Initialize_MM58274_RTC_Chip:0bc4…
0025 00             db         0h

                MM58274_Tens_Hours                          XREF[16]:    Convert_Time_Offset_To_Software_…
                                                                         Convert_Time_Offset_To_Software_…
                                                                         Convert_Time_Offset_To_Software_…
                                                                         Convert_Time_Offset_To_Software_…
                                                                         Convert_Time_Offset_To_Software_…
                                                                         Calculate_RTC_Date_Time_Offset:0…
                                                                         Calculate_RTC_Date_Time_Offset:0…
                                                                         Calculate_RTC_Date_Time_Offset:0…
                                                                         Calculate_RTC_Date_Time_Offset:0…
                                                                         Calculate_RTC_Date_Time_Offset:0…
                                                                         Increment_Software_RTC_Counters:…
                                                                         Increment_Software_RTC_Counters:…
                                                                         Increment_Software_RTC_Counters:…
                                                                         Increment_Software_RTC_Counters:…
                                                                         Output_Status_Code_1_To_ND120:0a…
                                                                         Initialize_MM58274_RTC_Chip:0ada…
0026 00             db         0h

                Display_Working_Buffer_Start
0027 00             db         0h
0028 00             db         0h
0029 00             db         0h
002a 00             db         0h
002b 00             db         0h
002c 00             db         0h
002d 00             db         0h
002e 00             db         0h
002f 00             db         0h
0030 00             db         0h
0031 00             db         0h
0032 00             db         0h
0033 00             db         0h
0034 00             db         0h
0035 00             db         0h
0036 00             db         0h
0037 00             db         0h
0038 00             db         0h
0039 00             db         0h
003a 00             db         0h
003b 00             db         0h
003c 00             db         0h
003d 00             db         0h
003e 00             db         0h
003f 00             db         0h
0040 00             db         0h
0041 00             db         0h
0042 00             db         0h

                BYTE_0043                                   XREF[1]:     Delay_And_Sample_Port_C:04bd(R)
0043 00             db         0h

                BYTE_0044                                   XREF[1]:     Delay_And_Sample_Port_C:04bd(R)
0044 00             db         0h

                BYTE_0045                                   XREF[3]:     Wait_For_FIFO_Commands_And_Proce…
                                                                         Delay_And_Sample_Port_C:04b4(R),
                                                                         Delay_And_Sample_Port_C:04c8(R)
0045 00             db         0h

                PortC_ValueToWrite                          XREF[5]:     Wait_For_FIFO_Commands_And_Proce…
                                                                         Process_7Segment_Display_Bit_Man…
                                                                         Process_7Segment_Display_Bit_Man…
                                                                         Delay_And_Sample_Port_C:04ab(R),
                                                                         Delay_And_Sample_Port_C:04c8(R)
0046 00             db         0h
```

```
                    Date_Offset_High_Byte                       XREF[3]:     Convert_Time_Offset_To_Software…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
        0047 00          db          0h

                    Date_Offset_Low_Byte                        XREF[4]:     Convert_Time_Offset_To_Software…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
        0048 00          db          0h

                    Time_Offset_High_Byte                       XREF[2]:     Convert_Time_Offset_To_Software…
                                                                             Calculate_RTC_Date_Time_Offset:0…
        0049 00          db          0h

                    Time_Offset_Low_Byte                        XREF[2]:     Convert_Time_Offset_To_Software…
                                                                             Calculate_RTC_Date_Time_Offset:0…
        004a 00          db          0h

                    Display_Bit_Mask_Parameter                  XREF[17]:    Wait_For_FIFO_Commands_And_Proce…
                                                                             Wait_For_FIFO_Commands_And_Proce…
                                                                             Wait_For_FIFO_Commands_And_Proce…
                                                                             Process_7Segment_Display_Bit_Ma…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
        004b 00          db          0h

                    Display_Bit_Shift_Parameter                 XREF[24]:    Wait_For_FIFO_Commands_And_Proce…
                                                                             Wait_For_FIFO_Commands_And_Proce…
                                                                             Wait_For_FIFO_Commands_And_Proce…
                                                                             Process_7Segment_Display_Bit_Ma…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             [more]
        004c 00          db          0h

                    Calculation_Working_High                    XREF[29]:    Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Convert_Time_Offset_To_Software…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
                                                                             [more]
        004d 00          db          0h
```

```
                Calculation_Working_Low                        XREF[31]:    Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Calculate_RTC_Date_Time_Offset:0…
                                                                            Calculate_RTC_Date_Time_Offset:0…
                                                                            Calculate_RTC_Date_Time_Offset:0…
                                                                            [more]
004e 00             db         0h

                Display_Buffer_Index_Counter                   XREF[11]:    Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Process_7Segment_Display_Bit_Ma…
                                                                            Process_7Segment_Display_Bit_Ma…
                                                                            Process_7Segment_Display_Bit_Ma…
                                                                            Calculate_RTC_Date_Time_Offset:0…
                                                                            Calculate_RTC_Date_Time_Offset:0…
                                                                            Calculate_RTC_Date_Time_Offset:0…
004f 00             db         0h

                Display_Bit_Loop_Counter                       XREF[2]:     Process_7Segment_Display_Bit_Ma…
                                                                            Process_7Segment_Display_Bit_Ma…
0050 00             db         0h

                Display_Position_Loop_Counter                  XREF[20]:    Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Process_Display_Data_Alternativ…
                                                                            Process_Display_Data_Alternativ…
                                                                            Process_Display_Data_Alternativ…
                                                                            Process_Display_Data_Alternativ…
                                                                            Process_Display_Data_Alternativ…
                                                                            Process_Display_Data_Alternativ…
                                                                            Process_7Segment_Display_Bit_Ma…
                                                                            Process_7Segment_Display_Bit_Ma…
                                                                            Process_7Segment_Display_Bit_Ma…
                                                                            Process_7Segment_Display_Bit_Ma…
0051 00             db         0h

                Display_7Segment_Digit_0                       XREF[37]:    Load_Display_Character_Data_Fro…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Format_7Segment_Display_Pattern…
                                                                            Process_Display_Data_Alternativ…
                                                                            Process_Display_Data_Alternativ…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Convert_Time_Offset_To_Software_…
                                                                            Calculate_RTC_Date_Time_Offset:0…
                                                                            Calculate_RTC_Date_Time_Offset:0…
                                                                            Initialize_MM58274_RTC_Chip:0afa…
                                                                            Initialize_MM58274_RTC_Chip:0afd…
                                                                            Initialize_MM58274_RTC_Chip:0b21…
                                                                            Initialize_MM58274_RTC_Chip:0b24…
                                                                            [more]
0052 00             db         0h
```

```
                      Display_7Segment_Digit_1                       XREF[15]:     Load_Display_Character_Data_From…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Convert_Time_Offset_To_Software_…
                                                                                   Convert_Time_Offset_To_Software_…
                                                                                   Convert_Time_Offset_To_Software_…
                                                                                   Convert_Time_Offset_To_Software_…
                                                                                   Convert_Time_Offset_To_Software_…
                                                                                   Convert_Time_Offset_To_Software_…
                                                                                   Convert_Time_Offset_To_Software_…
                                                                                   Convert_Time_Offset_To_Software_…
        0053 00              db         0h

                      Display_7Segment_Digit_2                       XREF[9]:      Load_Display_Character_Data_From…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Process_Display_Data_Alternative…
        0054 00              db         0h

                      Display_7Segment_Digit_3                       XREF[11]:     Load_Display_Character_Data_From…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Process_7Segment_Display_Bit_Man…
        0055 00              db         0h

                      Display_7Segment_Digit_4                       XREF[6]:      Load_Display_Character_Data_From…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Process_Display_Data_Alternative…
        0056 00              db         0h

                      Display_7Segment_Digit_5                       XREF[6]:      Load_Display_Character_Data_From…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Process_Display_Data_Alternative…
        0057 00              db         0h

                      Display_7Segment_Digit_6                       XREF[9]:      Load_Display_Character_Data_From…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Process_7Segment_Display_Bit_Man…
        0058 00              db         0h

                      Display_7Segment_Digit_7                       XREF[10]:     Load_Display_Character_Data_From…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Format_7Segment_Display_Patterns…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Process_Display_Data_Alternative…
                                                                                   Process_7Segment_Display_Bit_Man…
        0059 00              db         0h
```

```
                     PortC_SavedValue                        XREF[18]:    Format_7Segment_Display_Pattern…
                                                                          Format_7Segment_Display_Pattern…
                                                                          Format_7Segment_Display_Pattern…
                                                                          Format_7Segment_Display_Pattern…
                                                                          Format_7Segment_Display_Pattern…
                                                                          Format_7Segment_Display_Pattern…
                                                                          Format_7Segment_Display_Pattern…
                                                                          Format_7Segment_Display_Pattern…
                                                                          Process_Display_Data_Alternative…
                                                                          Process_Display_Data_Alternative…
                                                                          Process_Display_Data_Alternative…
                                                                          Process_Display_Data_Alternative…
                                                                          Process_Display_Data_Alternative…
                                                                          Process_Display_Data_Alternative…
                                                                          Process_Display_Data_Alternative…
                                                                          Process_Display_Data_Alternative…
                                                                          Delay_And_Sample_Port_C:04a9(W),
                                                                          Delay_And_Sample_Port_C:04d3(R)
        005a 00          db         0h

                     Timer_Counter_16_Cycles_13ms            XREF[3]:     Wait_For_FIFO_Commands_And_Proc…
                                                                          Timer_1200Hz_CPU_Performance_Mo…
                                                                          Timer_1200Hz_CPU_Performance_Mo…
        005b 00          db         0h

                     Timer_Counter_200_Cycles_167ms          XREF[3]:     Wait_For_FIFO_Commands_And_Proc…
                                                                          Timer_1200Hz_CPU_Performance_Mo…
                                                                          Timer_1200Hz_CPU_Performance_Mo…
        005c 00          db         0h

                     Timer_Counter_2_Cycles_333ms            XREF[3]:     Wait_For_FIFO_Commands_And_Proc…
                                                                          Timer_1200Hz_CPU_Performance_Mo…
                                                                          Timer_1200Hz_CPU_Performance_Mo…
        005d 00          db         0h

                     CPU_Utilization_Buffer_Input            XREF[1]:     Shift_CPU_Utilization_History_Bu…
        005e 00          db         0h

                     BYTE_005f                               XREF[1]:     Shift_CPU_Utilization_History_Bu…
        005f 00          db         0h

                     CPU_Status_Accumulator                  XREF[3]:     Update_CPU_Ring_PONI_IONI_Stati…
                                                                          Update_CPU_Ring_PONI_IONI_Stati…
                                                                          Shift_CPU_Utilization_History_Bu…
        0060 00          db         0h

                     CPU_Utilization_History_Buffer_Start    XREF[2]:     Shift_CPU_Utilization_History_Bu…
                                                                          Shift_CPU_Utilization_History_Bu…
        0061 00          db         0h

                     BYTE_0062                               XREF[2]:     Shift_CPU_Utilization_History_Bu…
                                                                          Shift_CPU_Utilization_History_Bu…
        0062 00          db         0h

                     BYTE_0063                               XREF[2]:     Shift_CPU_Utilization_History_Bu…
                                                                          Shift_CPU_Utilization_History_Bu…
        0063 00          db         0h

                     BYTE_0064                               XREF[2]:     Shift_CPU_Utilization_History_Bu…
                                                                          Shift_CPU_Utilization_History_Bu…
        0064 00          db         0h

                     BYTE_0065                               XREF[2]:     Shift_CPU_Utilization_History_Bu…
                                                                          Shift_CPU_Utilization_History_Bu…
        0065 00          db         0h

                     BYTE_0066                               XREF[2]:     Shift_CPU_Utilization_History_Bu…
                                                                          Shift_CPU_Utilization_History_Bu…
        0066 00          db         0h

                     BYTE_0067                               XREF[2]:     Shift_CPU_Utilization_History_Bu…
                                                                          Shift_CPU_Utilization_History_Bu…
        0067 00          db         0h

                     BYTE_0068                               XREF[2]:     Shift_CPU_Utilization_History_Bu…
                                                                          Shift_CPU_Utilization_History_Bu…
        0068 00          db         0h

                     BYTE_0069                               XREF[2]:     Shift_CPU_Utilization_History_Bu…
                                                                          Shift_CPU_Utilization_History_Bu…
        0069 00          db         0h
```

```
                BYTE_006a                                      XREF[2]:     Shift_CPU_Utilization_History_Bu…
                                                                            Shift_CPU_Utilization_History_Bu…
0060a 00            db          0h

                BYTE_006b                                      XREF[2]:     Shift_CPU_Utilization_History_Bu…
                                                                            Shift_CPU_Utilization_History_Bu…
006b 00             db          0h

                BYTE_006c                                      XREF[2]:     Shift_CPU_Utilization_History_Bu…
                                                                            Shift_CPU_Utilization_History_Bu…
006c 00             db          0h

                BYTE_006d                                      XREF[1]:     Shift_CPU_Utilization_History_Bu…
006d 00             db          0h

                BYTE_006e                                      XREF[1]:     Shift_CPU_Utilization_History_Bu…
006e 00             db          0h

                CPU_Utilization_History_Buffer_End             XREF[2]:     Update_CPU_Ring_PONI_IONI_Statis…
                                                                            Shift_CPU_Utilization_History_Bu…
006f 00             db          0h

                PortD_SavedValue                               XREF[3]:     Sample_ND120_CPU_Status_Signals:…
                                                                            Sample_ND120_CPU_Status_Signals:…
                                                                            Sample_ND120_CPU_Status_Signals:…
0070 00             db          0h

                LEV0_LHIT_Sample_Counter_128                   XREF[3]:     Wait_For_FIFO_Commands_And_Proce…
                                                                            Monitor_LEV0_LHIT_Signals_Update…
                                                                            Monitor_LEV0_LHIT_Signals_Update…
0071 00             db          0h

                LHIT_Cache_Hit_Accumulator                     XREF[3]:     Monitor_LEV0_LHIT_Signals_Update…
                                                                            Monitor_LEV0_LHIT_Signals_Update…
                                                                            Monitor_LEV0_LHIT_Signals_Update…
0072 00             db          0h

                LEV0_CPU_Busy_Time_Accumulator                 XREF[3]:     Monitor_LEV0_LHIT_Signals_Update…
                                                                            Monitor_LEV0_LHIT_Signals_Update…
                                                                            Monitor_LEV0_LHIT_Signals_Update…
0073 00             db          0h
0074 00             db          0h
0075 00             db          0h
0076 00             db          0h
0077 00             db          0h
0078 00             db          0h
0079 00             db          0h
007a 00             db          0h
007b 00             db          0h
007c 00             db          0h
007d 00             db          0h
007e 00             db          0h
007f 00             db          0h

                RAM ENDS AT 0x007F
                *****
                ROM START AT 0x0080
                ENDS AT 0x0FFF
                RTC_Leap_Year_Constant_High                    XREF[2]:     Convert_Time_Offset_To_Software_…
                                                                            Calculate_RTC_Date_Time_Offset:0…
0080 02             db          2h

                RTC_Leap_Year_Constant_Low                     XREF[2]:     Convert_Time_Offset_To_Software_…
                                                                            Calculate_RTC_Date_Time_Offset:0…
0081 da             db          DAh

                RTC_Hour_Constant_High                         XREF[1]:     Convert_Time_Offset_To_Software_…
0082 0e             db          Eh

                RTC_Hour_Constant_Low                          XREF[1]:     Convert_Time_Offset_To_Software_…
0083 10             db          10h
0084 07             db          7h
0085 bb             db          BBh
0086 00             db          0h
0087 10             db          10h
0088 20             db          20h
0089 30             db          30h
008a 40             db          40h
008b 50             db          50h
008c 60             db          60h
008d 70             db          70h

                BYTE_008e                                      XREF[1]:     Wait_For_FIFO_Commands_And_Proce…
008e 00             db          0h
```

```
                        BYTE_008f                                       XREF[1]:     Wait_For_FIFO_Commands_And_Proce
008f 10              db         10h
0090 20              db         20h
0091 30              db         30h
0092 40              db         40h
0093 50              db         50h
0094 60              db         60h
0095 70              db         70h
0096 00              db         0h
0097 10              db         10h
0098 20              db         20h
0099 30              db         30h
009a 40              db         40h
009b 50              db         50h
009c 60              db         60h
009d 70              db         70h
009e 00              db         0h
009f 10              db         10h
00a0 20              db         20h
00a1 30              db         30h
00a2 40              db         40h
00a3 50              db         50h
00a4 60              db         60h
00a5 70              db         70h
00a6 00              db         0h
00a7 10              db         10h
00a8 20              db         20h
00a9 30              db         30h
00aa 40              db         40h
00ab 50              db         50h
00ac 60              db         60h
00ad 70              db         70h
00ae 00              db         0h
00af 10              db         10h
00b0 20              db         20h
00b1 30              db         30h
00b2 40              db         40h
00b3 50              db         50h
00b4 60              db         60h
00b5 70              db         70h
00b6 00              db         0h
00b7 10              db         10h
00b8 20              db         20h
00b9 30              db         30h
00ba 40              db         40h
00bb 50              db         50h
00bc 60              db         60h
00bd 70              db         70h
00be 00              db         0h
00bf 10              db         10h
00c0 20              db         20h
00c1 30              db         30h
00c2 40              db         40h
00c3 50              db         50h
00c4 60              db         60h
00c5 70              db         70h
00c6 00              db         0h
00c7 10              db         10h
00c8 20              db         20h
00c9 30              db         30h
00ca 40              db         40h
00cb 50              db         50h
00cc 60              db         60h
00cd 70              db         70h
00ce 00              db         0h
00cf 10              db         10h
00d0 20              db         20h
00d1 30              db         30h
00d2 40              db         40h
00d3 50              db         50h
00d4 60              db         60h
00d5 70              db         70h
00d6 00              db         0h
00d7 10              db         10h
00d8 20              db         20h
00d9 30              db         30h
00da 40              db         40h
00db 50              db         50h
00dc 60              db         60h
00dd 70              db         70h
00de 00              db         0h
00df 10              db         10h
00e0 20              db         20h
00e1 30              db         30h
```

```
00e2 40          db        40h
00e3 50          db        50h
00e4 60          db        60h
00e5 70          db        70h
00e6 00          db        0h
00e7 10          db        10h
00e8 20          db        20h
00e9 30          db        30h
00ea 40          db        40h
00eb 50          db        50h
00ec 60          db        60h
00ed 70          db        70h
00ee 00          db        0h
00ef 10          db        10h
00f0 20          db        20h
00f1 30          db        30h
00f2 40          db        40h
00f3 50          db        50h
00f4 60          db        60h
00f5 70          db        70h
00f6 00          db        0h
00f7 10          db        10h
00f8 20          db        20h
00f9 30          db        30h
00fa 40          db        40h
00fb 50          db        50h
00fc 60          db        60h
00fd 70          db        70h

            Debug_Trace_Address_Marker             XREF[9]:    Wait_For_FIFO_Commands_And_Proce..
                                                               Wait_For_FIFO_Commands_And_Proce..
                                                               Wait_For_FIFO_Commands_And_Proce..
                                                               Wait_For_FIFO_Commands_And_Proce..
                                                               Wait_For_FIFO_Commands_And_Proce..
                                                               Wait_For_FIFO_Commands_And_Proce..
                                                               Wait_For_FIFO_Commands_And_Proce..
                                                               Wait_For_FIFO_Commands_And_Proce..
                                                               Wait_For_FIFO_Commands_And_Proce..
00fe 00          db        0h
00ff 10          db        10h
0100 20          db        20h
0101 30          db        30h
0102 40          db        40h
0103 50          db        50h
0104 60          db        60h
0105 70          db        70h
0106 00          db        0h
0107 00          db        0h
0108 00          db        0h
0109 00          db        0h
010a 00          db        0h
010b 00          db        0h
010c 00          db        0h
010d 00          db        0h
010e 00          db        0h
010f 00          db        0h
```

```
                  **************************************************************
                  *                          FUNCTION                          *
                  **************************************************************
                  undefined Wait_For_FIFO_Commands_And_Process()
undefined           <UNASSIGNED>   <RETURN>
byte                  X:1            bVar2                               XREF[1]:     01a5(W)
byte                  X:1            bVar3                               XREF[1]:     01a8(W)
                  ND-120 Performance Monitor - Main System Entry Point
                  MC68705 Panel/Calendar Controller with Real-Time Performance ...

                  =============================================================...
                  SYSTEM OVERVIEW
                  =============================================================...

                  HARDWARE ARCHITECTURE:
                  - MC68705 CPU: 4MHz crystal (main execution)
                  - Timer Clock: 38.4kHz PANOSC (ND-120 clock ÷ 1024)
                  - Timer Interrupt: 1200Hz (38.4kHz ÷ 32 prescaler)
                  - Sample Period: 833μs for real-time CPU monitoring

                  DUAL-CLOCK DESIGN BENEFITS:
                  - 4MHz: Fast command processing and display updates
                  - 38.4kHz: Precise timing for performance monitoring
                  - Independent domains prevent timing conflicts

                  =============================================================...
                  ND-120 CPU INTERFACE (MICROCODE LEVEL)
                  =============================================================...

                  COMMAND INTERFACE (ND-120 → MC68705):
                  LDPANC Microcode 06.2:
                  - Loads lower byte of IDB to MC68705 FIFO queue
                  - Sets /EMP_n HIGH indicating FIFO contains data
                  - Building block for TRR PANC high-level instruction

                  RESPONSE INTERFACE (MC68705 → ND-120):
                  EPANS Microcode (IDB Sources 20/21):
                  - IDB Source 20: Read MC68705 Port A, set /EMP_n HIGH
                  - IDB Source 21: Same as 20 + reset PANS bit 12 (RDY)
                  - Building block for TRA PANS high-level instruction

                  =============================================================...
                  PANS/PANC REGISTER INTERFACE
                  =============================================================...

                  PANS (Panel Status Register) - Read by "TRA PANS":
                  Bit 15: PAN - Panel installed (1=present)
                  Bit 14: FIP - FIFO ready for data
                  Bit 13: DAT - Last command requested data
                  Bit 12: RDY - Command completed (cleared by TRA PANS)
                  Bits 10-8: cmnd - Last command processed (0-7)
                  Bits 7-0: RPAN - Response data from MC68705

                  PANC (Panel Control Register) - Written by "TRR PANC":
                  Bit 13: DAT - Command requests data from panel
                  Bits 10-8: cmnd - Panel command type (0-7)
                  Bits 7-0: WPAN - Data to panel processor

                  =============================================================...
                  REAL-TIME PERFORMANCE MONITORING
                  =============================================================...

                  PORT D SIGNAL MONITORING (1200Hz sampling):
                  PD7: /EMP_n - FIFO status + Panel enable (dual function)
                  PD6: GND - Tied to ground
                  PD5: LEV0 - CPU Level 0 (HIGH=idle, LOW=busy levels 1-15)
                  PD4: LHIT - Cache hit (HIGH=hit, LOW=miss)
                  PD3: IONI - Interrupt system ON (SINTRAN OS control)
                  PD2: PONI - Memory protection ON (SINTRAN OS control)
                  PD[1:0]: PCR Ring (0-3) CPU privilege level

                  CPU UTILIZATION CALCULATION:
                  - Ring 0: User programs and idle time
                  - Ring 1-3: System, kernel, supervisor operations
                  - LEV0 signal: Direct idle/busy indication
                  - LHIT signal: Cache efficiency measurement
                  - Statistical processing: 15-sample sliding window

                  =============================================================...
                  PERFORMANCE METRICS GENERATED
                  =============================================================...

                  REAL-TIME MEASUREMENTS:
                  1. CPU Utilization: % time NOT in Level 0 (busy percentage)
                  2. Ring Distribution: Time spent in each privilege level
                  3. Cache Efficiency: Hit rate from LHIT signal sampling
                  4. System Load: Ring 2-3 activity (kernel/supervisor time)
```

```
0111 ae 7f          LDX        #0x7f

                    LAB_0113                                XREF[1]:     0116(j)
0113 6f 10              CLR        0x10,X=>BYTE_008f                            = 10h
0115 5a                 DECX
0116 2a fb              BPL        LAB_0113
0118 a6 10              LDA        #0x10
011a b7 5b              STA        Timer_Counter_16_Cycles_13ms
011c a6 c8              LDA        #0xc8
011e b7 5c              STA        Timer_Counter_200_Cycles_167ms
0120 a6 02              LDA        #0x2
0122 b7 5d              STA        Timer_Counter_2_Cycles_333ms
0124 a6 80              LDA        #0x80
0126 b7 71              STA        LEV0_LHIT_Sample_Counter_128
0128 3f 04              CLR        DDRA
012a a6 2f              LDA        #0x2f
012c b7 01              STA        PORTB
012e a6 ff              LDA        #0xff
0130 b7 05              STA        DDRB
0132 a6 f8              LDA        #0xf8
0134 b7 02              STA        PORTC
0136 a6 ff              LDA        #0xff
0138 b7 06              STA        DDRC
013a 3f 07              CLR        DDRD
013c cd 0a c0           JSR        Initialize_MM58274_RTC_Chip                   undefinedInitialize_MM58274_RTC…
                    Set  Timer Control register value 0x35

                    TIM=0  => Timer Interrupt Enable
                    TIN=1  => Timer Input Select External Clock
                    TIE=1  => Timer External Input Enable
                    PS=101 => Prescaler divide by 32.

                    --- DESIGN INFO --- (migh be wrong, but based on schemas)
                    PANOSC will only oscilate (as a function of RTOSC) when CLOSC…
                    PANOSC = RTOSC / 4.
                    RTOSC = 153.6Khz
                    PANOSC = RTOSC/4 = 38400 HZ

                    With prescale divider on 32, the timer interrupt should be 12…

                    68705 CPU is running on 4MHZ, so that means an Timer interrup…
013f a6 35              LDA        #0x35
0141 b7 09              STA        TCR_Timer_Control_Register
0143 a6 03              LDA        #0x3
0145 b7 08              STA        TDR_Timer_Data_Register
0147 a6 7f              LDA        #0x7f
0149 b7 0a              STA        MR_Misc_register
014b 9a                 CLI
014c 3f 1a              CLR        System_Configuration_Flags
```

```
                 MAIN LOOP: Monitor /EMP_n for FIFO and Panel Status Operations
                 Responds to both command loading (LDPANC) and status reading …

                 /EMP_n DUAL FUNCTION MONITORING:
                 The /EMP_n signal (Port D pin 7) serves two purposes:
                 1. FIFO Status: HIGH when LDPANC microcode loads commands
                 2. Panel Enable: HIGH when EPANS microcode reads responses

                 IDLE STATE - Waiting for ND-120 Microcode:
                 1. Set FIFO_Empty_Timeout_Counter = 0xFF (255 cycles)
                 2. Clear PORTB bit 4 (signal ready)
                 3. Poll /EMP_n continuously:
                    - /EMP_n HIGH: ND-120 microcode active → goto ACTIVE
                    - /EMP_n LOW: No microcode activity → continue polling
                 4. Timeout and restart (prevents infinite blocking)

                 ND-120 MICROCODE OPERATIONS that trigger /EMP_n HIGH:

                 COMMAND LOADING (LDPANC microcode 06.2):
                 - ND-120 executes LDPANC microcode
                 - Lower byte of IDB loaded to MC68705 FIFO
                 - /EMP_n set HIGH to indicate FIFO has data
                 - MC68705 enters ACTIVE state to process command

                 RESPONSE READING (EPANS microcode):
                 - IDB Source 20: Read Panel Status (sets /EMP_n HIGH)
                 - IDB Source 21: Read Panel Status + reset RDY bit
                 - MC68705 must output response data via Port A
                 - Response flows: Port A → IDB → ND-120 register

                 ACTIVE STATE - Responding to Microcode:
                 1. Process commands from FIFO (if LDPANC triggered /EMP_n)
                 2. Output response data via Port A (if EPANS triggered /EMP_n)
                 3. Multi-phase display and timing processing
                 4. Continue while /EMP_n HIGH (microcode still active)

                 TIMING COORDINATION:
                 - LDPANC: Loads command, MC68705 processes in background
                 - EPANS: ND-120 waits for MC68705 response on Port A
                 - /EMP_n provides handshaking between microcode and MC68705

                 MICROCODE ADVANTAGES:
                 - Low-level control enables precise timing
                 - Hardware FIFO buffers commands for asynchronous processing
                 - Status reading can interrupt command processing if needed
                 - Efficient integration with ND-120 instruction execution
                 LAB_014e                                         XREF[2]:     015f(j), 01b1(j)
014e a6 ff           LDA        #0xff
0150 b7 1f           STA        FIFO_Empty_Timeout_Counter
0152 9b              SEI
0153 18 01           BSET       0x4,PORTB
0155 19 01           BCLR       0x4,PORTB
0157 9a              CLI

                 Wait for signal "/EMP" to go HIGH on port D pin 7.
                 LAB_0158                                         XREF[1]:     015d(j)
0158 0e 03 06        BRSET      0x7,PORTD,LAB_0161
015b 3a 1f           DEC        FIFO_Empty_Timeout_Counter
015d 26 f9           BNE        LAB_0158
015f 20 ed           BRA        LAB_014e
```

```
                   ACTIVE STATE: Respond to ND-120 Microcode Operations
                   Handles both command processing (LDPANC) and status reading (…

                   /EMP_n SIGNAL DUAL FUNCTION:
                   Port D pin 7 (/EMP_n) serves two distinct microcode operation…

                   1. FIFO DATA AVAILABLE (from LDPANC microcode 06.2):
                      - ND-120 executes LDPANC to load command/data to FIFO
                      - /EMP_n set HIGH indicating FIFO contains data
                      - MC68705 processes commands via Process_PANC_Command_From…

                   2. PANEL STATUS ENABLE (from EPANS microcode):
                      - IDB Source 20: Enable panel status reading
                      - IDB Source 21: Enable panel status + reset RDY flag
                      - /EMP_n set HIGH to enable Port A → IDB data flow
                      - MC68705 must output response via Output_Response_To_ND12…

                   PROCESSING PIPELINE (when /EMP_n HIGH):
                   1. Command Processing:
                      - Read FIFO data using Strobe_WMM_Read_From_FIFO()
                      - Execute command logic based on type (0-7)
                      - Generate response data for later EPANS reading

                   2. Working Buffer Management:
                      - Clear RAM 0x27-0x46 for fresh processing
                      - Prevent data contamination between commands

                   3. Display Processing Pipeline:
                     - Load_Display_Character_Data_From_ROM(): Character patter…
                      - Multi-phase bit manipulation and 7-segment formatting
                      - Update panel display based on command results

                   4. Hardware Interfaces:
                      - MM58274 RTC communication via Port C
                      - Timer synchronization with 1200Hz performance monitoring
                      - CPU statistics exchange and accumulation

                   5. Response Generation:
                      - Format data for ND-120 consumption
                      - Output via Port A for EPANS microcode reading
                      - Update status flags and completion indicators

                   MICROCODE TIMING COORDINATION:
                   - LDPANC can queue multiple commands in FIFO
                   - EPANS can read responses at any time (even during processin…
                   - /EMP_n provides handshaking between microcode and MC68705
                   - Hardware buffering enables asynchronous operation

                   LOOP CONTINUATION:
                   Process commands while /EMP_n remains HIGH
                   - Multiple LDPANC operations keep /EMP_n active
                   - EPANS operations may also maintain /EMP_n
                   - Only exit when ND-120 microcode activity ceases

                   INTER-OPERATION DELAY:
                   320 clock cycles between operations provides:
                   - Settling time for hardware state changes
                   - Prevention of bus conflicts during direction changes
                   - Proper setup/hold times for next microcode operation
                   LAB_0161                                       XREF[2]:     0158(j), 01a2(j)
      0161 cd 04 d8       JSR        Process_PANC_Command_From_FIFO              undefinedProcess_PANC_Command_F…
      0164 ae 1f          LDX        #0x1f

                   LAB_0166                                       XREF[1]:     0169(j)
      0166 6f 27          CLR        0x27,X=>PortC_ValueToWrite
      0168 5a             DECX
      0169 2a fb          BPL        LAB_0166
      016b cd 01 b4       JSR        Load_Display_Character_Data_From_ROM        undefinedLoad_Display_Character…
      016e a6 40          LDA        #0x40
      0170 b7 4b          STA        Display_Bit_Mask_Parameter
      0172 a6 04          LDA        #0x4
      0174 b7 4c          STA        Display_Bit_Shift_Parameter
      0176 cd 04 79       JSR        Process_7Segment_Display_Bit_Manipulation   undefinedProcess_7Segment_Displ…
      0179 cd 01 ed       JSR        Format_7Segment_Display_Patterns            undefinedFormat_7Segment_Displa…
      017c a6 20          LDA        #0x20
      017e b7 4b          STA        Display_Bit_Mask_Parameter
      0180 a6 02          LDA        #0x2
      0182 b7 4c          STA        Display_Bit_Shift_Parameter
      0184 cd 04 79       JSR        Process_7Segment_Display_Bit_Manipulation   undefinedProcess_7Segment_Displ…
      0187 cd 03 7b       JSR        Process_Display_Data_Alternative_Mode       undefinedProcess_Display_Data_A…
      018a a6 10          LDA        #0x10
      018c b7 4b          STA        Display_Bit_Mask_Parameter
      018e a6 01          LDA        #0x1
      0190 b7 4c          STA        Display_Bit_Shift_Parameter
```

```
0192 cd 04 79        JSR        Process_7Segment_Display_Bit_Manipulation      undefinedProcess_7Segment_Displ…
0195 1b 01           BCLR       0x5,PORTB

                LAB_0197                                        XREF[1]:     019b(j)
0197 b6 08           LDA        TDR_Timer_Data_Register
0199 a1 03           CMP        #0x3
019b 26 fa           BNE        LAB_0197
019d 9b              SEI
019e cd 04 a7        JSR        Delay_And_Sample_Port_C                         voidDelay_And_Sample_Port_C(void)
01a1 9a              CLI
01a2 0e 03 bc        BRSET      0x7,PORTD,LAB_0161
                Wait for 320 Clock Ticks.
01a5 ae 20           LDX        #0x20

                LAB_01a7                                        XREF[1]:     01af(j)
01a7 9f              TXA
01a8 ae 0a           LDX        #0xa

                LAB_01aa                                        XREF[1]:     01ab(j)
01aa 5a              DECX
01ab 26 fd           BNE        LAB_01aa
01ad 97              TAX
01ae 5a              DECX
01af 26 f6           BNE        LAB_01a7
01b1 cc 01 4e        JMP        LAB_014e

                **************************************************************
                *                          FUNCTION                          *
                **************************************************************
                undefined Load_Display_Character_Data_From_ROM()
 undefined        ⚠<UNASSIGNED>   <RETURN>
                Load Display Character Data from ROM Lookup Table
                Converts 4 display characters to 7-segment patterns using ROM…

                INPUT: Display characters from RAM_0010-0013 (set by PANC com…
                OUTPUT: 7-segment patterns in Display_7Segment_Digit_0 throug…

                ROM TABLE LOOKUP:
                Uses ROM_Character_To_7Segment_Lookup table at 0x0D22:
                - Each character maps to 2 bytes of 7-segment pattern data
                - Index calculation: character_value × 2 for table lookup
                - Supports full ASCII character set to 7-segment conversion

                CONVERSION PROCESS:
                For each of 4 input characters:
                1. Multiply character value by 2 (16-bit ROM table index)
                2. Read 2 bytes from ROM table: pattern_low, pattern_high
                3. Store in consecutive 7-segment digit buffers

                MEMORY MAPPING:
                - RAM_0010 (char 0) → Display_7Segment_Digit_0, Display_7Seg…
                - RAM_0011 (char 1) → Display_7Segment_Digit_2, Display_7Seg…
                - RAM_0012 (char 2) → Display_7Segment_Digit_4, Display_7Seg…
                - RAM_0013 (char 3) → Display_7Segment_Digit_6, Display_7Seg…

                DISPLAY OUTPUT:
                7-segment patterns drive panel display hardware (DISP1-5 sign…
                providing operator-visible status and information display.
                Load_Display_Character_Data_From_ROM            XREF[1]:     Wait_For_FIFO_Commands_And_Proce…
01b4 be 13           LDX        RAM_0013
01b6 58              ASLX
01b7 d6 0d 22        LDA        DAT_0d22,X
01ba b7 59           STA        Display_7Segment_Digit_7
01bc 5c              INCX
01bd d6 0d 22        LDA        DAT_0d22,X
01c0 b7 58           STA        Display_7Segment_Digit_6
01c2 be 12           LDX        RAM_0012
01c4 58              ASLX
01c5 d6 0d 22        LDA        DAT_0d22,X
01c8 b7 57           STA        Display_7Segment_Digit_5
01ca 5c              INCX
01cb d6 0d 22        LDA        DAT_0d22,X
01ce b7 56           STA        Display_7Segment_Digit_4
01d0 be 11           LDX        RAM_0011
01d2 58              ASLX
01d3 d6 0d 22        LDA        DAT_0d22,X
01d6 b7 55           STA        Display_7Segment_Digit_3
01d8 5c              INCX
01d9 d6 0d 22        LDA        DAT_0d22,X
01dc b7 54           STA        Display_7Segment_Digit_2
01de be 10           LDX        RAM_0010
01e0 58              ASLX
01e1 d6 0d 22        LDA        DAT_0d22,X
01e4 b7 53           STA        Display_7Segment_Digit_1
01e6 5c              INCX
```

```
        01e7 d6 0d 22       LDA        DAT_0d22,X
        01ea b7 52          STA        Display_7Segment_Digit_0
        01ec 81             RTS
```

```
                 **************************************************************
                 *                           FUNCTION                         *
                 **************************************************************
                 undefined Format_7Segment_Display_Patterns()
 undefined         ⚠<UNASSIGNED>   <RETURN>
                 Format 7-Segment Display Patterns Based on Operational Mode
                 Central display formatting function with multiple display mod…

                 ============================================================…
                 DISPLAY MODE ARCHITECTURE
                 ============================================================…

                 The MC68705 provides sophisticated display capabilities throu…
                 operational modes selected by Display_Control_Mode_Flags and …

                 ============================================================…
                 MODE 1: REAL-TIME STATUS DISPLAY (Flags bits 1+2 set)
                 ============================================================…

                 BLINKING TIME DISPLAY:
                 - Alternates between two patterns based on Software_RTC_Tenth…
                 - Pattern 1 (< 0x4F): BYTE_0052=0x6B, BYTE_0053=0x77
                 - Pattern 2 (≥ 0x4F): BYTE_0052=0x11, BYTE_0053=0x3F
                 - Creates visual blinking effect for time indication

                 TIME DIGIT CONVERSION:
                 - Uses ROM_BCD_To_7Segment_Table (0x0E44) for digit patterns
                 - Converts Software RTC counters to 7-segment patterns
                 - ROM_7Segment_Character_Table (0x0CF8) provides final patter…
                 - Shows current time with blinking seconds indication

                 ============================================================…
                 MODE 2: STANDARD RTC TIME DISPLAY (Flag bit 1 set, bit 2 clea…
                 ============================================================…

                 RTC REGISTER DISPLAY:
                 - Displays time from Software RTC counters (0x22-0x25)
                 - Note: Variable names misleading - these are software counte…
                 - BCD digit conversion via ROM_BCD_To_7Segment_Table lookup
                 - Decimal point indicators: OR with 0x80 on positions 3,5,7

                 TIME FORMAT:
                 - HH:MM:SS format with decimal points as separators
                 - Real-time updates from 333ms software RTC increments
                 - Stable, non-blinking time display for normal operation

                 ============================================================…
                 MODE 3: CPU UTILIZATION DATA DISPLAY (Parameter bits 1-0 set)
                 ============================================================…

                 PERFORMANCE METRICS DISPLAY:
                 - Shows CPU utilization data from RAM_0015, RAM_0016
                 - 2-bit field extraction and conversion to display patterns
                 - Uses ROM_7Segment_Character_Table with offset +4
                 - Provides operator visibility into system performance

                 UTILIZATION INDICATORS:
                 - CPU busy percentage (derived from LEV0 signal)
                 - Cache hit efficiency (derived from LHIT signal)
                 - System load indicators (Ring level distribution)
                 - Real-time performance feedback for operators

                 ============================================================…
                 MODE 4: ENHANCED DIAGNOSTIC MODE (Flag bit 3 set)
                 ============================================================…

                 MINIMAL DISPLAY MODE:
                 - Clears all segments: Display_7Segment_Digit_0 through _7 = 0
                 - Shows only essential indicators: positions 3,4 = 8
                 - Used for diagnostic operations or reduced display requireme…
                 - Indicates system in special operational state

                 ============================================================…
                 MODE 5: STANDARD COMMAND DISPLAY (Default)
                 ============================================================…

                 COMMAND-DRIVEN DISPLAY:
                 - Shows data from PANC command buffers (RAM_0014-0018)
                 - Complex 3-bit field extraction and positioning
                 - ROM_7Segment_Character_Table lookup with offset +0x10
                 - Full ASCII character support via ROM table conversion

                 SPECIAL PROCESSING:
                 - Pattern validation: Clear segments if value = 'w' (0x77)
                 - Prevents invalid display patterns from corrupting output
                 - Ensures display integrity during command processing
```

```
        01f0 04 1b 03        BRSET      0x2,Display_Control_Mode_Flags,LAB_01f6
        01f3 cc 02 e7        JMP        LAB_02e7

                         LAB_01f6                                        XREF[1]:     01f0(j)
        01f6 cc 03 23        JMP        LAB_0323

                         LAB_01f9                                        XREF[1]:     01ed(j)
        01f9 b6 1d           LDA        CPU_Utilization_Parameter_1
        01fb a4 03           AND        #0x3
        01fd 27 03           BEQ        LAB_0202
        01ff cc 02 aa        JMP        LAB_02aa

                         LAB_0202                                        XREF[1]:     01fd(j)
        0202 07 1b 0e        BRCLR      0x3,Display_Control_Mode_Flags,LAB_0213
        0205 ae 07           LDX        #0x7

                         LAB_0207                                        XREF[1]:     020a(j)
        0207 6f 52           CLR        0x52,X=>Display_7Segment_Digit_7
        0209 5a              DECX
        020a 2a fb           BPL        LAB_0207
        020c a6 08           LDA        #0x8
        020e b7 55           STA        Display_7Segment_Digit_3
        0210 b7 56           STA        Display_7Segment_Digit_4
        0212 81              RTS

                         LAB_0213                                        XREF[1]:     0202(j)
        0213 b6 16           LDA        RAM_0016
        0215 a4 07           AND        #0x7
        0217 ab 10           ADD        #0x10
        0219 97              TAX
        021a d6 0c f8        LDA        DAT_0cf8,X
        021d b7 59           STA        Display_7Segment_Digit_7
        021f b6 16           LDA        RAM_0016
        0221 44              LSRA
        0222 44              LSRA
        0223 44              LSRA
        0224 a4 07           AND        #0x7
        0226 ab 10           ADD        #0x10
        0228 97              TAX
        0229 d6 0c f8        LDA        DAT_0cf8,X
        022c b7 58           STA        Display_7Segment_Digit_6
        022e be 16           LDX        RAM_0016
        0230 b6 15           LDA        RAM_0015
        0232 58              ASLX
        0233 49              ROLA
        0234 58              ASLX
        0235 49              ROLA
        0236 a4 07           AND        #0x7
        0238 ab 10           ADD        #0x10
        023a 97              TAX
        023b d6 0c f8        LDA        DAT_0cf8,X
        023e b7 57           STA        Display_7Segment_Digit_5
        0240 b6 15           LDA        RAM_0015
        0242 44              LSRA
        0243 a4 07           AND        #0x7
        0245 ab 10           ADD        #0x10
        0247 97              TAX
        0248 d6 0c f8        LDA        DAT_0cf8,X
        024b b7 56           STA        Display_7Segment_Digit_4
        024d b6 15           LDA        RAM_0015
        024f 44              LSRA
        0250 44              LSRA
        0251 44              LSRA
        0252 44              LSRA
        0253 a4 07           AND        #0x7
        0255 ab 10           ADD        #0x10
        0257 97              TAX
        0258 d6 0c f8        LDA        DAT_0cf8,X
        025b b7 55           STA        Display_7Segment_Digit_3
        025d be 15           LDX        RAM_0015
        025f b6 14           LDA        RAM_0014
        0261 58              ASLX
        0262 49              ROLA
        0263 a4 07           AND        #0x7
        0265 ab 10           ADD        #0x10
        0267 97              TAX
        0268 d6 0c f8        LDA        DAT_0cf8,X
        026b b7 54           STA        Display_7Segment_Digit_2
        026d b6 14           LDA        RAM_0014
        026f 44              LSRA
        0270 44              LSRA
        0271 a4 07           AND        #0x7
        0273 ab 10           ADD        #0x10
        0275 97              TAX
        0276 d6 0c f8        LDA        DAT_0cf8,X
```

```
0279 b7 53          STA        Display_7Segment_Digit_1
027b b6 14          LDA        RAM_0014
027d 48             ASLA
027e 49             ROLA
027f 49             ROLA
0280 49             ROLA
0281 a4 07          AND        #0x7
0283 ab 10          ADD        #0x10
0285 97             TAX
0286 d6 0c f8       LDA        DAT_0cf8,X
0289 b7 52          STA        Display_7Segment_Digit_0
028b 5f             CLRX
028c a6 77          LDA        #0x77

                LAB_028e                                        XREF[1]:     0299(j)
028e e1 52          CMP        0x52,X
0290 26 09          BNE        LAB_029b
0292 6f 52          CLR        0x52,X
0294 5c             INCX
0295 a3 07          CPX        #0x7
0297 27 02          BEQ        LAB_029b
0299 20 f3          BRA        LAB_028e

                LAB_029b                                        XREF[2]:     0290(j), 0297(j)
029b 08 1b 01       BRSET      0x4,Display_Control_Mode_Flags,LAB_029f
029e 81             RTS

                LAB_029f                                        XREF[1]:     029b(j)
029f a6 4f          LDA        #0x4f
02a1 b7 52          STA        Display_7Segment_Digit_0
02a3 a6 66          LDA        #0x66
02a5 b7 53          STA        Display_7Segment_Digit_1
02a7 1e 55          BSET       0x7,Display_7Segment_Digit_3
02a9 81             RTS

                LAB_02aa                                        XREF[1]:     01ff(j)
02aa b6 16          LDA        RAM_0016
02ac b7 5a          STA        PortC_SavedValue
02ae ae 03          LDX        #0x3
02b0 bf 51          STX        Display_Position_Loop_Counter

                LAB_02b2                                        XREF[1]:     02c6(j)
02b2 b6 5a          LDA        PortC_SavedValue
02b4 a4 03          AND        #0x3
02b6 ab 04          ADD        #0x4
02b8 97             TAX
02b9 d6 0c f8       LDA        DAT_0cf8,X
02bc be 51          LDX        Display_Position_Loop_Counter
02be e7 56          STA        0x56,X=>Display_7Segment_Digit_7
02c0 34 5a          LSR        PortC_SavedValue
02c2 34 5a          LSR        PortC_SavedValue
02c4 3a 51          DEC        Display_Position_Loop_Counter
02c6 2a ea          BPL        LAB_02b2
02c8 b6 15          LDA        RAM_0015
02ca b7 5a          STA        PortC_SavedValue
02cc ae 03          LDX        #0x3
02ce bf 51          STX        Display_Position_Loop_Counter

                LAB_02d0                                        XREF[1]:     02e4(j)
02d0 b6 5a          LDA        PortC_SavedValue
02d2 a4 03          AND        #0x3
02d4 ab 04          ADD        #0x4
02d6 97             TAX
02d7 d6 0c f8       LDA        DAT_0cf8,X
02da be 51          LDX        Display_Position_Loop_Counter
02dc e7 52          STA        0x52,X=>Display_7Segment_Digit_3
02de 34 5a          LSR        PortC_SavedValue
02e0 34 5a          LSR        PortC_SavedValue
02e2 3a 51          DEC        Display_Position_Loop_Counter
02e4 2a ea          BPL        LAB_02d0
02e6 81             RTS

                LAB_02e7                                        XREF[1]:     01f3(j)
02e7 ae 07          LDX        #0x7
02e9 bf 4f          STX        Display_Buffer_Index_Counter
02eb ae 03          LDX        #0x3
02ed bf 51          STX        Display_Position_Loop_Counter

                LAB_02ef                                        XREF[1]:     031a(j)
02ef ee 22          LDX        0x22,X=>Software_RTC_Units_Hours
02f1 58             ASLX
02f2 5c             INCX
02f3 d6 0e 44       LDA        DAT_0e44,X
02f6 ab 10          ADD        #0x10
02f8 97             TAX
```

```
02f9 d6 0c f8       LDA         DAT_0cf8,X
02fc be 4f          LDX         Display_Buffer_Index_Counter
02fe e7 52          STA         0x52,X=>Display_7Segment_Digit_7
0300 be 51          LDX         Display_Position_Loop_Counter
0302 ee 22          LDX         0x22,X=>Software_RTC_Units_Hours
0304 58             ASLX
0305 d6 0e 44       LDA         DAT_0e44,X
0308 ab 10          ADD         #0x10
030a 97             TAX
030b d6 0c f8       LDA         DAT_0cf8,X
030e 3a 4f          DEC         Display_Buffer_Index_Counter
0310 be 4f          LDX         Display_Buffer_Index_Counter
0312 e7 52          STA         0x52,X=>Display_7Segment_Digit_6
0314 3a 4f          DEC         Display_Buffer_Index_Counter
0316 3a 51          DEC         Display_Position_Loop_Counter
0318 be 51          LDX         Display_Position_Loop_Counter
031a 2a d3          BPL         LAB_02ef
031c 1e 59          BSET        0x7,Display_7Segment_Digit_7
031e 1e 57          BSET        0x7,Display_7Segment_Digit_5
0320 1e 55          BSET        0x7,Display_7Segment_Digit_3
0322 81             RTS

                LAB_0323                                XREF[1]:     01f6(j)
0323 b6 20          LDA         Software_RTC_Tenths_Seconds
0325 a1 4f          CMP         #0x4f
0327 25 0a          BCS         LAB_0333
0329 ae 11          LDX         #0x11
032b bf 52          STX         Display_7Segment_Digit_0
032d ae 3f          LDX         #0x3f
032f bf 53          STX         Display_7Segment_Digit_1
0331 20 08          BRA         LAB_033b

                LAB_0333                                XREF[1]:     0327(j)
0333 ae 6b          LDX         #0x6b
0335 bf 52          STX         Display_7Segment_Digit_0
0337 ae 77          LDX         #0x77
0339 bf 53          STX         Display_7Segment_Digit_1

                LAB_033b                                XREF[1]:     0331(j)
033b 97             TAX
033c 58             ASLX
033d d6 0e 44       LDA         DAT_0e44,X
0340 ab 10          ADD         #0x10
0342 97             TAX
0343 d6 0c f8       LDA         DAT_0cf8,X
0346 b7 54          STA         Display_7Segment_Digit_2
0348 be 20          LDX         Software_RTC_Tenths_Seconds
034a 58             ASLX
034b 5c             INCX
034c d6 0e 44       LDA         DAT_0e44,X
034f ab 10          ADD         #0x10
0351 97             TAX
0352 d6 0c f8       LDA         DAT_0cf8,X
0355 b7 55          STA         Display_7Segment_Digit_3
0357 3f 56          CLR         Display_7Segment_Digit_4
0359 a6 80          LDA         #0x80
035b b7 57          STA         Display_7Segment_Digit_5
035d be 21          LDX         Software_RTC_Units_Seconds
035f 58             ASLX
0360 d6 0e 44       LDA         DAT_0e44,X
0363 ab 10          ADD         #0x10
0365 97             TAX
0366 d6 0c f8       LDA         DAT_0cf8,X
0369 b7 58          STA         Display_7Segment_Digit_6
036b be 21          LDX         Software_RTC_Units_Seconds
036d 58             ASLX
036e 5c             INCX
036f d6 0e 44       LDA         DAT_0e44,X
0372 ab 10          ADD         #0x10
0374 97             TAX
0375 d6 0c f8       LDA         DAT_0cf8,X
0378 b7 59          STA         Display_7Segment_Digit_7
037a 81             RTS
```

```
                             **************************************************************
                             *                          FUNCTION                          *
                             **************************************************************
                             undefined Process_Display_Data_Alternative_Mode()
             undefined         <UNASSIGNED>   <RETURN>
                             Process Display Data for Alternative Display Modes
                             Handles special display formatting when not in standard mode.

                             CONDITIONAL PROCESSING:
                             Only executes when both conditions are FALSE:
                             - Display_Control_Mode_Flags bit 1 = 0 (not status display mo…
                             - CPU_Utilization_Parameter_1 bits 1-0 = 0 (not utilization m…

                             STANDARD ALTERNATIVE MODE:
                             Uses data from RAM_0017, RAM_0018 for display generation:
                             1. Extract 3-bit fields from data and convert to 7-segment
                             2. Complex bit manipulation for field positioning
                             3. ROM table lookup via DAT_0cf8 with offset +0x10

                             ENHANCED MODE CHECK (Display_Control_Mode_Flags bit 3 set):
                             - Additional processing using RAM_0016 data
                             - More complex bit field extraction and positioning
                             - Special pattern validation: clear if segments = 'w' (0x77)

                             SIMPLIFIED MODE (bit 3 clear):
                             - Limited display: BYTE_0054 only (based on RAM_0017 bit 7)
                             - Clear positions 2-3: BYTE_0052 = BYTE_0053 = 0
                             - Validation check: clear segments 2-6 if = 'w'

                             FALLBACK PROCESSING:
                             If primary conditions not met, uses RAM_0017, RAM_0018:
                             - Break data into 2-bit fields
                             - Convert each field to display pattern via DAT_0cf8 + 4
                             - Fill both upper (0x56-0x59) and lower (0x52-0x55) buffers

                             BIT FIELD EXTRACTION:
                             Uses shifting and masking to extract 2-bit or 3-bit fields:
                             - (data & 3): Extract bits 1-0
                             - (data >> 2 & 7): Extract bits 4-2
                             - (data >> 7): Extract bit 7
                             - Complex combinations for precise field positioning

                             PURPOSE:
                             Provides alternative display formatting for special modes
                             when standard command-driven display is not active.
                             Process_Display_Data_Alternative_Mode          XREF[1]:     Wait_For_FIFO_Commands_And_Proce…
        037b 02 1b 0b        BRSET      0x1,Display_Control_Mode_Flags,LAB_0389
        037e b6 1d           LDA        CPU_Utilization_Parameter_1
        0380 a4 03           AND        #0x3
        0382 a1 00           CMP        #0x0
        0384 26 03           BNE        LAB_0389
        0386 cc 03 8c        JMP        LAB_038c

                             LAB_0389                                       XREF[2]:     037b(j), 0384(j)
        0389 cc 04 3c        JMP        LAB_043c

                             LAB_038c                                       XREF[1]:     0386(j)
        038c b6 18           LDA        RAM_0018
        038e a4 07           AND        #0x7
        0390 ab 10           ADD        #0x10
        0392 97              TAX
        0393 d6 0c f8        LDA        DAT_0cf8,X
        0396 b7 59           STA        Display_7Segment_Digit_7
        0398 b6 18           LDA        RAM_0018
        039a 44              LSRA
        039b 44              LSRA
        039c 44              LSRA
        039d a4 07           AND        #0x7
        039f ab 10           ADD        #0x10
        03a1 97              TAX
        03a2 d6 0c f8        LDA        DAT_0cf8,X
        03a5 b7 58           STA        Display_7Segment_Digit_6
        03a7 be 18           LDX        RAM_0018
        03a9 b6 17           LDA        RAM_0017
        03ab 58              ASLX
        03ac 49              ROLA
        03ad 58              ASLX
        03ae 49              ROLA
        03af a4 07           AND        #0x7
        03b1 ab 10           ADD        #0x10
        03b3 97              TAX
        03b4 d6 0c f8        LDA        DAT_0cf8,X
        03b7 b7 57           STA        Display_7Segment_Digit_5
        03b9 b6 17           LDA        RAM_0017
```

```
03bb 44             LSRA
03bc a4 07          AND        #0x7
03be ab 10          ADD        #0x10
03c0 97             TAX
03c1 d6 0c f8       LDA        DAT_0cf8,X
03c4 b7 56          STA        Display_7Segment_Digit_4
03c6 b6 17          LDA        RAM_0017
03c8 44             LSRA
03c9 44             LSRA
03ca 44             LSRA
03cb 44             LSRA
03cc a4 07          AND        #0x7
03ce ab 10          ADD        #0x10
03d0 97             TAX
03d1 d6 0c f8       LDA        DAT_0cf8,X
03d4 b7 55          STA        Display_7Segment_Digit_3
03d6 06 1b 24       BRSET      0x3,Display_Control_Mode_Flags,LAB_03fd
03d9 b6 17          LDA        RAM_0017
03db 49             ROLA
03dc 49             ROLA
03dd a4 01          AND        #0x1
03df ab 10          ADD        #0x10
03e1 97             TAX
03e2 d6 0c f8       LDA        DAT_0cf8,X
03e5 b7 54          STA        Display_7Segment_Digit_2
03e7 3f 53          CLR        Display_7Segment_Digit_1
03e9 3f 52          CLR        Display_7Segment_Digit_0
03eb ae 02          LDX        #0x2
03ed a6 77          LDA        #0x77

                LAB_03ef                                       XREF[1]:     03fa(j)
03ef e1 52          CMP        0x52,X=>Display_7Segment_Digit_2
03f1 26 09          BNE        LAB_03fc
03f3 6f 52          CLR        0x52,X=>Display_7Segment_Digit_2
03f5 5c             INCX
03f6 a3 07          CPX        #0x7
03f8 27 02          BEQ        LAB_03fc
03fa 20 f3          BRA        LAB_03ef

                LAB_03fc                                       XREF[2]:     03f1(j), 03f8(j)
03fc 81             RTS

                LAB_03fd                                       XREF[1]:     03d6(j)
03fd be 17          LDX        RAM_0017
03ff b6 16          LDA        RAM_0016
0401 58             ASLX
0402 49             ROLA
0403 a4 07          AND        #0x7
0405 ab 10          ADD        #0x10
0407 97             TAX
0408 d6 0c f8       LDA        DAT_0cf8,X
040b b7 54          STA        Display_7Segment_Digit_2
040d b6 16          LDA        RAM_0016
040f 44             LSRA
0410 44             LSRA
0411 a4 07          AND        #0x7
0413 ab 10          ADD        #0x10
0415 97             TAX
0416 d6 0c f8       LDA        DAT_0cf8,X
0419 b7 53          STA        Display_7Segment_Digit_1
041b b6 16          LDA        RAM_0016
041d 49             ROLA
041e 49             ROLA
041f 49             ROLA
0420 49             ROLA
0421 a4 07          AND        #0x7
0423 ab 10          ADD        #0x10
0425 97             TAX
0426 d6 0c f8       LDA        DAT_0cf8,X
0429 b7 52          STA        Display_7Segment_Digit_0
042b 5f             CLRX
042c a6 77          LDA        #0x77

                LAB_042e                                       XREF[1]:     0439(j)
042e e1 52          CMP        0x52,X
0430 26 09          BNE        LAB_043b
0432 6f 52          CLR        0x52,X
0434 5c             INCX
0435 a3 07          CPX        #0x7
0437 27 02          BEQ        LAB_043b
0439 20 f3          BRA        LAB_042e

                LAB_043b                                       XREF[2]:     0430(j), 0437(j)
043b 81             RTS
```

```
                 LAB_043c                                XREF[1]:     0389(j)
0443c b6 18          LDA        RAM_0018
043e b7 5a          STA        PortC_SavedValue
0440 ae 03          LDX        #0x3
0442 bf 51          STX        Display_Position_Loop_Counter

                 LAB_0444                                XREF[1]:     0458(j)
0444 b6 5a          LDA        PortC_SavedValue
0446 a4 03          AND        #0x3
0448 ab 04          ADD        #0x4
044a 97             TAX
044b d6 0c f8       LDA        DAT_0cf8,X
044e be 51          LDX        Display_Position_Loop_Counter
0450 e7 56          STA        0x56,X=>Display_7Segment_Digit_7
0452 34 5a          LSR        PortC_SavedValue
0454 34 5a          LSR        PortC_SavedValue
0456 3a 51          DEC        Display_Position_Loop_Counter
0458 2a ea          BPL        LAB_0444
045a b6 17          LDA        RAM_0017
045c b7 5a          STA        PortC_SavedValue
045e ae 03          LDX        #0x3
0460 bf 51          STX        Display_Position_Loop_Counter

                 LAB_0462                                XREF[1]:     0476(j)
0462 b6 5a          LDA        PortC_SavedValue
0464 a4 03          AND        #0x3
0466 ab 04          ADD        #0x4
0468 97             TAX
0469 d6 0c f8       LDA        DAT_0cf8,X
046c be 51          LDX        Display_Position_Loop_Counter
046e e7 52          STA        0x52,X=>Display_7Segment_Digit_3
0470 34 5a          LSR        PortC_SavedValue
0472 34 5a          LSR        PortC_SavedValue
0474 3a 51          DEC        Display_Position_Loop_Counter
0476 2a ea          BPL        LAB_0462
0478 81             RTS

                 **************************************************************
                 *                          FUNCTION                          *
                 **************************************************************
                 undefined Process_7Segment_Display_Bit_Manipulation()
 undefined         ⚠<UNASSIGNED>   <RETURN>
                 Process 7-Segment Display Bit Manipulation
                 Complex bit processing for 7-segment display pattern generati…

                 PROCESSING PARAMETERS:
                 - BYTE_004b: Bit mask value (0x40, 0x20, or 0x10 in 3 phases)
                 - BYTE_004c: Bit shift count (4, 2, or 1 in 3 phases)

                 INPUT BUFFERS:
                 - 0x52-0x55: Lower 7-segment buffer (4 bytes)
                 - 0x56-0x59: Upper 7-segment buffer (4 bytes)

                 OUTPUT BUFFER:
                 - 0x27-0x46: Working display buffer (32 bytes)

                 BIT MANIPULATION ALGORITHM:
                 For each of 4 display positions (RAM_0051 = 3 down to 0):
                   For each of 8 bits (BYTE_0050 = 7 down to 0):
                     1. Extract LSB from upper buffer (0x56+pos)
                     2. Shift buffer right by 1 bit
                     3. If LSB was 1, set bVar5 = BYTE_004b (mask value)
                     4. Extract LSB from lower buffer (0x52+pos)
                     5. Shift buffer right by 1 bit
                     6. If LSB was 1, OR bVar5 with BYTE_004c (shift count)
                     7. Store result in working buffer at BYTE_004f position
                     8. Decrement BYTE_004f (starts at 0x1F, counts down)

                 MULTI-PHASE PROCESSING:
                 Called 3 times with different parameters:
                 - Phase 1: mask=0x40, shift=4 - Extract high-order bits
                 - Phase 2: mask=0x20, shift=2 - Extract middle bits
                 - Phase 3: mask=0x10, shift=1 - Extract low-order bits

                 RESULT:
                 Converts 7-segment pattern data into final display bit patter…
                 for output to panel hardware (DISP1-5 signals).
                 Process_7Segment_Display_Bit_Manipulation      XREF[3]:     Wait_For_FIFO_Commands_And_Proce…
                                                                              Wait_For_FIFO_Commands_And_Proce…
                                                                              Wait_For_FIFO_Commands_And_Proce…
0479 a6 1f          LDA        #0x1f
047b b7 4f          STA        Display_Buffer_Index_Counter
047d ae 03          LDX        #0x3
047f bf 51          STX        Display_Position_Loop_Counter
```

```
                LAB_0481                                XREF[1]:     04a4(j)
0481 a6 07          LDA         #0x7
0483 b7 50          STA         Display_Bit_Loop_Counter

                LAB_0485                                XREF[1]:     049e(j)
0485 4f             CLRA
0486 64 56          LSR         0x56,X=>Display_7Segment_Digit_7
0488 24 02          BCC         LAB_048c
048a ba 4b          ORA         Display_Bit_Mask_Parameter

                LAB_048c                                XREF[1]:     0488(j)
048c 64 52          LSR         0x52,X=>Display_7Segment_Digit_3
048e 24 02          BCC         LAB_0492
0490 ba 4c          ORA         Display_Bit_Shift_Parameter

                LAB_0492                                XREF[1]:     048e(j)
0492 be 4f          LDX         Display_Buffer_Index_Counter
0494 3a 4f          DEC         Display_Buffer_Index_Counter
0496 ea 27          ORA         0x27,X=>PortC_ValueToWrite
0498 e7 27          STA         0x27,X=>PortC_ValueToWrite
049a be 51          LDX         Display_Position_Loop_Counter
049c 3a 50          DEC         Display_Bit_Loop_Counter
049e 2a e5          BPL         LAB_0485
04a0 3a 51          DEC         Display_Position_Loop_Counter
04a2 be 51          LDX         Display_Position_Loop_Counter
04a4 2a db          BPL         LAB_0481
04a6 81             RTS

                **************************************************************
                *                          FUNCTION                          *
                **************************************************************
                void Delay_And_Sample_Port_C(void)
 void              <VOID>         <RETURN>
                Delay and Sample Port C Status
                Simple delay routine with Port C sampling for timing coordina…

                OPERATION:
                1. Sample current Port C state: PortC_SavedValue = PORTC
                2. Execute dual delay loops:
                   - First delay: 29 cycles (0x1D down to 0)
                   - Second delay: 31 cycles (0x1F down to 0)
                   - Total delay: ~60 cycles

                PURPOSE:
                Provides timing delays for:
                - Port C signal settling after MM58274 operations
                - Coordination between display processing phases
                - Synchronization with external hardware timing

                TIMING:
                At 4MHz MC68705 clock: ~60 cycles = ~15µs delay
                Ensures proper setup/hold times for Port C operations
                involving MM58274 RTC chip communication.

                USAGE:
                Called during command processing pipeline to provide
                necessary delays between critical timing operations.
                Delay_And_Sample_Port_C                 XREF[1]:     Wait_For_FIFO_Commands_And_Proce…
04a7 b6 02          LDA         PORTC
04a9 b7 5a          STA         PortC_SavedValue
04ab b6 46          LDA         PortC_ValueToWrite
04ad 97             TAX
04ae aa 80          ORA         #0x80
04b0 bf 02          STX         PORTC
04b2 b7 02          STA         PORTC
04b4 b6 45          LDA         BYTE_0045
04b6 b7 02          STA         PORTC
04b8 1e 02          BSET        0x7,PORTC
04ba ae 1d          LDX         #0x1d

                LAB_04bc                                XREF[1]:     04c3(j)
04bc e6 27          LDA         0x27,X=>BYTE_0044
04be b7 02          STA         PORTC
04c0 1e 02          BSET        0x7,PORTC
04c2 5a             DECX
04c3 2a f7          BPL         LAB_04bc
04c5 a6 1f          LDA         #0x1f

                LAB_04c7                                XREF[1]:     04d1(j)
04c7 97             TAX
04c8 ee 27          LDX         0x27,X=>PortC_ValueToWrite
04ca ee 86          LDX         0x86,X
04cc bf 02          STX         PORTC
04ce 1e 02          BSET        0x7,PORTC
```

```
04d0 4a              DECA
04d1 2a f4           BPL        LAB_04c7
04d3 b6 5a           LDA        PortC_SavedValue
04d5 b7 02           STA        PORTC
04d7 81              RTS

                 **************************************************************
                 *                          FUNCTION                          *
                 **************************************************************
                 undefined Process_PANC_Command_From_FIFO()
 undefined          ⚠<UNASSIGNED>   <RETURN>
                 Process Commands from ND-120 LDPANC Microcode
                 Handles low-level panel commands loaded via microcode 06.2 (L…

                 ND-120 MICROCODE COMMAND FLOW:
                 1. ND-120 software prepares command data in register
                 2. Microcode 06.2 (LDPANC) executed multiple times:
                    - First LDPANC: Load command byte to FIFO
                    - Additional LDPANC: Load parameter bytes to FIFO
                 3. Each LDPANC sets /EMP_n HIGH indicating FIFO has data
                 4. MC68705 processes all FIFO data sequentially

                 MICROCODE ADVANTAGES:
                 - Direct hardware control for precise timing
                 - Can interrupt other operations if urgent
                 - Efficient for both single-byte and multi-byte operations
                 - Hardware FIFO provides buffering and flow control

                 COMMAND BYTE FORMAT (from first LDPANC):
                 Bit 7-4: Command flags and microcode options
                 Bit 3: Data Direction (0=MC68705→ND-120, 1=ND-120→MC68705)
                 Bit 2-0: Command Type (0-7)

                 DIRECTION CONTROL:
                 Bit 3 = 0: WRITE to ND-120 (response via EPANS microcode)
                   - MC68705 outputs data via Port A
                   - ND-120 reads using IDB Source 20/21 (EPANS)
                   - Immediate response available

                 Bit 3 = 1: READ from ND-120 (data via additional LDPANC)
                   - Process command with parameters from FIFO
                   - May generate response for later EPANS reading

                 COMMAND PROCESSING:
                 Commands 0-7 executed via switch statement:
                 0: Set time/date (5 parameter bytes via 5 additional LDPANC)
                 1: Set time only (4 parameter bytes via 4 additional LDPANC)
                 2: Read CPU status (2 parameter bytes) - uses PORTD sampling
                 3: Set display mode (3 parameter bytes)
                 4: Load ROM characters (1 parameter byte) - ROM table lookup
                 5: Set system config (1 parameter byte)
                 6: Set display data (4 parameter bytes) - direct character lo…
                 7: System control (2 parameter bytes) - includes soft reset

                 MICROCODE INTEGRATION:
                 Higher-level instructions (TRR PANC, TRA PANS) are built
                 from these microcode primitives, providing both assembly
                 language convenience and microcode-level control when needed.
                 Process_PANC_Command_From_FIFO                   XREF[1]:     Wait_For_FIFO_Commands_And_Proce…
04d8 cd 09 bc        JSR        Strobe_WMM_Read_From_FIFO                         undefinedStrobe_WMM_Read_From_F…
04db b7 1c           STA        Last_PANC_Command_Byte
04dd 06 1c 03        BRSET      0x3,Last_PANC_Command_Byte,LAB_04e3
04e0 cc 06 be        JMP        Output_RTC_Time_Data_To_ND120                     undefinedOutput_RTC_Time_Data_T…
                 -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                 LAB_04e3                                         XREF[1]:     04dd(j)
04e3 b6 1c           LDA        Last_PANC_Command_Byte
04e5 a4 07           AND        #0x7
04e7 48              ASLA
04e8 48              ASLA
04e9 97              TAX

                 switchD_04ea::switchD
04ea dc 04 fc        JMP        0x4fc,X

                 LAB_04ed                                         XREF[10]:    055a(j), 0563(j), 0572(j),
                                                                               0591(j), 05c0(j), 05dc(j),
                                                                               0605(j), 060d(j), 0624(j),
                                                                               0668(j)
04ed 01 1d 03        BRCLR      0x0,CPU_Utilization_Parameter_1,LAB_04f3
04f0 cd 06 6b        JSR        Calculate_CPU_Utilization_Percentage              undefinedCalculate_CPU_Utilizat…

                 LAB_04f3                                         XREF[1]:     04ed(j)
04f3 5f              CLRX
04f4 cd 06 94        JSR        Process_Performance_Statistics                    undefinedProcess_Performance_St…
```

```
04f7 5c             INCX
04f8 cd 06 94       JSR        Process_Performance_Statistics                  undefinedProcess_Performance_St…
04fb 81             RTS

                switchD_04ea::caseD_0                          XREF[1]:    04ea(j)
04fc cc 05 1b       JMP        LAB_051b
04ff 9d             ??         9Dh

                switchD_04ea::caseD_1                          XREF[1]:    04ea(j)
0500 cc 05 75       JMP        LAB_0575
0503 9d             ??         9Dh

                switchD_04ea::caseD_2                          XREF[1]:    04ea(j)
0504 cc 05 94       JMP        LAB_0594
0507 9d             ??         9Dh

                switchD_04ea::caseD_3                          XREF[1]:    04ea(j)
0508 cc 05 c3       JMP        LAB_05c3
050b 9d             ??         9Dh

                switchD_04ea::caseD_4                          XREF[1]:    04ea(j)
050c cc 05 df       JMP        LAB_05df
050f 9d             ??         9Dh

                switchD_04ea::caseD_5                          XREF[1]:    04ea(j)
0510 cc 06 08       JMP        LAB_0608
0513 9d             ??         9Dh

                switchD_04ea::caseD_6                          XREF[1]:    04ea(j)
0514 cc 06 10       JMP        LAB_0610
0517 9d             ??         9Dh

                switchD_04ea::caseD_7                          XREF[1]:    04ea(j)
0518 cc 06 27       JMP        LAB_0627

                LAB_051b                                       XREF[1]:    04fc(j)
051b cd 09 bc       JSR        Strobe_WMM_Read_From_FIFO                       undefinedStrobe_WMM_Read_From_F…
051e b7 14          STA        RAM_0014
0520 cd 09 bc       JSR        Strobe_WMM_Read_From_FIFO                       undefinedStrobe_WMM_Read_From_F…
0523 b7 16          STA        RAM_0016
0525 cd 09 bc       JSR        Strobe_WMM_Read_From_FIFO                       undefinedStrobe_WMM_Read_From_F…
0528 b7 15          STA        RAM_0015
052a cd 09 bc       JSR        Strobe_WMM_Read_From_FIFO                       undefinedStrobe_WMM_Read_From_F…
052d b7 18          STA        RAM_0018
052f cd 09 bc       JSR        Strobe_WMM_Read_From_FIFO                       undefinedStrobe_WMM_Read_From_F…
0532 b7 17          STA        RAM_0017
0534 13 1b          BCLR       0x1,Display_Control_Mode_Flags
0536 17 1b          BCLR       0x3,Display_Control_Mode_Flags
0538 19 1b          BCLR       0x4,Display_Control_Mode_Flags
053a ae 50          LDX        #0x50
053c bf 10          STX        RAM_0010
053e 05 1a 25       BRCLR      0x2,System_Configuration_Flags,LAB_0566
0541 ae 54          LDX        #0x54
0543 bf 11          STX        RAM_0011
0545 b6 1a          LDA        System_Configuration_Flags
0547 a4 18          AND        #0x18
0549 44             LSRA
054a b7 52          STA        Display_7Segment_Digit_0
054c b6 1a          LDA        System_Configuration_Flags
054e a4 03          AND        #0x3
0550 ba 52          ORA        Display_7Segment_Digit_0
0552 a1 08          CMP        #0x8
0554 25 07          BCS        LAB_055d
0556 b7 13          STA        RAM_0013
0558 3f 12          CLR        RAM_0012
055a cc 04 ed       JMP        LAB_04ed

                LAB_055d                                       XREF[1]:    0554(j)
055d ab 30          ADD        #0x30
055f b7 13          STA        RAM_0013
0561 3f 12          CLR        RAM_0012
0563 cc 04 ed       JMP        LAB_04ed

                LAB_0566                                       XREF[1]:    053e(j)
0566 ae 45          LDX        #0x45
0568 bf 11          STX        RAM_0011
056a ae 58          LDX        #0x58
056c bf 12          STX        RAM_0012
056e ae 4d          LDX        #0x4d
0570 bf 13          STX        RAM_0013
0572 cc 04 ed       JMP        LAB_04ed

                LAB_0575                                       XREF[1]:    0500(j)
0575 cd 09 bc       JSR        Strobe_WMM_Read_From_FIFO                       undefinedStrobe_WMM_Read_From_F…
0578 b7 18          STA        RAM_0018
```

```
057a cd 09 bc       JSR       Strobe_WMM_Read_From_FIFO                 undefinedStrobe_WMM_Read_From_F…
057d b7 17          STA       RAM_0017
057f cd 09 bc       JSR       Strobe_WMM_Read_From_FIFO                 undefinedStrobe_WMM_Read_From_F…
0582 b7 16          STA       RAM_0016
0584 cd 09 bc       JSR       Strobe_WMM_Read_From_FIFO                 undefinedStrobe_WMM_Read_From_F…
0587 b7 15          STA       RAM_0015
0589 3f 14          CLR       RAM_0014
058b 18 1b          BSET      0x4,Display_Control_Mode_Flags
058d 13 1b          BCLR      0x1,Display_Control_Mode_Flags
058f 17 1b          BCLR      0x3,Display_Control_Mode_Flags
0591 cc 04 ed       JMP       LAB_04ed

                 LAB_0594                                  XREF[1]:     0504(j)
0594 cd 09 bc       JSR       Strobe_WMM_Read_From_FIFO                 undefinedStrobe_WMM_Read_From_F…
0597 b7 18          STA       RAM_0018
0599 cd 09 bc       JSR       Strobe_WMM_Read_From_FIFO                 undefinedStrobe_WMM_Read_From_F…
                 PANC Command 3: Set Display Mode (0x0B)
                 Input: 3 bytes from DGA
                 Byte 1 → RAM_0016 (Display mode 1)
                 Byte 2 → RAM_0018 (Display mode 2)
                 Byte 3 → RAM_0017 (Display mode 3)

                 Processing:
                 1. Update control flags: RAM_001b = (RAM_001b & 0xED) | 0x08
                    - Clear bit 4, set bit 3 (display mode indicator)
                 2. Clear data registers: RAM_0015 = 0, RAM_0014 = 0

                 Memory Layout:
                 - RAM_0014: Cleared
                 - RAM_0015: Cleared
                 - RAM_0016: Display mode parameter 1
                 - RAM_0017: Display mode parameter 3
                 - RAM_0018: Display mode parameter 2
059c b7 17          STA       RAM_0017
059e 12 1b          BSET      0x1,Display_Control_Mode_Flags
05a0 19 1b          BCLR      0x4,Display_Control_Mode_Flags
05a2 17 1b          BCLR      0x3,Display_Control_Mode_Flags
05a4 a6 10          LDA       #0x10
05a6 b7 1d          STA       CPU_Utilization_Parameter_1
05a8 a6 00          LDA       #0x0
05aa b7 1e          STA       CPU_Utilization_Parameter_2
05ac b6 19          LDA       ND120_CPU_Status_Ring_PONI_IONI
05ae a4 0f          AND       #0xf
05b0 ab 10          ADD       #0x10
05b2 b7 12          STA       RAM_0012
05b4 b6 19          LDA       ND120_CPU_Status_Ring_PONI_IONI
05b6 a4 30          AND       #0x30
05b8 44             LSRA
05b9 44             LSRA
05ba 44             LSRA
05bb 44             LSRA
                 PANC Command 4: Load Display Characters from ROM (0x0C)
                 Input: 1 byte from DGA (character selection index)

                 ROM Table Lookup Process:
                 1. Table 1 (ROM 0x0D12): Use bits 1-0, multiply by 2
                    - RAM_0013 = ROM_TABLE_1[index]     (character 3)
                    - RAM_0012 = ROM_TABLE_1[index+1]   (character 2)

                 2. Table 2 (ROM 0x0D1A): Use bits 4-3, shift right 2
                    - RAM_0011 = ROM_TABLE_2[index]     (character 1)
                    - RAM_0010 = ROM_TABLE_2[index+1]   (character 0)

                 Character Selection:
                 - Bits 1-0: Select character pair from Table 1
                 - Bits 4-3: Select character pair from Table 2
                 - Result: 4 display characters loaded into RAM_0010-0013
05bc ab 3a          ADD       #0x3a
05be b7 13          STA       RAM_0013
05c0 cc 04 ed       JMP       LAB_04ed

                 LAB_05c3                                  XREF[1]:     0508(j)
05c3 16 1b          BSET      0x3,Display_Control_Mode_Flags
05c5 13 1b          BCLR      0x1,Display_Control_Mode_Flags
05c7 19 1b          BCLR      0x4,Display_Control_Mode_Flags
05c9 cd 09 bc       JSR       Strobe_WMM_Read_From_FIFO                 undefinedStrobe_WMM_Read_From_F…
05cc b7 16          STA       RAM_0016
05ce 3f 15          CLR       RAM_0015
05d0 3f 14          CLR       RAM_0014
05d2 cd 09 bc       JSR       Strobe_WMM_Read_From_FIFO                 undefinedStrobe_WMM_Read_From_F…
05d5 b7 18          STA       RAM_0018
05d7 cd 09 bc       JSR       Strobe_WMM_Read_From_FIFO                 undefinedStrobe_WMM_Read_From_F…
05da b7 17          STA       RAM_0017
05dc cc 04 ed       JMP       LAB_04ed
```

```
                  LAB_05df                                    XREF[1]:     050c(j)
05df cd 09 bc        JSR        Strobe_WMM_Read_From_FIFO                      undefinedStrobe_WMM_Read_From_F…
05e2 b7 52           STA        Display_7Segment_Digit_0
05e4 a4 03           AND        #0x3
05e6 48              ASLA
05e7 97              TAX
05e8 d6 0d 12        LDA        DAT_0d12,X                                     = 3Ah    :
05eb b7 13           STA        RAM_0013
05ed 5c              INCX
05ee d6 0d 12        LDA        DAT_0d12,X                                     = 3Ah    :
05f1 b7 12           STA        RAM_0012
05f3 b6 52           LDA        Display_7Segment_Digit_0
05f5 a4 18           AND        #0x18
05f7 44              LSRA
05f8 44              LSRA
05f9 97              TAX
05fa d6 0d 1a        LDA        DAT_0d1a,X                                     = 43h    C
05fd b7 11           STA        RAM_0011
05ff 5c              INCX
0600 d6 0d 1a        LDA        DAT_0d1a,X                                     = 43h    C
0603 b7 10           STA        RAM_0010
0605 cc 04 ed        JMP        LAB_04ed

                  LAB_0608                                    XREF[1]:     0510(j)
0608 cd 09 bc        JSR        Strobe_WMM_Read_From_FIFO                      undefinedStrobe_WMM_Read_From_F…
060b b7 1a           STA        System_Configuration_Flags
060d cc 04 ed        JMP        LAB_04ed

                  PANC Command 5: Set System Configuration (0x0D)
                  Input: 1 byte from DGA (configuration value)

                  Processing:
                  - Direct storage: Configuration byte → RAM_001a
                  - No additional processing or transformations

                  Configuration Usage:
                  - Bit 2: Mode select in Command 0 (0=Normal "PEXM", 1=Test "P…
                  - Bits 4-3: Test mode format selection for Command 0
                  - Bits 1-0: Test mode options for Command 0
                  - Used by other commands for operational control
                  LAB_0610                                    XREF[1]:     0514(j)
0610 cd 09 bc        JSR        Strobe_WMM_Read_From_FIFO                      undefinedStrobe_WMM_Read_From_F…
0613 b7 11           STA        RAM_0011
0615 cd 09 bc        JSR        Strobe_WMM_Read_From_FIFO                      undefinedStrobe_WMM_Read_From_F…
                  PANC Command 6: Set Direct Display Data (0x0E)
                  Input: 4 bytes from DGA (display characters)
                  Byte 1 → RAM_0011 (Display character 1)
                  Byte 2 → RAM_0010 (Display character 0)
                  Byte 3 → RAM_0013 (Display character 3)
                  Byte 4 → RAM_0012 (Display character 2)

                  Processing:
                  - Direct loading: Characters stored as-is in display buffer
                  - No transformation or lookup table processing
                  - Characters used directly for 7-segment panel display

                  Memory Layout:
                  - RAM_0010: Display character 0 (rightmost)
                  - RAM_0011: Display character 1
                  - RAM_0012: Display character 2
                  - RAM_0013: Display character 3 (leftmost)
0618 b7 10           STA        RAM_0010
061a cd 09 bc        JSR        Strobe_WMM_Read_From_FIFO                      undefinedStrobe_WMM_Read_From_F…
061d b7 13           STA        RAM_0013
061f cd 09 bc        JSR        Strobe_WMM_Read_From_FIFO                      undefinedStrobe_WMM_Read_From_F…
0622 b7 12           STA        RAM_0012
0624 cc 04 ed        JMP        LAB_04ed

                  LAB_0627                                    XREF[1]:     0518(j)
0627 cd 09 bc        JSR        Strobe_WMM_Read_From_FIFO                      undefinedStrobe_WMM_Read_From_F…
062a b7 1d           STA        CPU_Utilization_Parameter_1
062c cd 09 bc        JSR        Strobe_WMM_Read_From_FIFO                      undefinedStrobe_WMM_Read_From_F…
062f b7 1e           STA        CPU_Utilization_Parameter_2
0631 a4 f0           AND        #0xf0
0633 a1 40           CMP        #0x40
0635 26 15           BNE        LAB_064c
0637 a6 40           LDA        #0x40
0639 b7 0a           STA        MR_Misc_register
063b a6 47           LDA        #0x47
063d b7 09           STA        TCR_Timer_Control_Register
063f a6 01           LDA        #0x1
0641 b7 0b           STA        PCR_Program_Control_Register
0643 3f 04           CLR        DDRA
0645 3f 05           CLR        DDRB
```

```
0647 3f 06         CLR        DDRC
0649 cc 01 10      JMP        Wait_For_FIFO_Commands_And_Process              undefinedWait_For_FIFO_Commands…
                -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                LAB_064c                                       XREF[1]:     0635(j)
064c 15 1b         BCLR       0x2,Display_Control_Mode_Flags
064e a1 10         CMP        #0x10
0650 26 02         BNE        LAB_0654
0652 14 1b         BSET       0x2,Display_Control_Mode_Flags

                LAB_0654                                       XREF[1]:     0650(j)
0654 b6 1d         LDA        CPU_Utilization_Parameter_1
0656 48            ASLA
0657 39 1e         ROL        CPU_Utilization_Parameter_2
0659 48            ASLA
065a 39 1e         ROL        CPU_Utilization_Parameter_2
065c b6 1e         LDA        CPU_Utilization_Parameter_2
065e a4 3f         AND        #0x3f
0660 b7 1e         STA        CPU_Utilization_Parameter_2
0662 b6 1d         LDA        CPU_Utilization_Parameter_1
0664 a4 3f         AND        #0x3f
0666 b7 1d         STA        CPU_Utilization_Parameter_1
0668 cc 04 ed      JMP        LAB_04ed
```

```
                     **************************************************************
                     *                          FUNCTION                          *
                     **************************************************************
                     undefined Calculate_CPU_Utilization_Percentage()
     undefined         ⚠<UNASSIGNED>   <RETURN>
                     Calculate CPU Utilization Percentage from Ring Level Data
                     Performs sophisticated CPU utilization calculation using ring…

                     INPUT PROCESSING:
                     - Uses data from Calculate_CPU_Ring_Utilization for base calc…
                     - CPU_Utilization_Parameter_2: Shift count for precision cont…
                     - rtc_datetime.hours: Ring level timing data (misleading vari…

                     UTILIZATION ALGORITHM:
                     High-precision multi-byte arithmetic for accurate percentage …

                     1. INITIALIZATION:
                        - ring_accumulator = rtc_datetime.hours (if utilization en…
                        - Multi-byte registers: rtc_datetime.month, rtc_datetime.d…

                     2. PRECISION SHIFTING:
                        For shift_count = CPU_Utilization_Parameter_2:
                        - Perform 24-bit right shift operation
                        - Maintains precision through carry propagation
                        - ring_accumulator >> 1, month >> 1 | day << 7, day >> 1 |…

                     3. PERCENTAGE CONVERSION:
                        - Extract fractional bits: rtc_datetime.month & 0x0F
                        - Convert to percentage via bit position arithmetic
                        - Each bit position represents percentage increment

                     4. BINARY TO PERCENTAGE:
                        - Left shift with bit position counting
                        - Converts binary fraction to decimal percentage
                        - Range: 0-100% CPU utilization

                     CPU UTILIZATION INTERPRETATION:
                     Ring 0 Time = User/Idle Processing:
                     - Low utilization: CPU mostly in Ring 0 (user programs)
                     - Indicates system has available capacity

                     Ring 1-3 Time = System/Kernel Processing:
                     - High utilization: CPU busy in Ring 1-3 (system operations)
                     - Ring 2-3 time indicates kernel/supervisor load

                     CALCULATION PRECISION:
                     - 24-bit arithmetic for sub-percentage accuracy
                     - Sliding window averaging reduces measurement noise
                     - 1200Hz sampling provides high temporal resolution

                     RESULT USAGE:
                     Calculated utilization percentage used for:
                     - Operator display (visual CPU load indicator)
                     - Performance monitoring and trending
                     - System capacity planning
                     - Workload characterization

                     PERFORMANCE METRICS OUTPUT:
                     - Overall CPU busy percentage (100% - Ring 0 time)
                     - System load percentage (Ring 2-3 time)
                     - Real-time responsiveness indicator
                     Calculate_CPU_Utilization_Percentage           XREF[1]:     Process_PANC_Command_From_FIFO:0…
    066b 3f 52           CLR        Display_7Segment_Digit_0
    066d 07 1b 04        BRCLR      0x3,Display_Control_Mode_Flags,LAB_0674
    0670 b6 16           LDA        RAM_0016
    0672 b7 52           STA        Display_7Segment_Digit_0

                     LAB_0674                                       XREF[1]:     066d(j)
    0674 be 1e           LDX        CPU_Utilization_Parameter_2
    0676 5d              TSTX
    0677 27 09           BEQ        LAB_0682

                     LAB_0679                                       XREF[1]:     0680(j)
    0679 34 52           LSR        Display_7Segment_Digit_0
    067b 36 17           ROR        RAM_0017
    067d 36 18           ROR        RAM_0018
    067f 5a              DECX
    0680 26 f7           BNE        LAB_0679

                     LAB_0682                                       XREF[1]:     0677(j)
    0682 b6 18           LDA        RAM_0018
    0684 a4 0f           AND        #0xf
    0686 97              TAX
    0687 3f 17           CLR        RAM_0017
```

```
0689 3f 18          CLR        RAM_0018
068b 99             SEC

                 LAB_068c                                     XREF[1]:     0691(j)
068c 39 18          ROL        RAM_0018
068e 39 17          ROL        RAM_0017
0690 5a             DECX
0691 2a f9          BPL        LAB_068c
0693 81             RTS

                 **************************************************************
                 *                          FUNCTION                          *
                 **************************************************************
                 undefined Process_Performance_Statistics()
 undefined         <UNASSIGNED>   <RETURN>
                 Process Performance Statistics for Idle vs Busy Time Analysis
                 Handles statistical processing of CPU utilization data for di…

                 DUAL-MODE PROCESSING:
                 Processes both idle time (Ring 0) and busy time (Ring 1-3) st…
                 using sophisticated bit manipulation for accurate utilization…

                 IDLE TIME PROCESSING (Ring 0 - User/Idle):
                 if (CPU_Utilization_Parameter_1 & 0x10):
                 - idle_accumulator = ~utilization_buffer[0x0F] & utilization_…
                 - AND-NOT operation isolates idle time periods
                 - Counts time when CPU is in Ring 0 (user programs or true id…

                 BUSY TIME PROCESSING (Ring 1-3 - System/Kernel):
                 busy_accumulator = utilization_buffer[0x07] (base value)
                 if (CPU_Utilization_Parameter_1 & 0x08):
                 - busy_accumulator = (~utilization_buffer[0x0F] | utilization…
                 - OR-NOT operation with masking for busy time isolation
                 - Counts time when CPU is in Ring 1-3 (system operations)

                 COMBINED UTILIZATION CALCULATION:
                 - Final result: busy_accumulator |= idle_accumulator
                 - Combines both measurements for complete utilization picture
                 - Updates both utilization_buffer[0x0E] and [0x07] for next c…

                 STATISTICAL BUFFERING:
                 Uses 15-byte history buffer (0x61-0x6F) for data smoothing:
                 - Buffer provides 200ms sliding window of measurements
                 - Reduces noise and provides stable utilization readings
                 - Enables trend analysis and performance characterization

                 CPU UTILIZATION INTERPRETATION:
                 Ring 0 (Idle) Time:
                 - High percentage: System has available capacity
                 - Low percentage: CPU heavily loaded with user programs

                 Ring 1-3 (Busy) Time:
                 - Ring 1: System services and library functions
                 - Ring 2: Operating system kernel operations
                 - Ring 3: Hardware control and supervisor functions
                 - Higher ring activity indicates more critical system load

                 PERFORMANCE METRICS:
                 Final utilization calculation enables:
                 - Real-time CPU load display for operators
                 - Performance trend analysis over time
                 - System capacity planning and optimization
                 - Workload characterization (user vs system vs kernel)

                 OUTPUT USAGE:
                 Processed statistics feed into display formatting functions
                 for operator panel indicators showing current CPU utilization,
                 cache performance, and system load in real-time.
                 Process_Performance_Statistics                XREF[2]:     Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
0694 4f             CLRA
0695 09 1d 05       BRCLR      0x4,CPU_Utilization_Parameter_1,LAB_069d
0698 e6 6d          LDA        0x6d,X
069a 43             COMA
069b e4 5e          AND        0x5e,X

                 LAB_069d                                     XREF[1]:     0695(j)
069d b7 52          STA        Display_7Segment_Digit_0
069f e6 17          LDA        0x17,X
06a1 07 1d 07       BRCLR      0x3,CPU_Utilization_Parameter_1,LAB_06ab
06a4 e6 6d          LDA        0x6d,X
06a6 43             COMA
06a7 ea 5e          ORA        0x5e,X
06a9 e4 17          AND        0x17,X
```

```
                 LAB_06ab                                      XREF[1]:     06a1(j)
06ab ba 52           ORA        Display_7Segment_Digit_0
06ad e7 5e           STA        0x5e,X
06af e7 17           STA        0x17,X
06b1 81              RTS

                 **************************************************************
                 *                          FUNCTION                          *
                 **************************************************************
                 undefined Update_CPU_Ring_PONI_IONI_Statistics()
 undefined         ⚠<UNASSIGNED>   <RETURN>
                 Update CPU Ring/PONI/IONI Statistics Accumulator
                 Processes CPU privilege level and protection status for perfo…

                 STATISTICAL ACCUMULATION:
                 Takes current CPU status sample and updates historical accumu…
                 buffer for ring level distribution and system mode analysis.

                 PROCESSING LOGIC:
                 1. status_mask = ~utilization_buffer[0x0F]: Get accumulation …
                 2. Combine current status with existing buffer data
                 3. Store updated status: utilization_buffer[0x00] = combined_…
                 4. Update current status for next cycle

                 RING LEVEL TRACKING:
                 Accumulates time spent in each CPU privilege ring (0-3):
                 - Ring 0: User application execution time
                 - Ring 1: System service execution time
                 - Ring 2: Kernel operation execution time
                 - Ring 3: Supervisor/hardware control time

                 SYSTEM MODE TRACKING:
                 Accumulates SINTRAN OS protection state:
                 - PONI: Memory protection active periods
                 - IONI: Interrupt system active periods
                 - Combined with ring data shows OS activity patterns

                 UTILIZATION BUFFER:
                 15-byte sliding window (0x61-0x6F) maintains recent history
                 for statistical smoothing and trend analysis.

                 PERFORMANCE METRICS:
                 Data used to calculate:
                 - CPU utilization by privilege level
                 - System vs user execution ratio
                 - Memory protection usage patterns
                 - Interrupt system activity levels
                 Update_CPU_Ring_PONI_IONI_Statistics          XREF[1]:     Timer_1200Hz_CPU_Performance_Mon…
06b2 b6 6f           LDA        CPU_Utilization_History_Buffer_End
06b4 43              COMA
06b5 b4 60           AND        CPU_Status_Accumulator
06b7 ba 19           ORA        ND120_CPU_Status_Ring_PONI_IONI
06b9 b7 60           STA        CPU_Status_Accumulator
06bb b7 19           STA        ND120_CPU_Status_Ring_PONI_IONI
06bd 81              RTS
```

```
                 **************************************************************
                 *                          FUNCTION                          *
                 **************************************************************
                 undefined Output_RTC_Time_Data_To_ND120()
 undefined         ⚠<UNASSIGNED>   <RETURN>
                 Output RTC/Time Data to PANS Register (Write Operations)
                 Handles PANC commands with bit 3=0 (MC68705 writes data to ND…

                 WRITE OPERATION FLOW:
                 1. ND-120 executes "TRR PANC" with DAT bit=0, cmnd=address
                 2. MC68705 reads MM58274 or internal time data
                 3. Response written to PANS RPAN field via Output_Response_To…
                 4. ND-120 executes "TRA PANS" to read time data from RPAN fie…

                 COMMAND PROCESSING:
                 Input: Command byte from PANC register via FIFO
                 - Bits 2-0: Address/register select (0-7)
                 - Bit 2: Must be 1 for RTC operations to proceed
                 - Bit 5: RTC operation mode (0=read, 1=write/init)

                 ADDRESS MAPPING:
                 Commands 0-7 map to different time/status data:
                 - 0: Seconds data with special time calculation
                 - 1-3: Minutes, hours, other time components
                 - 4-7: Extended time data and control registers

                 MM58274 INTERFACE:
                 1. Set MM58274 address via Port C (PC[2:0])
                 2. Read RTC register or use internal software counters
                 3. Format data for ND-120 consumption
                 4. Output via PA bus to PANS RPAN field

                 SPECIAL PROCESSING:
                 - Address 0: Triggers complex time calculation (CopyData_In_R…
                 - Address 3: May reinitialize MM58274 RTC chip
                 - Bit 5=1: RTC write/initialization mode vs standard read

                 RESPONSE FORMAT:
                 PANS register updated with:
                 - RPAN[7:0]: Time/date data or status value
                 - DAT bit: Set to indicate data available
                 - RDY bit: Set to indicate operation complete
                 - cmnd field: Echo of processed command

                 ND-120 DATA RETRIEVAL:
                 "TRA PANS" instruction transfers complete PANS register to CP…
                 providing time data in register A for software processing.
                 Output_RTC_Time_Data_To_ND120                 XREF[1]:     Process_PANC_Command_From_FIFO:0…
06be be 1c           LDX        Last_PANC_Command_Byte
06c0 9f              TXA
06c1 a4 07           AND        #0x7
06c3 b7 1c           STA        Last_PANC_Command_Byte
06c5 b6 02           LDA        PORTC
06c7 a4 f8           AND        #0xf8
06c9 ba 1c           ORA        Last_PANC_Command_Byte
06cb b7 02           STA        PORTC
06cd 9f              TXA
06ce a4 04           AND        #0x4
06d0 26 01           BNE        LAB_06d3
06d2 81              RTS

                 LAB_06d3                                      XREF[1]:     06d0(j)
06d3 1b 01           BCLR       0x5,PORTB
06d5 9f              TXA
06d6 a4 03           AND        #0x3
06d8 a8 03           EOR        #0x3
06da b7 1c           STA        Last_PANC_Command_Byte
06dc 9f              TXA
06dd a4 20           AND        #0x20
06df 26 19           BNE        LAB_06fa
06e1 1d 01           BCLR       0x6,PORTB
06e3 cd 09 bc        JSR        Strobe_WMM_Read_From_FIFO                    undefinedStrobe_WMM_Read_From_F…
06e6 be 1c           LDX        Last_PANC_Command_Byte
06e8 e7 47           STA        0x47,X
06ea cd 09 c5        JSR        Output_Response_To_ND120_IDB                 undefinedOutput_Response_To_ND1…
06ed 1a 01           BSET       0x5,PORTB
06ef a3 00           CPX        #0x0
06f1 26 06           BNE        LAB_06f9
06f3 cd 07 15        JSR        Convert_Time_Offset_To_Software_RTC          undefinedConvert_Time_Offset_To…
06f6 cd 09 ee        JSR        Output_Status_Code_1_To_ND120                undefinedOutput_Status_Code_1_T…

                 LAB_06f9                                      XREF[1]:     06f1(j)
06f9 81              RTS
```

```
                     LAB_06fa                                 XREF[1]:     06df(j)
06fa 1c 01              BSET       0x6,PORTB
06fc cd 09 bc           JSR        Strobe_WMM_Read_From_FIFO                    undefinedStrobe_WMM_Read_From_F…
06ff be 1c              LDX        Last_PANC_Command_Byte
0701 a3 03              CPX        #0x3
0703 26 06              BNE        LAB_070b
0705 cd 0a c0           JSR        Initialize_MM58274_RTC_Chip                  undefinedInitialize_MM58274_RTC…
0708 cd 08 03           JSR        Calculate_RTC_Date_Time_Offset               undefinedCalculate_RTC_Date_Tim…

                     LAB_070b                                 XREF[1]:     0703(j)
070b be 1c              LDX        Last_PANC_Command_Byte
070d e6 47              LDA        0x47,X
070f cd 09 c5           JSR        Output_Response_To_ND120_IDB                 undefinedOutput_Response_To_ND1…
0712 1a 01              BSET       0x5,PORTB
0714 81                 RTS

                     **************************************************************
                     *                          FUNCTION                          *
                     **************************************************************
                     undefined Convert_Time_Offset_To_Software_RTC()
     undefined         ⚠<UNASSIGNED>   <RETURN>
                     Convert Time Offset to Software RTC Format
                     Converts calculated time offset back to software RTC counter …

                     INPUT DATA:
                     - BYTE_0047, BYTE_0048: 16-bit date offset (from Calculate_RT…
                     - BYTE_0049, BYTE_004a: 16-bit time offset
                     - BYTE_0080, BYTE_0081, BYTE_0082, BYTE_0083: Conversion cons…

                     INITIALIZATION:
                     1. Clear Software RTC seconds buffer (0x20-0x25)
                     2. Load date offset to BYTE_004d, BYTE_004e
                     3. Initialize leap year cycle counter

                     YEAR CONVERSION:
                     1. Subtract leap year cycle values until underflow
                     2. Handle 4-year leap cycles (MM58274_Tens_Hours tracks cycle)
                     3. Different subtraction values: 0xDA normal, 0xDC leap year
                     4. Increment Software_RTC_Tenths_Seconds (years) for each cyc…
                     5. Add base year offset: +0x4F

                     MONTH CONVERSION:
                     1. Use ROM table DAT_0dde for days-in-month values
                     2. Subtract month lengths until underflow
                     3. Increment Software_RTC_Units_Seconds (months)
                     4. Special handling for February in leap years (+2 days)

                     DAY CONVERSION:
                     1. Check remainder bit 0 for half-day indication
                     2. Set Software_RTC_Units_Minutes to 0xC if half-day
                     3. Software_RTC_Tens_Seconds = (remainder >> 1) + 1

                     HOUR/MINUTE CONVERSION:
                     1. Load time offset to BYTE_004d, BYTE_004e
                     2. Subtract hour constants (BYTE_0082, BYTE_0083) repeatedly
                     3. Increment Software_RTC_Units_Minutes (hours) for each subt…

                     MINUTE CONVERSION:
                     1. Subtract 60-minute periods (0x3C) from remainder
                     2. Increment Software_RTC_Tens_Minutes (minutes)
                     3. Final remainder = Software_RTC_Units_Hours (seconds)

                     PURPOSE:
                     Reverse conversion from absolute time offset back to
                     software RTC format for display or MM58274 synchronization.
                     Handles leap years, variable month lengths, and time zones.

                     RESULT:
                     Software RTC counters (0x20-0x25) updated with converted time…
                     Convert_Time_Offset_To_Software_RTC      XREF[1]:     Output_RTC_Time_Data_To_ND120:06…
0715 ae 05              LDX        #0x5

                     LAB_0717                                 XREF[1]:     071a(j)
0717 6f 20              CLR        0x20,X=>Software_RTC_Units_Hours
0719 5a                 DECX
071a 2a fb              BPL        LAB_0717
071c b6 80              LDA        RTC_Leap_Year_Constant_High                  = 2h
071e b7 4b              STA        Display_Bit_Mask_Parameter
0720 b6 81              LDA        RTC_Leap_Year_Constant_Low                   = DAh
0722 b7 4c              STA        Display_Bit_Shift_Parameter
0724 a6 03              LDA        #0x3
0726 b7 26              STA        MM58274_Tens_Hours
0728 b6 47              LDA        Date_Offset_High_Byte
072a b7 4d              STA        Calculation_Working_High
```

```
072c b6 48         LDA        Date_Offset_Low_Byte
072e b7 4e         STA        Calculation_Working_Low

               LAB_0730                                XREF[1]:     0757(j)
0730 b6 4d         LDA        Calculation_Working_High
0732 b7 52         STA        Display_7Segment_Digit_0
0734 b6 4e         LDA        Calculation_Working_Low
0736 b7 53         STA        Display_7Segment_Digit_1
0738 b6 4e         LDA        Calculation_Working_Low
073a b0 4c         SUB        Display_Bit_Shift_Parameter
073c b7 4e         STA        Calculation_Working_Low
073e b6 4d         LDA        Calculation_Working_High
0740 b2 4b         SBC        Display_Bit_Mask_Parameter
0742 b7 4d         STA        Calculation_Working_High
0744 a6 da         LDA        #0xda
0746 b7 4c         STA        Display_Bit_Shift_Parameter
0748 25 0f         BCS        LAB_0759
074a 3c 20         INC        Software_RTC_Tenths_Seconds
074c 3c 26         INC        MM58274_Tens_Hours
074e 05 26 06      BRCLR      0x2,MM58274_Tens_Hours,LAB_0757
0751 3c 4c         INC        Display_Bit_Shift_Parameter
0753 3c 4c         INC        Display_Bit_Shift_Parameter
0755 3f 26         CLR        MM58274_Tens_Hours

               LAB_0757                                XREF[1]:     074e(j)
0757 20 d7         BRA        LAB_0730

               LAB_0759                                XREF[1]:     0748(j)
0759 b6 52         LDA        Display_7Segment_Digit_0
075b b7 4d         STA        Calculation_Working_High
075d b6 53         LDA        Display_7Segment_Digit_1
075f b7 4e         STA        Calculation_Working_Low
0761 b6 20         LDA        Software_RTC_Tenths_Seconds
0763 ab 4f         ADD        #0x4f
0765 b7 20         STA        Software_RTC_Tenths_Seconds
0767 3c 21         INC        Software_RTC_Units_Seconds
0769 3f 4b         CLR        Display_Bit_Mask_Parameter
076b a6 3e         LDA        #0x3e
076d b7 4c         STA        Display_Bit_Shift_Parameter

               LAB_076f                                XREF[1]:     0798(j)
076f b6 4d         LDA        Calculation_Working_High
0771 b7 52         STA        Display_7Segment_Digit_0
0773 b6 4e         LDA        Calculation_Working_Low
0775 b7 53         STA        Display_7Segment_Digit_1
0777 b6 4e         LDA        Calculation_Working_Low
0779 b0 4c         SUB        Display_Bit_Shift_Parameter
077b b7 4e         STA        Calculation_Working_Low
077d b6 4d         LDA        Calculation_Working_High
077f b2 4b         SBC        Display_Bit_Mask_Parameter
0781 b7 4d         STA        Calculation_Working_High
0783 25 15         BCS        LAB_079a
0785 3c 21         INC        Software_RTC_Units_Seconds
0787 be 21         LDX        Software_RTC_Units_Seconds
0789 d6 0d de      LDA        DAT_0dde,X
078c a3 02         CPX        #0x2
078e 26 06         BNE        LAB_0796
0790 3d 26         TST        MM58274_Tens_Hours
0792 26 02         BNE        LAB_0796
0794 ab 02         ADD        #0x2

               LAB_0796                                XREF[2]:     078e(j), 0792(j)
0796 b7 4c         STA        Display_Bit_Shift_Parameter
0798 20 d5         BRA        LAB_076f

               LAB_079a                                XREF[1]:     0783(j)
079a b6 52         LDA        Display_7Segment_Digit_0
079c b7 4d         STA        Calculation_Working_High
079e b6 53         LDA        Display_7Segment_Digit_1
07a0 b7 4e         STA        Calculation_Working_Low
07a2 44            LSRA
07a3 24 04         BCC        LAB_07a9
07a5 ae 0c         LDX        #0xc
07a7 bf 23         STX        Software_RTC_Units_Minutes

               LAB_07a9                                XREF[1]:     07a3(j)
07a9 4c            INCA
07aa b7 22         STA        Software_RTC_Tens_Seconds
07ac b6 49         LDA        Time_Offset_High_Byte
07ae b7 4d         STA        Calculation_Working_High
07b0 b6 4a         LDA        Time_Offset_Low_Byte
07b2 b7 4e         STA        Calculation_Working_Low
07b4 b6 82         LDA        RTC_Hour_Constant_High                       = Eh
07b6 b7 4b         STA        Display_Bit_Mask_Parameter
07b8 b6 83         LDA        RTC_Hour_Constant_Low                        = 10h
```

```
07ba b7 4c          STA        Display_Bit_Shift_Parameter

                LAB_07bc                                        XREF[1]:     07d4(j)
07bc b6 4d          LDA        Calculation_Working_High
07be b7 52          STA        Display_7Segment_Digit_0
07c0 b6 4e          LDA        Calculation_Working_Low
07c2 b7 53          STA        Display_7Segment_Digit_1
07c4 b6 4e          LDA        Calculation_Working_Low
07c6 b0 4c          SUB        Display_Bit_Shift_Parameter
07c8 b7 4e          STA        Calculation_Working_Low
07ca b6 4d          LDA        Calculation_Working_High
07cc b2 4b          SBC        Display_Bit_Mask_Parameter
07ce b7 4d          STA        Calculation_Working_High
07d0 25 04          BCS        LAB_07d6
07d2 3c 23          INC        Software_RTC_Units_Minutes
07d4 20 e6          BRA        LAB_07bc

                LAB_07d6                                        XREF[1]:     07d0(j)
07d6 b6 52          LDA        Display_7Segment_Digit_0
07d8 b7 4d          STA        Calculation_Working_High
07da b6 53          LDA        Display_7Segment_Digit_1
07dc b7 4e          STA        Calculation_Working_Low
07de 3f 4b          CLR        Display_Bit_Mask_Parameter
07e0 a6 3c          LDA        #0x3c
07e2 b7 4c          STA        Display_Bit_Shift_Parameter

                LAB_07e4                                        XREF[1]:     07fc(j)
07e4 b6 4d          LDA        Calculation_Working_High
07e6 b7 52          STA        Display_7Segment_Digit_0
07e8 b6 4e          LDA        Calculation_Working_Low
07ea b7 53          STA        Display_7Segment_Digit_1
07ec b6 4e          LDA        Calculation_Working_Low
07ee b0 4c          SUB        Display_Bit_Shift_Parameter
07f0 b7 4e          STA        Calculation_Working_Low
07f2 b6 4d          LDA        Calculation_Working_High
07f4 b2 4b          SBC        Display_Bit_Mask_Parameter
07f6 b7 4d          STA        Calculation_Working_High
07f8 25 04          BCS        LAB_07fe
07fa 3c 24          INC        Software_RTC_Tens_Minutes
07fc 20 e6          BRA        LAB_07e4

                LAB_07fe                                        XREF[1]:     07f8(j)
07fe b6 53          LDA        Display_7Segment_Digit_1
0800 b7 25          STA        Software_RTC_Units_Hours
0802 81             RTS
```

```
                     **************************************************************
                     *                          FUNCTION                          *
                     **************************************************************
                     undefined Calculate_RTC_Date_Time_Offset()
     undefined         ⚠<UNASSIGNED>   <RETURN>
                     Calculate RTC Date/Time Offset for Complex Time Operations
                     Performs sophisticated date/time calculations using software …

                     INPUT DATA (Software RTC counters):
                     - Software_RTC_Tenths_Seconds (0x20): Actually years
                     - Software_RTC_Units_Seconds (0x21): Actually months (1-12)
                     - Software_RTC_Tens_Seconds (0x22): Actually days (1-31)
                     - Software_RTC_Units_Minutes (0x23): Actually hours (0-23)
                     - Software_RTC_Tens_Minutes (0x24): Actually minutes (0-59)
                     - Software_RTC_Units_Hours (0x25): Actually seconds (0-59)

                     YEAR CALCULATION:
                     1. Initialize: BYTE_004d,004e,004f = 0, MM58274_Tens_Hours = 3
                     2. Loop until years (Software_RTC_Tenths_Seconds + 0xB1) < co…
                        - Add leap year cycle values (BYTE_0080, BYTE_0081)
                        - Handle 4-year leap cycle with different day counts
                        - Accumulate total days in 16-bit result

                     MONTH CALCULATION:
                     1. Adjust for leap year if MM58274_Tens_Hours = 0
                     2. Use ROM table DAT_0deb for days-in-month lookup
                     3. Add month offset to accumulated days
                     4. Handle year rollover if month > 12

                     DAY CALCULATION:
                     1. Add (days - 1) * 2 to handle day offset
                     2. Carry propagation through 16-bit arithmetic

                     HOUR CALCULATION:
                     1. Use ROM table DAT_0e1f for hour offsets
                     2. Binary multiplication by shifting for minutes
                     3. Add hours to accumulated time offset

                     FINAL RESULT:
                     - BYTE_0047, BYTE_0048: 16-bit date offset
                     - BYTE_0049, BYTE_004a: 16-bit time offset

                     PURPOSE:
                     Converts software RTC format to absolute time offset
                     for MM58274 hardware RTC synchronization or time
                     display calculations requiring exact date/time values.

                     ROM TABLES USED:
                     - DAT_0deb: Days in month lookup table
                     - DAT_0e1f: Hour/time offset calculation table
                     Calculate_RTC_Date_Time_Offset                 XREF[1]:     Output_RTC_Time_Data_To_ND120:07…
  0803 b6 20           LDA        Software_RTC_Tenths_Seconds
  0805 a0 4f           SUB        #0x4f
  0807 b7 20           STA        Software_RTC_Tenths_Seconds
  0809 3f 4d           CLR        Calculation_Working_High
  080b 3f 4e           CLR        Calculation_Working_Low
  080d 3f 4f           CLR        Display_Buffer_Index_Counter
  080f a6 03           LDA        #0x3
  0811 b7 26           STA        MM58274_Tens_Hours
  0813 b6 80           LDA        RTC_Leap_Year_Constant_High                   = 2h
  0815 b7 4b           STA        Display_Bit_Mask_Parameter
  0817 b6 81           LDA        RTC_Leap_Year_Constant_Low                    = DAh
  0819 b7 4c           STA        Display_Bit_Shift_Parameter

                     LAB_081b                                       XREF[1]:     083e(j)
  081b 3c 4f           INC        Display_Buffer_Index_Counter
  081d b6 4f           LDA        Display_Buffer_Index_Counter
  081f b1 20           CMP        Software_RTC_Tenths_Seconds
  0821 22 1d           BHI        LAB_0840
  0823 b6 4e           LDA        Calculation_Working_Low
  0825 bb 4c           ADD        Display_Bit_Shift_Parameter
  0827 b7 4e           STA        Calculation_Working_Low
  0829 b6 4d           LDA        Calculation_Working_High
  082b b9 4b           ADC        Display_Bit_Mask_Parameter
  082d b7 4d           STA        Calculation_Working_High
  082f a6 da           LDA        #0xda
  0831 b7 4c           STA        Display_Bit_Shift_Parameter
  0833 3c 26           INC        MM58274_Tens_Hours
  0835 05 26 06        BRCLR      0x2,MM58274_Tens_Hours,LAB_083e
  0838 3c 4c           INC        Display_Bit_Shift_Parameter
  083a 3c 4c           INC        Display_Bit_Shift_Parameter
  083c 3f 26           CLR        MM58274_Tens_Hours
```

```
                LAB_083e                                   XREF[1]:     0835(j)
083e 20 db          BRA         LAB_081b

                LAB_0840                                   XREF[1]:     0821(j)
0840 b6 21          LDA         Software_RTC_Units_Seconds
0842 3d 26          TST         MM58274_Tens_Hours
0844 26 02          BNE         LAB_0848
0846 ab 0d          ADD         #0xd

                LAB_0848                                   XREF[1]:     0844(j)
0848 48             ASLA
0849 97             TAX
084a 5c             INCX
084b d6 0d eb       LDA         DAT_0deb,X
084e bb 4e          ADD         Calculation_Working_Low
0850 b7 4e          STA         Calculation_Working_Low
0852 5a             DECX
0853 d6 0d eb       LDA         DAT_0deb,X
0856 b9 4d          ADC         Calculation_Working_High
0858 b7 4d          STA         Calculation_Working_High
085a b6 22          LDA         Software_RTC_Tens_Seconds
085c 4a             DECA
085d 48             ASLA
085e bb 4e          ADD         Calculation_Working_Low
0860 b7 4e          STA         Calculation_Working_Low
0862 24 02          BCC         LAB_0866
0864 3c 4d          INC         Calculation_Working_High

                LAB_0866                                   XREF[1]:     0862(j)
0866 b6 4d          LDA         Calculation_Working_High
0868 b7 47          STA         Date_Offset_High_Byte
086a b6 4e          LDA         Calculation_Working_Low
086c b7 48          STA         Date_Offset_Low_Byte
086e b6 23          LDA         Software_RTC_Units_Minutes
0870 a1 0c          CMP         #0xc
0872 25 0a          BCS         LAB_087e
0874 a0 0c          SUB         #0xc
0876 3c 48          INC         Date_Offset_Low_Byte
0878 3d 48          TST         Date_Offset_Low_Byte
087a 26 02          BNE         LAB_087e
087c 3c 47          INC         Date_Offset_High_Byte

                LAB_087e                                   XREF[2]:     0872(j), 087a(j)
087e 48             ASLA
087f 97             TAX
0880 5c             INCX
0881 d6 0e 1f       LDA         DAT_0e1f,X
0884 b7 4e          STA         Calculation_Working_Low
0886 5a             DECX
0887 d6 0e 1f       LDA         DAT_0e1f,X
088a b7 4d          STA         Calculation_Working_High
088c ae 05          LDX         #0x5
088e 3f 4b          CLR         Display_Bit_Mask_Parameter
0890 a6 3c          LDA         #0x3c
0892 b7 4c          STA         Display_Bit_Shift_Parameter
0894 b6 24          LDA         Software_RTC_Tens_Minutes
0896 b7 52          STA         Display_7Segment_Digit_0

                LAB_0898                                   XREF[1]:     08ad(j)
0898 34 52          LSR         Display_7Segment_Digit_0
089a 24 0c          BCC         LAB_08a8
089c b6 4e          LDA         Calculation_Working_Low
089e bb 4c          ADD         Display_Bit_Shift_Parameter
08a0 b7 4e          STA         Calculation_Working_Low
08a2 b6 4d          LDA         Calculation_Working_High
08a4 b9 4b          ADC         Display_Bit_Mask_Parameter
08a6 b7 4d          STA         Calculation_Working_High

                LAB_08a8                                   XREF[1]:     089a(j)
08a8 38 4c          ASL         Display_Bit_Shift_Parameter
08aa 39 4b          ROL         Display_Bit_Mask_Parameter
08ac 5a             DECX
08ad 2a e9          BPL         LAB_0898
08af b6 25          LDA         Software_RTC_Units_Hours
08b1 bb 4e          ADD         Calculation_Working_Low
08b3 b7 4e          STA         Calculation_Working_Low
08b5 24 02          BCC         LAB_08b9
08b7 3c 4d          INC         Calculation_Working_High

                LAB_08b9                                   XREF[1]:     08b5(j)
08b9 b6 4d          LDA         Calculation_Working_High
08bb b7 49          STA         Time_Offset_High_Byte
08bd b6 4e          LDA         Calculation_Working_Low
08bf b7 4a          STA         Time_Offset_Low_Byte
```

```
08c1 b6 20         LDA        Software_RTC_Tenths_Seconds
08c3 ab 4f         ADD        #0x4f
08c5 b7 20         STA        Software_RTC_Tenths_Seconds
08c7 81            RTS

                **************************************************************
                *                          FUNCTION                          *
                **************************************************************
                undefined Timer_1200Hz_CPU_Performance_Monitor()
 undefined      ⚠<UNASSIGNED>  <RETURN>
                1200Hz Timer Interrupt - ND-120 CPU Performance Monitor
                Samples ND-120 CPU status every 833μs for real-time performan…

                TIMER CONFIGURATION:
                - Clock Source: 38.4kHz PANOSC (ND-120 main clock ÷ 1024)
                - Prescaler: ÷32 (TCR = 0x35)
                - Interrupt Rate: 38.4kHz ÷ 32 = 1200Hz
                - Sample Period: 833.33μs

                HIERARCHICAL TIMING:
                Level 1: 1200Hz base sampling (CPU status, cache, utilization)
                Level 2: 75Hz processing (13.3ms) - History buffer management
                Level 3: 6Hz processing (167ms) - Statistics calculation
                Level 4: 3Hz processing (333ms) - Software RTC increment

                PERFORMANCE MONITORING:
                - Sample_ND120_CPU_Status_Signals(): Read PORTD CPU state
                - Update_CPU_Ring_PONI_IONI_Statistics(): Process privilege/p…
                - Monitor_LEV0_LHIT_Signals_Update_Display(): Track idle time…
                - Shift_CPU_Utilization_History_Buffer(): Maintain 15-sample …
                - Increment_Software_RTC_Counters(): Update internal time cou…

                CONTINUOUS OPERATION:
                This interrupt runs independently of main loop state, providi…
                real-time system monitoring whether FIFO commands are being
                processed or the system is idle waiting for /EMP_n.
                Timer_1200Hz_CPU_Performance_Monitor           XREF[1]:     0ff8(*)
08c8 a6 03         LDA        #0x3
08ca b7 08         STA        TDR_Timer_Data_Register
08cc 1f 09         BCLR       0x7,TCR_Timer_Control_Register
08ce cd 09 d4      JSR        Sample_ND120_CPU_Status_Signals               undefinedSample_ND120_CPU_Statu…
08d1 cd 06 b2      JSR        Update_CPU_Ring_PONI_IONI_Statistics          undefinedUpdate_CPU_Ring_PONI_I…
08d4 03 1b 03      BRCLR      0x1,Display_Control_Mode_Flags,LAB_08da
08d7 cd 09 85      JSR        Monitor_LEV0_LHIT_Signals_Update_Display      undefinedMonitor_LEV0_LHIT_Sign…

                LAB_08da                                       XREF[1]:     08d4(j)
08da 3a 5b         DEC        Timer_Counter_16_Cycles_13ms
08dc 26 07         BNE        LAB_08e5
08de cd 09 48      JSR        Shift_CPU_Utilization_History_Buffer          undefinedShift_CPU_Utilization_…
08e1 a6 10         LDA        #0x10
08e3 b7 5b         STA        Timer_Counter_16_Cycles_13ms

                LAB_08e5                                       XREF[1]:     08dc(j)
08e5 3a 5c         DEC        Timer_Counter_200_Cycles_167ms
08e7 26 0f         BNE        LAB_08f8
08e9 a6 c8         LDA        #0xc8
08eb b7 5c         STA        Timer_Counter_200_Cycles_167ms
08ed 3a 5d         DEC        Timer_Counter_2_Cycles_333ms
08ef 26 07         BNE        LAB_08f8
08f1 a6 02         LDA        #0x2
08f3 b7 5d         STA        Timer_Counter_2_Cycles_333ms
08f5 cd 08 f9      JSR        Increment_Software_RTC_Counters               undefinedIncrement_Software_RTC…

                LAB_08f8                                       XREF[2]:     08e7(j), 08ef(j)
08f8 80            RTI
```

```
                    ***************************************************************
                    *                          FUNCTION                           *
                    ***************************************************************
                    undefined Increment_Software_RTC_Counters()
  undefined         ⚠<UNASSIGNED>   <RETURN>
                    Increment Software RTC Counters (Internal Time Keeping)
                    Updates software-based time counters every 333ms independent …

                    IMPORTANT: These are NOT MM58274 hardware registers!
                    Variable names are misleading - they represent software time …
                    that increment based on timer interrupts, not actual RTC chip…

                    SOFTWARE TIME REPRESENTATION:
                    - Software_RTC_Units_Hours (0x25): Actually tenths of seconds…
                    - Software_RTC_Tens_Minutes (0x24): Actually seconds (0-59)
                    - Software_RTC_Units_Minutes (0x23): Actually minutes (0-23, …
                    - Software_RTC_Tens_Seconds (0x22): Actually days (1-31)
                    - Software_RTC_Units_Seconds (0x21): Actually months (1-12)
                    - Software_RTC_Tenths_Seconds (0x20): Actually years
                    - Software_RTC_Tens_Hours (0x26): Actually leap year cycle (0…

                    TIME INCREMENT LOGIC:
                    1. Increment tenths of seconds (0x25) every 333ms
                    2. Carry propagation: 60 tenths → 1 second
                    3. Carry propagation: 60 seconds → 1 minute
                    4. Carry propagation: 24 hours → 1 day
                    5. Days-in-month lookup using ROM table at 0x0E37
                    6. Leap year handling for February (28/29 days)
                    7. Month rollover: 12 months → 1 year
                    8. Leap year cycle: 4 years → reset

                    DISPLAY INTEGRATION:
                    These software counters provide time data for panel display
                    when time-related PANC commands are processed or when
                    automatic time display mode is active.
                    Increment_Software_RTC_Counters                XREF[1]:     Timer_1200Hz_CPU_Performance_Mon…
  08f9 a6 3c             LDA        #0x3c
  08fb 3c 25             INC        Software_RTC_Units_Hours
  08fd b1 25             CMP        Software_RTC_Units_Hours
  08ff 22 46             BHI        LAB_0947
  0901 3f 25             CLR        Software_RTC_Units_Hours
  0903 a6 3c             LDA        #0x3c
  0905 3c 24             INC        Software_RTC_Tens_Minutes
  0907 b1 24             CMP        Software_RTC_Tens_Minutes
  0909 22 3c             BHI        LAB_0947
  090b 3f 24             CLR        Software_RTC_Tens_Minutes
  090d a6 18             LDA        #0x18
  090f 3c 23             INC        Software_RTC_Units_Minutes
  0911 b1 23             CMP        Software_RTC_Units_Minutes
  0913 22 32             BHI        LAB_0947
  0915 3f 23             CLR        Software_RTC_Units_Minutes
  0917 be 21             LDX        Software_RTC_Units_Seconds
  0919 d6 0e 37          LDA        DAT_0e37,X
  091c a1 1c             CMP        #0x1c
  091e 26 05             BNE        LAB_0925
  0920 3d 26             TST        MM58274_Tens_Hours
  0922 26 01             BNE        LAB_0925
  0924 4c                INCA

                    LAB_0925                                       XREF[2]:     091e(j), 0922(j)
  0925 3c 22             INC        Software_RTC_Tens_Seconds
  0927 b1 22             CMP        Software_RTC_Tens_Seconds
  0929 24 1c             BCC        LAB_0947
  092b a6 01             LDA        #0x1
  092d b7 22             STA        Software_RTC_Tens_Seconds
  092f a6 0c             LDA        #0xc
  0931 3c 21             INC        Software_RTC_Units_Seconds
  0933 b1 21             CMP        Software_RTC_Units_Seconds
  0935 22 10             BHI        LAB_0947
  0937 a6 01             LDA        #0x1
  0939 b7 21             STA        Software_RTC_Units_Seconds
  093b 3c 20             INC        Software_RTC_Tenths_Seconds
  093d 3c 26             INC        MM58274_Tens_Hours
  093f b6 26             LDA        MM58274_Tens_Hours
  0941 a1 03             CMP        #0x3
  0943 23 02             BLS        LAB_0947
  0945 3f 26             CLR        MM58274_Tens_Hours

                    LAB_0947                                       XREF[6]:     08ff(j), 0909(j), 0913(j),
                                                                                0929(j), 0935(j), 0943(j)
  0947 81                RTS
```

```
                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined Shift_CPU_Utilization_History_Buffer()
     undefined        ⚠<UNASSIGNED>  <RETURN>
                    Shift CPU Utilization History Buffer (15-byte sliding window)
                    Maintains rolling history of CPU performance data for statist…

                    BUFFER ORGANIZATION:
                    - 15-byte circular buffer: 0x61 (CPU_Utilization_History_Buff…
                    - Data flow: oldest → 0x6F → 0x6E → ... → 0x62 → 0x61 ← ne…
                    - Called every 13.3ms (16 timer cycles at 1200Hz)

                    SHIFTING OPERATION:
                    Each call shifts all data one position toward buffer end:
                    - CPU_Utilization_History_Buffer_End (0x6F) ← 0x6E ← 0x6D ←…
                    - Oldest data at 0x6F is discarded
                    - New data enters at CPU_Utilization_Buffer_Input (0x5E) → 0…

                    SLIDING WINDOW PURPOSE:
                    - Provides 200ms history window (15 samples × 13.3ms)
                    - Smooths performance measurements over time
                    - Enables trend analysis and statistical calculations
                    - Reduces noise in utilization calculations
                    - Supports display filtering for stable readouts

                    PERFORMANCE DATA TYPES:
                    Buffer contains accumulated CPU performance metrics:
                    - Ring level distribution (time in each privilege level)
                    - PONI/IONI status activity (memory protection/interrupt stat…
                    - Combined status indicators for system load analysis

                    TIMING RELATIONSHIP:
                    - Input rate: 75Hz (every 13.3ms)
                    - Window duration: ~200ms of CPU history
                    - Update frequency: Continuous during all operations
                    - Independence: Runs during both IDLE and ACTIVE states

                    STATISTICAL USAGE:
                    Historical data used by Process_Performance_Statistics() and
                    Calculate_CPU_Utilization_Percentage() to generate stable,
                    averaged CPU utilization metrics for operator display and
                    system performance analysis.
                    Shift_CPU_Utilization_History_Buffer           XREF[1]:     Timer_1200Hz_CPU_Performance_Mon…
0948 b6 6c           LDA        BYTE_006c
094a b7 6f           STA        CPU_Utilization_History_Buffer_End
094c b6 6b           LDA        BYTE_006b
094e b7 6e           STA        BYTE_006e
0950 b6 6a           LDA        BYTE_006a
0952 b7 6d           STA        BYTE_006d
0954 b6 69           LDA        BYTE_0069
0956 b7 6c           STA        BYTE_006c
0958 b6 68           LDA        BYTE_0068
095a b7 6b           STA        BYTE_006b
095c b6 67           LDA        BYTE_0067
095e b7 6a           STA        BYTE_006a
0960 b6 66           LDA        BYTE_0066
0962 b7 69           STA        BYTE_0069
0964 b6 65           LDA        BYTE_0065
0966 b7 68           STA        BYTE_0068
0968 b6 64           LDA        BYTE_0064
096a b7 67           STA        BYTE_0067
096c b6 63           LDA        BYTE_0063
096e b7 66           STA        BYTE_0066
0970 b6 62           LDA        BYTE_0062
0972 b7 65           STA        BYTE_0065
0974 b6 61           LDA        CPU_Utilization_History_Buffer_Start
0976 b7 64           STA        BYTE_0064
0978 b6 60           LDA        CPU_Status_Accumulator
097a b7 63           STA        BYTE_0063
097c b6 5f           LDA        BYTE_005f
097e b7 62           STA        BYTE_0062
0980 b6 5e           LDA        CPU_Utilization_Buffer_Input
0982 b7 61           STA        CPU_Utilization_History_Buffer_Start
0984 81              RTS
```

```
                         **************************************************************
                         *                          FUNCTION                          *
                         **************************************************************
                         undefined Monitor_LEV0_LHIT_Signals_Update_Display()
         undefined         ⚠<UNASSIGNED>   <RETURN>
                         Monitor LEV0 and LHIT Signals for Performance Display
                         Tracks ND-120 CPU idle time and cache performance over ~107ms…

                         SIGNAL MONITORING (enabled when Display_Control_Mode_Flags bi…
                         - LEV0 (PD5): CPU Level 0 indicator (HIGH = Idle, LOW = Busy …
                         - LHIT (PD4): Cache hit indicator (HIGH = cache hit, LOW = ca…

                         SAMPLING WINDOW:
                         - 128 timer cycles at 1200Hz = ~106.7ms measurement window
                         - LEV0_LHIT_Sample_Counter_128 counts down from 128 to 0

                         ACCUMULATION LOGIC:
                         - LHIT_Cache_Hit_Accumulator++: Count cache hits (PD4 HIGH)
                         - LEV0_CPU_Busy_Time_Accumulator++: Count CPU busy time (PD5 …

                         DISPLAY UPDATE (every ~107ms):
                         When counter reaches 0:
                         1. Calculate performance metrics from accumulators
                         2. Convert to display characters (count >> 5) + 1 + 0x21
                         3. Update panel display:
                            - RAM_0011: Cache hit rate indicator
                            - RAM_0010: CPU utilization indicator
                         4. Shift accumulators right for next window

                         CPU UTILIZATION CALCULATION:
                         Busy% = (LEV0_CPU_Busy_Time_Accumulator / 128) × 100
                         Idle% = 100 - Busy%

                         Note: LEV0 signal is inverted in logic - we count when it's L…
                         Monitor_LEV0_LHIT_Signals_Update_Display       XREF[1]:     Timer_1200Hz_CPU_Performance_Mon…
        0985 3a 71           DEC        LEV0_LHIT_Sample_Counter_128
        0987 26 26           BNE        LAB_09af
        0989 a6 80           LDA        #0x80
        098b b7 71           STA        LEV0_LHIT_Sample_Counter_128
        098d b6 72           LDA        LHIT_Cache_Hit_Accumulator
        098f 4d              TSTA
        0990 27 08           BEQ        LAB_099a
        0992 44              LSRA
        0993 b7 72           STA        LHIT_Cache_Hit_Accumulator
        0995 44              LSRA
        0996 44              LSRA
        0997 44              LSRA
        0998 44              LSRA
        0999 4c              INCA

                         LAB_099a                                       XREF[1]:     0990(j)
        099a ab 21           ADD        #0x21
        099c b7 11           STA        RAM_0011
        099e b6 73           LDA        LEV0_CPU_Busy_Time_Accumulator
        09a0 4d              TSTA
        09a1 27 08           BEQ        LAB_09ab
        09a3 44              LSRA
        09a4 b7 73           STA        LEV0_CPU_Busy_Time_Accumulator
        09a6 44              LSRA
        09a7 44              LSRA
        09a8 44              LSRA
        09a9 44              LSRA
        09aa 4c              INCA

                         LAB_09ab                                       XREF[1]:     09a1(j)
        09ab ab 21           ADD        #0x21
        09ad b7 10           STA        RAM_0010

                         LAB_09af                                       XREF[1]:     0987(j)
        09af 09 03 02        BRCLR      0x4,PORTD,LAB_09b4
        09b2 3c 72           INC        LHIT_Cache_Hit_Accumulator

                         LAB_09b4                                       XREF[1]:     09af(j)
        09b4 0a 03 02        BRSET      0x5,PORTD,LAB_09b9
        09b7 3c 73           INC        LEV0_CPU_Busy_Time_Accumulator

                         LAB_09b9                                       XREF[1]:     09b4(j)
        09b9 81              RTS

                         SWI_INTERRUPT                                  XREF[1]:     0ffc(*)
        09ba 9d              NOP
        09bb 80              RTI
```

```
                     ****************************************************************
                     *                          FUNCTION                            *
                     ****************************************************************
                     undefined Strobe_WMM_Read_From_FIFO()
   undefined         ⚠<UNASSIGNED>   <RETURN>
                     Read Command Data from FIFO (LDPANC Microcode Interface)
                     Reads data loaded by ND-120 LDPANC microcode into hardware FI…

                     ND-120 MICROCODE COMMAND LOADING:
                     1. ND-120 executes microcode 06.2 = LDPANC
                     2. Lower byte of IDB loaded into MC68705 FIFO queue
                     3. /EMP_n signal set HIGH indicating FIFO contains data
                     4. MC68705 detects /EMP_n and calls this function

                     MICROCODE vs HIGH-LEVEL INSTRUCTIONS:
                     - LDPANC microcode: Low-level FIFO loading operation
                     - TRR PANC instruction: High-level instruction using LDPANC
                     - Multiple LDPANC operations can build multi-byte commands

                     HARDWARE INTERFACE:
                     - IDB[7:0]: ND-120 data bus carrying command/parameter bytes
                     - FIFO Queue: Hardware buffer between ND-120 and MC68705
                     - PA[7:0]: MC68705 Port A connected to FIFO output via 74LS374
                     - PB3: WMM strobe signal for FIFO read coordination

                     READ SEQUENCE:
                     1. PORTB &= 0xF7: Clear PB3 (prepare read strobe)
                     2. PORTB |= 0x08: Set PB3 (assert WMM strobe to FIFO)
                     3. return PORTA: Read FIFO data from PA bus

                     FIFO DATA ORGANIZATION:
                     First LDPANC: Command byte
                     - Bits 7-4: Command flags and options
                     - Bit 3: Data direction
                     - Bits 2-0: Command type (0-7)

                     Subsequent LDPANC: Parameter bytes
                     - Multi-byte commands require sequential LDPANC operations
                     - FIFO queues all bytes for sequential MC68705 processing

                     TIMING:
                     WMM strobe ensures proper data transfer timing between
                     FIFO clock domain and MC68705 4MHz execution domain.

                     ASYNCHRONOUS OPERATION:
                     ND-120 can execute multiple LDPANC operations to queue
                     commands while MC68705 processes previous data in background.
                     Strobe_WMM_Read_From_FIFO                 XREF[25]:    Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Process_PANC_Command_From_FIFO:0…
                                                                            Output_RTC_Time_Data_To_ND120:06…
                                                                            [more]
  09bc 9b              SEI
  09bd 17 01           BCLR       0x3,PORTB
  09bf b6 00           LDA        PORTA
  09c1 16 01           BSET       0x3,PORTB
  09c3 9a              CLI
  09c4 81              RTS
```

```
                    **************************************************************
                    *                          FUNCTION                          *
                    **************************************************************
                    undefined Output_Response_To_ND120_IDB()
    undefined         ⚠<UNASSIGNED>   <RETURN>
                    Output Response Data for EPANS Microcode Reading
                    Provides response data when ND-120 uses EPANS microcode to re…

                    ND-120 MICROCODE RESPONSE READING:
                    The ND-120 can read MC68705 responses using microcode:

                    IDB Source 20 (octal) = EPANS (Enable Panel Status):
                    - Sets /EMP_n HIGH to enable panel status register
                    - MC68705 Port A data flows to IDB for ND-120 reading
                    - Does not affect status flags

                    IDB Source 21 (octal) = EPANS + Reset RDY:
                    - Same as source 20 but also resets PANS bit 12 (RDY)
                    - Acknowledges command completion
                    - Clears "operation complete" status flag

                    RESPONSE DATA PATH:
                    MC68705 Port A → 74LS374 Latch → IDB[7:0] → ND-120 Register

                    OUTPUT SEQUENCE:
                    1. PORTA = response_data: Place data on Port A bus
                    2. PORTB &= 0xFE: Clear PB0 (prepare WMM_n latch signal)
                    3. PORTB |= 0x01: Set PB0 (strobe WMM_n to latch data)
                    4. DDRA = 0: Return Port A to input mode

                    RESPONSE DATA TYPES:
                    - Time/date values from software RTC or MM58274
                    - CPU performance statistics (utilization, cache efficiency)
                    - System status information (PONI/IONI/Ring levels)
                    - Configuration acknowledgments and completion codes
                    - Error status or diagnostic information

                    MICROCODE TIMING:
                    When ND-120 executes EPANS microcode:
                    1. /EMP_n goes HIGH (triggers MC68705 if in polling loop)
                    2. MC68705 outputs current response data via this function
                    3. ND-120 microcode reads IDB containing Port A data
                    4. IDB Source 21 resets RDY flag acknowledging read

                    INTEGRATION WITH HIGH-LEVEL INSTRUCTIONS:
                    - TRA PANS instruction uses EPANS microcode internally
                    - Provides seamless interface between assembly and microcode
                    - Enables both programmed I/O and interrupt-driven access

                    ASYNCHRONOUS CAPABILITY:
                    ND-120 can read status at any time using EPANS microcode,
                    even interrupting command processing if necessary.
                    Output_Response_To_ND120_IDB                    XREF[2]:     Output_RTC_Time_Data_To_ND120:06…
                                                                                 Output_RTC_Time_Data_To_ND120:07…
  09c5 9b                SEI
  09c6 b7 00             STA          PORTA
  09c8 a6 ff             LDA          #0xff
  09ca b7 04             STA          DDRA
  09cc 11 01             BCLR         0x0,PORTB
  09ce 10 01             BSET         0x0,PORTB
  09d0 3f 04             CLR          DDRA
  09d2 9a                CLI
  09d3 81                RTS
```

```
                     **************************************************************
                     *                          FUNCTION                          *
                     **************************************************************
                     undefined Sample_ND120_CPU_Status_Signals()
     undefined         ⚠<UNASSIGNED>  <RETURN>
                     Sample ND-120 CPU Status Signals (1200Hz)
                     Reads and processes ND-120 CPU state signals for performance …

                     PORT D SIGNAL SAMPLING:
                     PD7: /EMP_n - FIFO Empty (LOW=empty) or Panel Enable (HIGH=ac…
                     PD6: GND - Tied to ground (unused)
                     PD5: LEV0 - CPU Level 0 active (HIGH=idle, LOW=busy levels 1-…
                     PD4: LHIT - Cache hit indicator (HIGH=cache hit occurred)
                     PD3: IONI - Interrupt system ON (SINTRAN OS controlled)
                     PD2: PONI - Memory protection ON (SINTRAN OS controlled)
                     PD[1:0]: PCR Ring level (0-3) - Current CPU privilege level

                     CPU RING LEVEL PROCESSING:
                     1. Extract PONI/IONI status: (PORTD & 0x0C) << 2
                     2. If PONI active (memory protection ON):
                        - Read PCR ring level from PORTD[1:0]
                        - Use as index into PCR_Ring_Level_Bit_Lookup_Table at 0x0…
                        - Table contains: [1, 2, 4, 8] = bit masks for ring levels
                        - Set corresponding bit: 1 << ring_level
                       - Accumulate in ND120_CPU_Status_Ring_PONI_IONI

                     RING LEVEL INTERPRETATION:
                     Ring 0: User programs (least privileged) - bit 0
                     Ring 1: System services - bit 1
                     Ring 2: Kernel operations - bit 2
                     Ring 3: Supervisor mode (most privileged) - bit 3

                     PCR_Ring_Level_Bit_Lookup_Table:
                     Index 0 (Ring 0): Value 1 (0x01) - User privilege
                     Index 1 (Ring 1): Value 2 (0x02) - System privilege
                     Index 2 (Ring 2): Value 4 (0x04) - Kernel privilege
                     Index 3 (Ring 3): Value 8 (0x08) - Supervisor privilege

                     PERFORMANCE ANALYSIS:
                     - Higher ring levels indicate more critical system activity
                     - Ring distribution shows CPU workload characteristics
                     - PONI/IONI status indicates SINTRAN OS protection state
                     - Combined with LEV0 signal provides complete CPU utilization…

                     SAMPLE RATE: 1200Hz (833μs period) for high-resolution monito…
                     Statistical data accumulated for utilization calculations and…
                     Sample_ND120_CPU_Status_Signals                XREF[1]:     Timer_1200Hz_CPU_Performance_Mo…
    09d4 b6 03           LDA        PORTD
    09d6 b7 70           STA        PortD_SavedValue
    09d8 a4 0c           AND        #0xc
    09da 48              ASLA
    09db 48              ASLA
    09dc b7 19           STA        ND120_CPU_Status_Ring_PONI_IONI
                     Value 4 is 0100. The logic checks if PB2 != 0, which is PONI …
                     If PONI = TRUE, read memory [0x0f0c + PCR] (PB0/PB1) and OR t…
                     In memory starting at 0x0F0C we have the values
                     1
                     2
                     4
                     8

                     In effect, the logic is that

                     IF (PONI=TRUE)
                     {
                        PortD_PONI_IONI_BIT_4_5 = 1<<PCR[1:0];  // Set bit 0 to 3 …
                     }

    09de 05 70 0c        BRCLR      0x2,PortD_SavedValue,LAB_09ed
    09e1 b6 70           LDA        PortD_SavedValue
    09e3 a4 03           AND        #0x3
    09e5 97              TAX
    09e6 d6 0f 0c        LDA        PCR_Ring_Level_Bit_Lookup_Table,X                = 1h
    09e9 ba 19           ORA        ND120_CPU_Status_Ring_PONI_IONI
    09eb b7 19           STA        ND120_CPU_Status_Ring_PONI_IONI

                     LAB_09ed                                       XREF[1]:     09de(j)
    09ed 81              RTS
```

```
                ***************************************************************
                *                          FUNCTION                           *
                ***************************************************************
                undefined Output_Status_Code_1_To_ND120()
  undefined       ⚠<UNASSIGNED>   <RETURN>
                Output Status Code 1 to ND-120 CPU
                Sends specific status response (value 1) to ND-120 via IDB in…

                HARDWARE SEQUENCE:
                1. PORTB &= 0xFD: Clear PB1 (prepare different strobe signal)
                2. PORTA = 1: Output status code 1 to PA bus
                3. PORTB |= 0x02: Set PB1 (strobe response latch)
                4. DDRA = 0: Return PA to input mode

                ALTERNATE LATCH CONTROL:
                Uses PB1 instead of PB0 for response latching, suggesting
                this is a special status response different from normal
                command responses that use PB0 (WMM_n signal).

                PURPOSE:
                Sends completion status or specific response code back
                to ND-120 for special operations, particularly after
                complex time calculations or RTC operations.

                USAGE:
                Called after time conversion operations to indicate
                successful completion with status code 1.
                Output_Status_Code_1_To_ND120                  XREF[1]:     Output_RTC_Time_Data_To_ND120:06…
09ee a6 ff          LDA         #0xff
09f0 b7 04          STA         DDRA
09f2 a6 0f          LDA         #0xf
09f4 b7 00          STA         PORTA
09f6 13 01          BCLR        0x1,PORTB
09f8 12 01          BSET        0x1,PORTB
09fa a6 f0          LDA         #0xf0
09fc b7 00          STA         PORTA
09fe 13 01          BCLR        0x1,PORTB
0a00 12 01          BSET        0x1,PORTB
0a02 a6 05          LDA         #0x5
0a04 b7 00          STA         PORTA
0a06 13 01          BCLR        0x1,PORTB
0a08 12 01          BSET        0x1,PORTB
0a0a b6 26          LDA         MM58274_Tens_Hours
0a0c a4 03          AND         #0x3
0a0e 48             ASLA
0a0f 48             ASLA
0a10 aa f1          ORA         #0xf1
0a12 b7 00          STA         PORTA
0a14 13 01          BCLR        0x1,PORTB
0a16 12 01          BSET        0x1,PORTB
0a18 be 20          LDX         Software_RTC_Tenths_Seconds
0a1a 58             ASLX
0a1b d6 0e 44       LDA         DAT_0e44,X
0a1e aa d0          ORA         #0xd0
0a20 b7 00          STA         PORTA
0a22 13 01          BCLR        0x1,PORTB
0a24 12 01          BSET        0x1,PORTB
0a26 5c             INCX
0a27 d6 0e 44       LDA         DAT_0e44,X
0a2a aa c0          ORA         #0xc0
0a2c b7 00          STA         PORTA
0a2e 13 01          BCLR        0x1,PORTB
0a30 12 01          BSET        0x1,PORTB
0a32 be 21          LDX         Software_RTC_Units_Seconds
0a34 58             ASLX
0a35 d6 0e 44       LDA         DAT_0e44,X
0a38 aa b0          ORA         #0xb0
0a3a b7 00          STA         PORTA
0a3c 13 01          BCLR        0x1,PORTB
0a3e 12 01          BSET        0x1,PORTB
0a40 5c             INCX
0a41 d6 0e 44       LDA         DAT_0e44,X
0a44 aa a0          ORA         #0xa0
0a46 b7 00          STA         PORTA
0a48 13 01          BCLR        0x1,PORTB
0a4a 12 01          BSET        0x1,PORTB
0a4c be 22          LDX         Software_RTC_Tens_Seconds
0a4e 58             ASLX
0a4f d6 0e 44       LDA         DAT_0e44,X
0a52 aa 90          ORA         #0x90
0a54 b7 00          STA         PORTA
0a56 13 01          BCLR        0x1,PORTB
0a58 12 01          BSET        0x1,PORTB
0a5a 5c             INCX
```

```
0a5b d6 0e 44       LDA        DAT_0e44,X
0a5e aa 80          ORA        #0x80
0a60 b7 00          STA        PORTA
0a62 13 01          BCLR       0x1,PORTB
0a64 12 01          BSET       0x1,PORTB
0a66 be 23          LDX        Software_RTC_Units_Minutes
0a68 58             ASLX
0a69 d6 0e 44       LDA        DAT_0e44,X
0a6c aa 70          ORA        #0x70
0a6e b7 00          STA        PORTA
0a70 13 01          BCLR       0x1,PORTB
0a72 12 01          BSET       0x1,PORTB
0a74 5c             INCX
0a75 d6 0e 44       LDA        DAT_0e44,X
0a78 aa 60          ORA        #0x60
0a7a b7 00          STA        PORTA
0a7c 13 01          BCLR       0x1,PORTB
0a7e 12 01          BSET       0x1,PORTB
0a80 be 24          LDX        Software_RTC_Tens_Minutes
0a82 58             ASLX
0a83 d6 0e 44       LDA        DAT_0e44,X
0a86 aa 50          ORA        #0x50
0a88 b7 00          STA        PORTA
0a8a 13 01          BCLR       0x1,PORTB
0a8c 12 01          BSET       0x1,PORTB
0a8e 5c             INCX
0a8f d6 0e 44       LDA        DAT_0e44,X
0a92 aa 40          ORA        #0x40
0a94 b7 00          STA        PORTA
0a96 13 01          BCLR       0x1,PORTB
0a98 12 01          BSET       0x1,PORTB
0a9a be 25          LDX        Software_RTC_Units_Hours
0a9c 58             ASLX
0a9d d6 0e 44       LDA        DAT_0e44,X
0aa0 aa 30          ORA        #0x30
0aa2 b7 00          STA        PORTA
0aa4 13 01          BCLR       0x1,PORTB
0aa6 12 01          BSET       0x1,PORTB
0aa8 be 25          LDX        Software_RTC_Units_Hours
0aaa d6 0e 44       LDA        DAT_0e44,X
0aad aa 20          ORA        #0x20
0aaf b7 00          STA        PORTA
0ab1 13 01          BCLR       0x1,PORTB
0ab3 12 01          BSET       0x1,PORTB
0ab5 a6 01          LDA        #0x1
0ab7 b7 00          STA        PORTA
0ab9 13 01          BCLR       0x1,PORTB
0abb 12 01          BSET       0x1,PORTB
0abd 3f 04          CLR        DDRA
0abf 81             RTS

                 **************************************************************
                 *                          FUNCTION                          *
                 **************************************************************
                 undefined Initialize_MM58274_RTC_Chip()
 undefined         ⚠<UNASSIGNED>   <RETURN>
                 Initialize MM58274 Real-Time Clock Chip
                 Resets and configures the MM58274 RTC for operation.

                 Hardware Control:
                 - PB2 (/ROCLK) controls MM58274: 0=Reset, 1=Enable

                 Initialization Sequence:
                 1. PORTB &= 0xFB (clear PB2 - assert /ROCLK low = reset RTC)
                 2. Clear all MM58274 register buffers (0x20-0x26, 0x52)
                 3. PORTA = 0 (clear data bus)
                 4. PORTB |= 0x04 (set PB2 - deassert /ROCLK high = enable RTC)
                 5. DDRA = 0 (ensure PA is input)

                 Buffer Clearing:
                 - 0x26: MM58274_Tens_Hours
                 - 0x20: MM58274_Tenths_Seconds
                 - 0x21-0x25: Other MM58274 time registers
                 - 0x52: Segment display buffer
                 Initialize_MM58274_RTC_Chip                  XREF[3]:     Wait_For_FIFO_Commands_And_Proce..
                                                                           Output_RTC_Time_Data_To_ND120:07..
                                                                           0bd8(j)
0ac0 a6 f0          LDA        #0xf0
0ac2 b7 04          STA        DDRA
0ac4 a6 00          LDA        #0x0
0ac6 b7 00          STA        PORTA
0ac8 15 01          BCLR       0x2,PORTB
0aca 14 01          BSET       0x2,PORTB
0acc a6 f0          LDA        #0xf0
0ace b7 00          STA        PORTA
```

```
0ad0 15 01      BCLR    0x2,PORTB
0ad2 b6 00      LDA     PORTA
0ad4 14 01      BSET    0x2,PORTB
0ad6 a4 0f      AND     #0xf
0ad8 44         LSRA
0ad9 44         LSRA
0ada b7 26      STA     MM58274_Tens_Hours
0adc a6 c0      LDA     #0xc0
0ade b7 00      STA     PORTA
0ae0 15 01      BCLR    0x2,PORTB
0ae2 b6 00      LDA     PORTA
0ae4 14 01      BSET    0x2,PORTB
0ae6 a4 0f      AND     #0xf
0ae8 b7 20      STA     Software_RTC_Tenths_Seconds
0aea a6 d0      LDA     #0xd0
0aec b7 00      STA     PORTA
0aee 15 01      BCLR    0x2,PORTB
0af0 b6 00      LDA     PORTA
0af2 14 01      BSET    0x2,PORTB
0af4 a4 0f      AND     #0xf
0af6 97         TAX
0af7 58         ASLX
0af8 58         ASLX
0af9 58         ASLX
0afa bf 52      STX     Display_7Segment_Digit_0
0afc 48         ASLA
0afd bb 52      ADD     Display_7Segment_Digit_0
0aff bb 20      ADD     Software_RTC_Tenths_Seconds
0b01 b7 20      STA     Software_RTC_Tenths_Seconds
0b03 a6 a0      LDA     #0xa0
0b05 b7 00      STA     PORTA
0b07 15 01      BCLR    0x2,PORTB
0b09 b6 00      LDA     PORTA
0b0b 14 01      BSET    0x2,PORTB
0b0d a4 0f      AND     #0xf
0b0f b7 21      STA     Software_RTC_Units_Seconds
0b11 a6 b0      LDA     #0xb0
0b13 b7 00      STA     PORTA
0b15 15 01      BCLR    0x2,PORTB
0b17 b6 00      LDA     PORTA
0b19 14 01      BSET    0x2,PORTB
0b1b a4 0f      AND     #0xf
0b1d 97         TAX
0b1e 58         ASLX
0b1f 58         ASLX
0b20 58         ASLX
0b21 bf 52      STX     Display_7Segment_Digit_0
0b23 48         ASLA
0b24 bb 52      ADD     Display_7Segment_Digit_0
0b26 bb 21      ADD     Software_RTC_Units_Seconds
0b28 b7 21      STA     Software_RTC_Units_Seconds
0b2a a6 80      LDA     #0x80
0b2c b7 00      STA     PORTA
0b2e 15 01      BCLR    0x2,PORTB
0b30 b6 00      LDA     PORTA
0b32 14 01      BSET    0x2,PORTB
0b34 a4 0f      AND     #0xf
0b36 b7 22      STA     Software_RTC_Tens_Seconds
0b38 a6 90      LDA     #0x90
0b3a b7 00      STA     PORTA
0b3c 15 01      BCLR    0x2,PORTB
0b3e b6 00      LDA     PORTA
0b40 14 01      BSET    0x2,PORTB
0b42 a4 0f      AND     #0xf
0b44 97         TAX
0b45 58         ASLX
0b46 58         ASLX
0b47 58         ASLX
0b48 bf 52      STX     Display_7Segment_Digit_0
0b4a 48         ASLA
0b4b bb 52      ADD     Display_7Segment_Digit_0
0b4d bb 22      ADD     Software_RTC_Tens_Seconds
0b4f b7 22      STA     Software_RTC_Tens_Seconds
0b51 a6 60      LDA     #0x60
0b53 b7 00      STA     PORTA
0b55 15 01      BCLR    0x2,PORTB
0b57 b6 00      LDA     PORTA
0b59 14 01      BSET    0x2,PORTB
0b5b a4 0f      AND     #0xf
0b5d b7 23      STA     Software_RTC_Units_Minutes
0b5f a6 70      LDA     #0x70
0b61 b7 00      STA     PORTA
0b63 15 01      BCLR    0x2,PORTB
0b65 b6 00      LDA     PORTA
0b67 14 01      BSET    0x2,PORTB
```

```
0b69 a4 0f          AND         #0xf
0b6b 97             TAX
0b6c 58             ASLX
0b6d 58             ASLX
0b6e 58             ASLX
0b6f bf 52          STX         Display_7Segment_Digit_0
0b71 48             ASLA
0b72 bb 52          ADD         Display_7Segment_Digit_0
0b74 bb 23          ADD         Software_RTC_Units_Minutes
0b76 b7 23          STA         Software_RTC_Units_Minutes
0b78 a6 40          LDA         #0x40
0b7a b7 00          STA         PORTA
0b7c 15 01          BCLR        0x2,PORTB
0b7e b6 00          LDA         PORTA
0b80 14 01          BSET        0x2,PORTB
0b82 a4 0f          AND         #0xf
0b84 b7 24          STA         Software_RTC_Tens_Minutes
0b86 a6 50          LDA         #0x50
0b88 b7 00          STA         PORTA
0b8a 15 01          BCLR        0x2,PORTB
0b8c b6 00          LDA         PORTA
0b8e 14 01          BSET        0x2,PORTB
0b90 a4 0f          AND         #0xf
0b92 97             TAX
0b93 58             ASLX
0b94 58             ASLX
0b95 58             ASLX
0b96 bf 52          STX         Display_7Segment_Digit_0
0b98 48             ASLA
0b99 bb 52          ADD         Display_7Segment_Digit_0
0b9b bb 24          ADD         Software_RTC_Tens_Minutes
0b9d b7 24          STA         Software_RTC_Tens_Minutes
0b9f a6 20          LDA         #0x20
0ba1 b7 00          STA         PORTA
0ba3 15 01          BCLR        0x2,PORTB
0ba5 b6 00          LDA         PORTA
0ba7 14 01          BSET        0x2,PORTB
0ba9 a4 0f          AND         #0xf
0bab b7 25          STA         Software_RTC_Units_Hours
0bad a6 30          LDA         #0x30
0baf b7 00          STA         PORTA
0bb1 15 01          BCLR        0x2,PORTB
0bb3 b6 00          LDA         PORTA
0bb5 14 01          BSET        0x2,PORTB
0bb7 a4 0f          AND         #0xf
0bb9 97             TAX
0bba 58             ASLX
0bbb 58             ASLX
0bbc 58             ASLX
0bbd bf 52          STX         Display_7Segment_Digit_0
0bbf 48             ASLA
0bc0 bb 52          ADD         Display_7Segment_Digit_0
0bc2 bb 25          ADD         Software_RTC_Units_Hours
0bc4 b7 25          STA         Software_RTC_Units_Hours
0bc6 a6 00          LDA         #0x0
0bc8 b7 00          STA         PORTA
0bca 15 01          BCLR        0x2,PORTB
0bcc b6 00          LDA         PORTA
0bce 14 01          BSET        0x2,PORTB
0bd0 a4 0f          AND         #0xf
0bd2 a4 08          AND         #0x8
0bd4 a1 08          CMP         #0x8
0bd6 26 03          BNE         LAB_0bdb
0bd8 cc 0a c0       JMP         Initialize_MM58274_RTC_Chip

                 LAB_0bdb                                    XREF[1]:     0bd6(j)
0bdb 3f 04          CLR         DDRA
0bdd 81             RTS
0bde 00             ??          00h
0bdf 00             ??          00h
0be0 00             ??          00h
0be1 00             ??          00h
0be2 00             ??          00h
0be3 00             ??          00h
0be4 00             ??          00h
0be5 00             ??          00h
0be6 00             ??          00h
0be7 00             ??          00h
0be8 00             ??          00h
0be9 00             ??          00h
0bea 00             ??          00h
0beb 00             ??          00h
0bec 00             ??          00h
0bed 00             ??          00h
0bee 00             ??          00h
```

```
0bef 00             ??         00h
0bf0 00             ??         00h
0bf1 00             ??         00h
0bf2 00             ??         00h
0bf3 00             ??         00h
0bf4 00             ??         00h
0bf5 00             ??         00h
0bf6 00             ??         00h
0bf7 00             ??         00h
0bf8 00             ??         00h
0bf9 00             ??         00h
0bfa 00             ??         00h
0bfb 00             ??         00h
0bfc 00             ??         00h
0bfd 00             ??         00h
0bfe 00             ??         00h
0bff 00             ??         00h
0c00 00             ??         00h
0c01 00             ??         00h
0c02 00             ??         00h
0c03 00             ??         00h
0c04 00             ??         00h
0c05 00             ??         00h
0c06 00             ??         00h
0c07 00             ??         00h
0c08 00             ??         00h
0c09 00             ??         00h
0c0a 00             ??         00h
0c0b 00             ??         00h
0c0c 00             ??         00h
0c0d 00             ??         00h
0c0e 00             ??         00h
0c0f 00             ??         00h
0c10 00             ??         00h
0c11 00             ??         00h
0c12 00             ??         00h
0c13 00             ??         00h
0c14 00             ??         00h
0c15 00             ??         00h
0c16 00             ??         00h
0c17 00             ??         00h
0c18 00             ??         00h
0c19 00             ??         00h
0c1a 00             ??         00h
0c1b 00             ??         00h
0c1c 00             ??         00h
0c1d 00             ??         00h
0c1e 00             ??         00h
0c1f 00             ??         00h
0c20 00             ??         00h
0c21 00             ??         00h
0c22 00             ??         00h
0c23 00             ??         00h
0c24 00             ??         00h
0c25 00             ??         00h
0c26 00             ??         00h
0c27 00             ??         00h
0c28 00             ??         00h
0c29 00             ??         00h
0c2a 00             ??         00h
0c2b 00             ??         00h
0c2c 00             ??         00h
0c2d 00             ??         00h
0c2e 00             ??         00h
0c2f 00             ??         00h
0c30 00             ??         00h
0c31 00             ??         00h
0c32 00             ??         00h
0c33 00             ??         00h
0c34 00             ??         00h
0c35 00             ??         00h
0c36 00             ??         00h
0c37 00             ??         00h
0c38 00             ??         00h
0c39 00             ??         00h
0c3a 00             ??         00h
0c3b 00             ??         00h
0c3c 00             ??         00h
0c3d 00             ??         00h
0c3e 00             ??         00h
0c3f 00             ??         00h
0c40 00             ??         00h
0c41 00             ??         00h
0c42 00             ??         00h
0c43 00             ??         00h
```

```
0c44 00             ??          00h
0c45 00             ??          00h
0c46 00             ??          00h
0c47 00             ??          00h
0c48 00             ??          00h
0c49 00             ??          00h
0c4a 00             ??          00h
0c4b 00             ??          00h
0c4c 00             ??          00h
0c4d 00             ??          00h
0c4e 00             ??          00h
0c4f 00             ??          00h
0c50 00             ??          00h
0c51 00             ??          00h
0c52 00             ??          00h
0c53 00             ??          00h
0c54 00             ??          00h
0c55 00             ??          00h
0c56 00             ??          00h
0c57 00             ??          00h
0c58 00             ??          00h
0c59 00             ??          00h
0c5a 00             ??          00h
0c5b 00             ??          00h
0c5c 00             ??          00h
0c5d 00             ??          00h
0c5e 00             ??          00h
0c5f 00             ??          00h
0c60 00             ??          00h
0c61 00             ??          00h
0c62 00             ??          00h
0c63 00             ??          00h
0c64 00             ??          00h
0c65 00             ??          00h
0c66 00             ??          00h
0c67 00             ??          00h
0c68 00             ??          00h
0c69 00             ??          00h
0c6a 00             ??          00h
0c6b 00             ??          00h
0c6c 00             ??          00h
0c6d 00             ??          00h
0c6e 00             ??          00h
0c6f 00             ??          00h
0c70 00             ??          00h
0c71 00             ??          00h
0c72 00             ??          00h
0c73 00             ??          00h
0c74 00             ??          00h
0c75 00             ??          00h
0c76 00             ??          00h
0c77 00             ??          00h
0c78 00             ??          00h
0c79 00             ??          00h
0c7a 00             ??          00h
0c7b 00             ??          00h
0c7c 00             ??          00h
0c7d 00             ??          00h
0c7e 00             ??          00h
0c7f 00             ??          00h
0c80 00             ??          00h
0c81 00             ??          00h
0c82 00             ??          00h
0c83 00             ??          00h
0c84 00             ??          00h
0c85 00             ??          00h
0c86 00             ??          00h
0c87 00             ??          00h
0c88 00             ??          00h
0c89 00             ??          00h
0c8a 00             ??          00h
0c8b 00             ??          00h
0c8c 00             ??          00h
0c8d 00             ??          00h
0c8e 00             ??          00h
0c8f 00             ??          00h
0c90 00             ??          00h
0c91 00             ??          00h
0c92 00             ??          00h
0c93 00             ??          00h
0c94 00             ??          00h
0c95 00             ??          00h
0c96 00             ??          00h
0c97 00             ??          00h
0c98 00             ??          00h
```

```
0c99 00              ??         00h
0c9a 00              ??         00h
0c9b 00              ??         00h
0c9c 00              ??         00h
0c9d 00              ??         00h
0c9e 00              ??         00h
0c9f 00              ??         00h
0ca0 00              ??         00h
0ca1 00              ??         00h
0ca2 00              ??         00h
0ca3 00              ??         00h
0ca4 00              ??         00h
0ca5 00              ??         00h
0ca6 00              ??         00h
0ca7 00              ??         00h
0ca8 00              ??         00h
0ca9 00              ??         00h
0caa 00              ??         00h
0cab 00              ??         00h
0cac 00              ??         00h
0cad 00              ??         00h
0cae 00              ??         00h
0caf 00              ??         00h
0cb0 00              ??         00h
0cb1 00              ??         00h
0cb2 00              ??         00h
0cb3 00              ??         00h
0cb4 00              ??         00h
0cb5 00              ??         00h
0cb6 00              ??         00h
0cb7 00              ??         00h
0cb8 00              ??         00h
0cb9 00              ??         00h
0cba 00              ??         00h
0cbb 00              ??         00h
0cbc 00              ??         00h
0cbd 00              ??         00h
0cbe 00              ??         00h
0cbf 00              ??         00h
0cc0 00              ??         00h
0cc1 00              ??         00h
0cc2 00              ??         00h
0cc3 00              ??         00h
0cc4 00              ??         00h
0cc5 00              ??         00h
0cc6 00              ??         00h
0cc7 00              ??         00h
0cc8 00              ??         00h
0cc9 00              ??         00h
0cca 00              ??         00h
0ccb 00              ??         00h
0ccc 00              ??         00h
0ccd 00              ??         00h
0cce 00              ??         00h
0ccf 00              ??         00h
0cd0 00              ??         00h
0cd1 00              ??         00h
0cd2 00              ??         00h
0cd3 00              ??         00h
0cd4 00              ??         00h
0cd5 00              ??         00h
0cd6 00              ??         00h
0cd7 00              ??         00h
0cd8 00              ??         00h
0cd9 00              ??         00h
0cda 00              ??         00h
0cdb 00              ??         00h
0cdc 00              ??         00h
0cdd 00              ??         00h
0cde 00              ??         00h
0cdf 00              ??         00h
0ce0 00              ??         00h
0ce1 00              ??         00h
0ce2 00              ??         00h
0ce3 00              ??         00h
0ce4 00              ??         00h
0ce5 00              ??         00h
0ce6 00              ??         00h
0ce7 00              ??         00h
0ce8 00              ??         00h
0ce9 00              ??         00h
0cea 00              ??         00h
0ceb 00              ??         00h
0cec 00              ??         00h
0ced 00              ??         00h
```

```
0cee 00          ??         00h
0cef 00          ??         00h
0cf0 00          ??         00h
0cf1 00          ??         00h
0cf2 00          ??         00h
0cf3 00          ??         00h
0cf4 00          ??         00h
0cf5 00          ??         00h
0cf6 00          ??         00h
0cf7 00          ??         00h

             DAT_0cf8                                 XREF[27]:    Format_7Segment_Display_Patterns…
                                                                   Format_7Segment_Display_Patterns…
                                                                   Format_7Segment_Display_Patterns…
                                                                   Format_7Segment_Display_Patterns…
                                                                   Format_7Segment_Display_Patterns…
                                                                   Format_7Segment_Display_Patterns…
                                                                   Format_7Segment_Display_Patterns…
                                                                   Format_7Segment_Display_Patterns…
                                                                   Format_7Segment_Display_Patterns…
                                                                   Format_7Segment_Display_Patterns…
                                                                   Process_Display_Data_Alternative…
                                                                   Process_Display_Data_Alternative…
                                                                   Process_Display_Data_Alternative…
                                                                   Process_Display_Data_Alternative…
                                                                   Process_Display_Data_Alternative…
                                                                   Process_Display_Data_Alternative…
                                                                   Process_Display_Data_Alternative…
                                                                   Process_Display_Data_Alternative…
                                                                   Process_Display_Data_Alternative…
                                                                   Process_Display_Data_Alternative…
                                                                   [more]
0cf8 00          ??         00h
0cf9 20          ??         20h
0cfa 28          ??         28h    (
0cfb 2a          ??         2Ah    *
0cfc 50          ??         50h    P
0cfd 51          ??         51h    Q
0cfe 54          ??         54h    T
0cff 55          ??         55h    U
0d00 5f          ??         5Fh    _
0d01 66          ??         66h    f
0d02 6e          ??         6Eh    n
0d03 4e          ??         4Eh    N
0d04 5d          ??         5Dh    ]
0d05 08          ??         08h
0d06 44          ??         44h    D
0d07 4f          ??         4Fh    O
0d08 77          ??         77h    w
0d09 11          ??         11h
0d0a 6b          ??         6Bh    k
0d0b 3b          ??         3Bh    ;
0d0c 1d          ??         1Dh
0d0d 3e          ??         3Eh    >
0d0e 7e          ??         7Eh    ~
0d0f 13          ??         13h
0d10 7f          ??         7Fh
0d11 3f          ??         3Fh    ?

             DAT_0d12                                 XREF[2]:     Process_PANC_Command_From_FIFO:0…
                                                                   Process_PANC_Command_From_FIFO:0…
0d12 3a          ??         3Ah    :
0d13 3a          ??         3Ah    :
0d14 52          ??         52h    R
0d15 3a          ??         3Ah    :
0d16 3a          ??         3Ah    :
0d17 57          ??         57h    W
0d18 52          ??         52h    R
0d19 57          ??         57h    W

             DAT_0d1a                                 XREF[2]:     Process_PANC_Command_From_FIFO:0…
                                                                   Process_PANC_Command_From_FIFO:0…
0d1a 43          ??         43h    C
0d1b 44          ??         44h    D
0d1c 44          ??         44h    D
0d1d 44          ??         44h    D
0d1e 43          ??         43h    C
0d1f 41          ??         41h    A
0d20 44          ??         44h    D
0d21 41          ??         41h    A
```

```
                       DAT_0d22                                             XREF[8]:     Load_Display_Character_Data_From..
                                                                                         Load_Display_Character_Data_From..
                                                                                         Load_Display_Character_Data_From..
                                                                                         Load_Display_Character_Data_From..
                                                                                         Load_Display_Character_Data_From..
                                                                                         Load_Display_Character_Data_From..
                                                                                         Load_Display_Character_Data_From..
                                                                                         Load_Display_Character_Data_From..
        0d22 00              ??         00h
        0d23 00              ??         00h
        0d24 55              ??         55h    U
        0d25 11              ??         11h
        0d26 55              ??         55h    U
        0d27 11              ??         11h
        0d28 55              ??         55h    U
        0d29 11              ??         11h
        0d2a 55              ??         55h    U
        0d2b 11              ??         11h
        0d2c 55              ??         55h    U
        0d2d 11              ??         11h
        0d2e 55              ??         55h    U
        0d2f 11              ??         11h
        0d30 55              ??         55h    U
        0d31 11              ??         11h
        0d32 4d              ??         4Dh    M
        0d33 9a              ??         9Ah
        0d34 41              ??         41h    A
        0d35 82              ??         82h
        0d36 45              ??         45h    E
        0d37 99              ??         99h
        0d38 45              ??         45h    E
        0d39 8b              ??         8Bh
        0d3a 49              ??         49h    I
        0d3b 83              ??         83h
        0d3c 4c              ??         4Ch    L
        0d3d 83              ??         83h
        0d3e 4c              ??         4Ch    L
        0d3f 9b              ??         9Bh
        0d40 45              ??         45h    E
        0d41 82              ??         82h
        0d42 88              ??         88h
        0d43 11              ??         11h
        0d44 8d              ??         8Dh
        0d45 11              ??         11h
        0d46 88              ??         88h
        0d47 1b              ??         1Bh
        0d48 8d              ??         8Dh
        0d49 1b              ??         1Bh
        0d4a 88              ??         88h
        0d4b b1              ??         B1h
        0d4c 8d              ??         8Dh
        0d4d b1              ??         B1h
        0d4e 88              ??         88h
        0d4f bb              ??         BBh
        0d50 8d              ??         8Dh
        0d51 bb              ??         BBh
        0d52 d8              ??         D8h
        0d53 11              ??         11h
        0d54 dd              ??         DDh
        0d55 11              ??         11h
        0d56 d8              ??         D8h
        0d57 1b              ??         1Bh
        0d58 dd              ??         DDh
        0d59 1b              ??         1Bh
        0d5a d8              ??         D8h
        0d5b b1              ??         B1h
        0d5c dd              ??         DDh
        0d5d b1              ??         B1h
        0d5e d8              ??         D8h
        0d5f bb              ??         BBh
        0d60 dd              ??         DDh
        0d61 bb              ??         BBh
        0d62 00              ??         00h
        0d63 00              ??         00h
        0d64 08              ??         08h
        0d65 00              ??         00h
        0d66 0e              ??         0Eh
        0d67 00              ??         00h
        0d68 0f              ??         0Fh
        0d69 01              ??         01h
        0d6a 0f              ??         0Fh
        0d6b 07              ??         07h
        0d6c 0f              ??         0Fh
        0d6d 1f              ??         1Fh
```

```
0d6e 0f            ??         0Fh
0d6f 7f            ??         7Fh
0d70 8f            ??         8Fh
0d71 ff            ??         FFh
0d72 ef            ??         EFh
0d73 ff            ??         FFh
0d74 ff            ??         FFh
0d75 ff            ??         FFh
0d76 aa            ??         AAh
0d77 55            ??         55h    U
0d78 88            ??         88h
0d79 11            ??         11h
0d7a 00            ??         00h
0d7b 40            ??         40h    @
0d7c 80            ??         80h
0d7d 01            ??         01h
0d7e 00            ??         00h
0d7f 20            ??         20h
0d80 02            ??         02h
0d81 40            ??         40h    @
0d82 0d            ??         0Dh
0d83 1a            ??         1Ah
0d84 03            ??         03h
0d85 02            ??         02h
0d86 05            ??         05h
0d87 19            ??         19h
0d88 05            ??         05h
0d89 0b            ??         0Bh
0d8a 09            ??         09h
0d8b 03            ??         03h
0d8c 0c            ??         0Ch
0d8d 0b            ??         0Bh
0d8e 0c            ??         0Ch
0d8f 1b            ??         1Bh
0d90 05            ??         05h
0d91 02            ??         02h
0d92 0d            ??         0Dh
0d93 1b            ??         1Bh
0d94 0d            ??         0Dh
0d95 0b            ??         0Bh
0d96 00            ??         00h
0d97 00            ??         00h
0d98 0d            ??         0Dh
0d99 11            ??         11h
0d9a 40            ??         40h    @
0d9b 80            ??         80h
0d9c 4d            ??         4Dh    M
0d9d 91            ??         91h
0d9e 00            ??         00h
0d9f 00            ??         00h
0da0 55            ??         55h    U
0da1 11            ??         11h
0da2 95            ??         95h
0da3 ba            ??         BAh
0da4 d5            ??         D5h
0da5 83            ??         83h
0da6 1d            ??         1Dh
0da7 3b            ??         3Bh    ;
0da8 54            ??         54h    T
0da9 a8            ??         A8h
0daa 1d            ??         1Dh
0dab 3a            ??         3Ah    :
0dac d4            ??         D4h
0dad a8            ??         A8h
0dae d4            ??         D4h
0daf 80            ??         80h
0db0 54            ??         54h    T
0db1 ab            ??         ABh
0db2 c1            ??         C1h
0db3 83            ??         83h
0db4 1c            ??         1Ch
0db5 38            ??         38h    8
0db6 1c            ??         1Ch
0db7 b0            ??         B0h
0db8 c2            ??         C2h
0db9 84            ??         84h
0dba 40            ??         40h    @
0dbb a8            ??         A8h
0dbc 63            ??         63h    c
0dbd 82            ??         82h
0dbe 61            ??         61h    a
0dbf 86            ??         86h
0dc0 55            ??         55h    U
0dc1 aa            ??         AAh
0dc2 d5            ??         D5h
```

```
0dc3 81              ??         81h
0dc4 55              ??         55h    U
0dc5 ae              ??         AEh
0dc6 d5              ??         D5h
0dc7 85              ??         85h
0dc8 d4              ??         D4h
0dc9 2b              ??         2Bh    +
0dca 1c              ??         1Ch
0dcb 10              ??         10h
0dcc 41              ??         41h    A
0dcd aa              ??         AAh
0dce 42              ??         42h    B
0dcf c0              ??         C0h
0dd0 41              ??         41h    A
0dd1 c6              ??         C6h
0dd2 22              ??         22h    "
0dd3 44              ??         44h    D
0dd4 22              ??         22h    "
0dd5 10              ??         10h
0dd6 16              ??         16h
0dd7 68              ??         68h    h
0dd8 dc              ??         DCh
0dd9 99              ??         99h
0dda 57              ??         57h    W
0ddb ea              ??         EAh
0ddc 13              ??         13h
0ddd 43              ??         43h    C

                DAT_0dde                                         XREF[1]:    Convert_Time_Offset_To_Software_…
0dde 00              ??         00h
0ddf 00              ??         00h
0de0 38              ??         38h    8
0de1 3e              ??         3Eh    >
0de2 3c              ??         3Ch    <
0de3 3e              ??         3Eh    >
0de4 3c              ??         3Ch    <
0de5 3e              ??         3Eh    >
0de6 3e              ??         3Eh    >
0de7 3c              ??         3Ch    <
0de8 3e              ??         3Eh    >
0de9 3c              ??         3Ch    <
0dea 3e              ??         3Eh    >

                DAT_0deb                                         XREF[2]:    Calculate_RTC_Date_Time_Offset:0…
                                                                             Calculate_RTC_Date_Time_Offset:0…
0deb 00              ??         00h
0dec 00              ??         00h
0ded 00              ??         00h
0dee 00              ??         00h
0def 00              ??         00h
0df0 3e              ??         3Eh    >
0df1 00              ??         00h
0df2 76              ??         76h    v
0df3 00              ??         00h
0df4 b4              ??         B4h
0df5 00              ??         00h
0df6 f0              ??         F0h
0df7 01              ??         01h
0df8 2e              ??         2Eh    .
0df9 01              ??         01h
0dfa 6a              ??         6Ah    j
0dfb 01              ??         01h
0dfc a8              ??         A8h
0dfd 01              ??         01h
0dfe e6              ??         E6h
0dff 02              ??         02h
0e00 22              ??         22h    "
0e01 02              ??         02h
0e02 60              ??         60h    `
0e03 02              ??         02h
0e04 9c              ??         9Ch
0e05 00              ??         00h
0e06 00              ??         00h
0e07 00              ??         00h
0e08 00              ??         00h
0e09 00              ??         00h
0e0a 3e              ??         3Eh    >
0e0b 00              ??         00h
0e0c 78              ??         78h    x
0e0d 00              ??         00h
0e0e b6              ??         B6h
0e0f 00              ??         00h
0e10 f2              ??         F2h
0e11 01              ??         01h
0e12 30              ??         30h    0
```

```
        0e13 01          ??         01h
        0e14 6c          ??         6Ch    l
        0e15 01          ??         01h
        0e16 aa          ??         AAh
        0e17 01          ??         01h
        0e18 e8          ??         E8h
        0e19 02          ??         02h
        0e1a 24          ??         24h    $
        0e1b 02          ??         02h
        0e1c 62          ??         62h    b
        0e1d 02          ??         02h
        0e1e 9e          ??         9Eh

                     DAT_0e1f                                XREF[2]:     Calculate_RTC_Date_Time_Offset:0…
                                                                          Calculate_RTC_Date_Time_Offset:0…
        0e1f 00          ??         00h
        0e20 00          ??         00h
        0e21 0e          ??         0Eh
        0e22 10          ??         10h
        0e23 1c          ??         1Ch
        0e24 20          ??         20h
        0e25 2a          ??         2Ah    *
        0e26 30          ??         30h    0
        0e27 38          ??         38h    8
        0e28 40          ??         40h    @
        0e29 46          ??         46h    F
        0e2a 50          ??         50h    P
        0e2b 54          ??         54h    T
        0e2c 60          ??         60h    `
        0e2d 62          ??         62h    b
        0e2e 70          ??         70h    p
        0e2f 70          ??         70h    p
        0e30 80          ??         80h
        0e31 7e          ??         7Eh    ~
        0e32 90          ??         90h
        0e33 8c          ??         8Ch
        0e34 a0          ??         A0h
        0e35 9a          ??         9Ah
        0e36 b0          ??         B0h

                     DAT_0e37                                XREF[1]:     Increment_Software_RTC_Counters:…
        0e37 00          ??         00h
        0e38 1f          ??         1Fh
        0e39 1c          ??         1Ch
        0e3a 1f          ??         1Fh
        0e3b 1e          ??         1Eh
        0e3c 1f          ??         1Fh
        0e3d 1e          ??         1Eh
        0e3e 1f          ??         1Fh
        0e3f 1f          ??         1Fh
        0e40 1e          ??         1Eh
        0e41 1f          ??         1Fh
        0e42 1e          ??         1Eh
        0e43 1f          ??         1Fh

                     DAT_0e44                                XREF[18]:    Format_7Segment_Display_Pattern…
                                                                          Format_7Segment_Display_Pattern…
                                                                          Format_7Segment_Display_Pattern…
                                                                          Format_7Segment_Display_Pattern…
                                                                          Format_7Segment_Display_Pattern…
                                                                          Format_7Segment_Display_Pattern…
                                                                          Output_Status_Code_1_To_ND120:0…
                                                                          Output_Status_Code_1_To_ND120:0…
                                                                          Output_Status_Code_1_To_ND120:0…
                                                                          Output_Status_Code_1_To_ND120:0…
                                                                          Output_Status_Code_1_To_ND120:0…
                                                                          Output_Status_Code_1_To_ND120:0…
                                                                          Output_Status_Code_1_To_ND120:0…
                                                                          Output_Status_Code_1_To_ND120:0…
                                                                          Output_Status_Code_1_To_ND120:0…
                                                                          Output_Status_Code_1_To_ND120:0…
                                                                          Output_Status_Code_1_To_ND120:0…
                                                                          Output_Status_Code_1_To_ND120:0…
        0e44 00          ??         00h
        0e45 00          ??         00h
        0e46 00          ??         00h
        0e47 01          ??         01h
        0e48 00          ??         00h
        0e49 02          ??         02h
        0e4a 00          ??         00h
        0e4b 03          ??         03h
        0e4c 00          ??         00h
        0e4d 04          ??         04h
        0e4e 00          ??         00h
        0e4f 05          ??         05h
```

```
0e50 00             ??          00h
0e51 06             ??          06h
0e52 00             ??          00h
0e53 07             ??          07h
0e54 00             ??          00h
0e55 08             ??          08h
0e56 00             ??          00h
0e57 09             ??          09h
0e58 01             ??          01h
0e59 00             ??          00h
0e5a 01             ??          01h
0e5b 01             ??          01h
0e5c 01             ??          01h
0e5d 02             ??          02h
0e5e 01             ??          01h
0e5f 03             ??          03h
0e60 01             ??          01h
0e61 04             ??          04h
0e62 01             ??          01h
0e63 05             ??          05h
0e64 01             ??          01h
0e65 06             ??          06h
0e66 01             ??          01h
0e67 07             ??          07h
0e68 01             ??          01h
0e69 08             ??          08h
0e6a 01             ??          01h
0e6b 09             ??          09h
0e6c 02             ??          02h
0e6d 00             ??          00h
0e6e 02             ??          02h
0e6f 01             ??          01h
0e70 02             ??          02h
0e71 02             ??          02h
0e72 02             ??          02h
0e73 03             ??          03h
0e74 02             ??          02h
0e75 04             ??          04h
0e76 02             ??          02h
0e77 05             ??          05h
0e78 02             ??          02h
0e79 06             ??          06h
0e7a 02             ??          02h
0e7b 07             ??          07h
0e7c 02             ??          02h
0e7d 08             ??          08h
0e7e 02             ??          02h
0e7f 09             ??          09h
0e80 03             ??          03h
0e81 00             ??          00h
0e82 03             ??          03h
0e83 01             ??          01h
0e84 03             ??          03h
0e85 02             ??          02h
0e86 03             ??          03h
0e87 03             ??          03h
0e88 03             ??          03h
0e89 04             ??          04h
0e8a 03             ??          03h
0e8b 05             ??          05h
0e8c 03             ??          03h
0e8d 06             ??          06h
0e8e 03             ??          03h
0e8f 07             ??          07h
0e90 03             ??          03h
0e91 08             ??          08h
0e92 03             ??          03h
0e93 09             ??          09h
0e94 04             ??          04h
0e95 00             ??          00h
0e96 04             ??          04h
0e97 01             ??          01h
0e98 04             ??          04h
0e99 02             ??          02h
0e9a 04             ??          04h
0e9b 03             ??          03h
0e9c 04             ??          04h
0e9d 04             ??          04h
0e9e 04             ??          04h
0e9f 05             ??          05h
0ea0 04             ??          04h
0ea1 06             ??          06h
0ea2 04             ??          04h
0ea3 07             ??          07h
0ea4 04             ??          04h
```

```
0ea5 08              ??          08h
0ea6 04              ??          04h
0ea7 09              ??          09h
0ea8 05              ??          05h
0ea9 00              ??          00h
0eaa 05              ??          05h
0eab 01              ??          01h
0eac 05              ??          05h
0ead 02              ??          02h
0eae 05              ??          05h
0eaf 03              ??          03h
0eb0 05              ??          05h
0eb1 04              ??          04h
0eb2 05              ??          05h
0eb3 05              ??          05h
0eb4 05              ??          05h
0eb5 06              ??          06h
0eb6 05              ??          05h
0eb7 07              ??          07h
0eb8 05              ??          05h
0eb9 08              ??          08h
0eba 05              ??          05h
0ebb 09              ??          09h
0ebc 06              ??          06h
0ebd 00              ??          00h
0ebe 06              ??          06h
0ebf 01              ??          01h
0ec0 06              ??          06h
0ec1 02              ??          02h
0ec2 06              ??          06h
0ec3 03              ??          03h
0ec4 06              ??          06h
0ec5 04              ??          04h
0ec6 06              ??          06h
0ec7 05              ??          05h
0ec8 06              ??          06h
0ec9 06              ??          06h
0eca 06              ??          06h
0ecb 07              ??          07h
0ecc 06              ??          06h
0ecd 08              ??          08h
0ece 06              ??          06h
0ecf 09              ??          09h
0ed0 07              ??          07h
0ed1 00              ??          00h
0ed2 07              ??          07h
0ed3 01              ??          01h
0ed4 07              ??          07h
0ed5 02              ??          02h
0ed6 07              ??          07h
0ed7 03              ??          03h
0ed8 07              ??          07h
0ed9 04              ??          04h
0eda 07              ??          07h
0edb 05              ??          05h
0edc 07              ??          07h
0edd 06              ??          06h
0ede 07              ??          07h
0edf 07              ??          07h
0ee0 07              ??          07h
0ee1 08              ??          08h
0ee2 07              ??          07h
0ee3 09              ??          09h
0ee4 08              ??          08h
0ee5 00              ??          00h
0ee6 08              ??          08h
0ee7 01              ??          01h
0ee8 08              ??          08h
0ee9 02              ??          02h
0eea 08              ??          08h
0eeb 03              ??          03h
0eec 08              ??          08h
0eed 04              ??          04h
0eee 08              ??          08h
0eef 05              ??          05h
0ef0 08              ??          08h
0ef1 06              ??          06h
0ef2 08              ??          08h
0ef3 07              ??          07h
0ef4 08              ??          08h
0ef5 08              ??          08h
0ef6 08              ??          08h
0ef7 09              ??          09h
0ef8 09              ??          09h
0ef9 00              ??          00h
```

```
0efa 09          ??         09h
0efb 01          ??         01h
0efc 09          ??         09h
0efd 02          ??         02h
0efe 09          ??         09h
0eff 03          ??         03h
0f00 09          ??         09h
0f01 04          ??         04h
0f02 09          ??         09h
0f03 05          ??         05h
0f04 09          ??         09h
0f05 06          ??         06h
0f06 09          ??         09h
0f07 07          ??         07h
0f08 09          ??         09h
0f09 08          ??         08h
0f0a 09          ??         09h
0f0b 09          ??         09h

                PCR_Ring_Level_Bit_Lookup_Table                XREF[1]:     Sample_ND120_CPU_Status_Signals…
0f0c 01          db         1h
0f0d 02          db         2h
0f0e 04          db         4h
0f0f 08          db         8h
0f10 00          ??         00h
0f11 00          ??         00h
0f12 00          ??         00h
0f13 00          ??         00h
0f14 00          ??         00h
0f15 00          ??         00h
0f16 00          ??         00h
0f17 00          ??         00h
0f18 00          ??         00h
0f19 00          ??         00h
0f1a 00          ??         00h
0f1b 00          ??         00h
0f1c 00          ??         00h
0f1d 00          ??         00h
0f1e 00          ??         00h
0f1f 00          ??         00h
0f20 00          ??         00h
0f21 00          ??         00h
0f22 00          ??         00h
0f23 00          ??         00h
0f24 00          ??         00h
0f25 00          ??         00h
0f26 00          ??         00h
0f27 00          ??         00h
0f28 00          ??         00h
0f29 00          ??         00h
0f2a 00          ??         00h
0f2b 00          ??         00h
0f2c 00          ??         00h
0f2d 00          ??         00h
0f2e 00          ??         00h
0f2f 00          ??         00h
0f30 00          ??         00h
0f31 00          ??         00h
0f32 00          ??         00h
0f33 00          ??         00h
0f34 00          ??         00h
0f35 00          ??         00h
0f36 00          ??         00h
0f37 00          ??         00h

                Mask bits 0x25 = 0010_0101
                CLK = 0 (Clock Oscilation type) 0=Crystal (1=resistor capacit…
                TOPT = 0 (Timer Mask/Programmable option) (0=All TCR bits are…
                CLS =1 (Timer Clock Source) (1=External TIMER PIN. 0=Internal…
                TIE = 0 (Timer External Input Enable)
                SNM = 0 (Secure/Non-Secure Mode Option)
                P2 = 1
                P1 = 0
                P0 = 1

                P2-P0 = (Prescaler option) => 101 => Divide by 32.

                MOR_Mask_Option_Register
0f38 25          db         25h                                 Mask Option Register
                *** START: Unused hole from 0xF39 to 0xF7F ***
0f39 00          db         0h
0f3a 00          db         0h
0f3b 00          db         0h
0f3c 00          db         0h
0f3d 00          db         0h
```

```
0f3e 00              db          0h
0f3f 00              db          0h
0f40 00              db          0h
0f41 00              db          0h
0f42 00              db          0h
0f43 00              db          0h
0f44 00              db          0h
0f45 00              db          0h
0f46 00              db          0h
0f47 00              db          0h
0f48 00              db          0h
0f49 00              db          0h
0f4a 00              db          0h
0f4b 00              db          0h
0f4c 00              db          0h
0f4d 00              db          0h
0f4e 00              db          0h
0f4f 00              db          0h
0f50 00              db          0h
0f51 00              db          0h
0f52 00              db          0h
0f53 00              db          0h
0f54 00              db          0h
0f55 00              db          0h
0f56 00              db          0h
0f57 00              db          0h
0f58 00              db          0h
0f59 00              db          0h
0f5a 00              db          0h
0f5b 00              db          0h
0f5c 00              db          0h
0f5d 00              db          0h
0f5e 00              db          0h
0f5f 00              db          0h
0f60 00              db          0h
0f61 00              db          0h
0f62 00              db          0h
0f63 00              db          0h
0f64 00              db          0h
0f65 00              db          0h
0f66 00              db          0h
0f67 00              db          0h
0f68 00              db          0h
0f69 00              db          0h
0f6a 00              db          0h
0f6b 00              db          0h
0f6c 00              db          0h
0f6d 00              db          0h
0f6e 00              db          0h
0f6f 00              db          0h
0f70 00              db          0h
0f71 00              db          0h
0f72 00              db          0h
0f73 00              db          0h
0f74 00              db          0h
0f75 00              db          0h
0f76 00              db          0h
0f77 00              db          0h
0f78 00              db          0h
0f79 00              db          0h
0f7a 00              db          0h
0f7b 00              db          0h
0f7c 00              db          0h
0f7d 00              db          0h
0f7e 00              db          0h
0f7f 00              db          0h
                *** END: Unused hole from 0xF39 to 0xF7F ***
0f80 00              ??          00h
0f81 00              ??          00h
0f82 00              ??          00h
0f83 00              ??          00h
0f84 00              ??          00h
0f85 00              ??          00h
0f86 00              ??          00h
0f87 00              ??          00h
0f88 00              ??          00h
0f89 00              ??          00h
0f8a 00              ??          00h
0f8b 00              ??          00h
0f8c 00              ??          00h
0f8d 00              ??          00h
0f8e 00              ??          00h
0f8f 00              ??          00h
0f90 00              ??          00h
0f91 00              ??          00h
```

```
0f92 00             ??          00h
0f93 00             ??          00h
0f94 00             ??          00h
0f95 00             ??          00h
0f96 00             ??          00h
0f97 00             ??          00h
0f98 00             ??          00h
0f99 00             ??          00h
0f9a 00             ??          00h
0f9b 00             ??          00h
0f9c 00             ??          00h
0f9d 00             ??          00h
0f9e 00             ??          00h
0f9f 00             ??          00h
0fa0 00             ??          00h
0fa1 00             ??          00h
0fa2 00             ??          00h
0fa3 00             ??          00h
0fa4 00             ??          00h
0fa5 00             ??          00h
0fa6 00             ??          00h
0fa7 00             ??          00h
0fa8 00             ??          00h
0fa9 00             ??          00h
0faa 00             ??          00h
0fab 00             ??          00h
0fac 00             ??          00h
0fad 00             ??          00h
0fae 00             ??          00h
0faf 00             ??          00h
0fb0 00             ??          00h
0fb1 00             ??          00h
0fb2 00             ??          00h
0fb3 00             ??          00h
0fb4 00             ??          00h
0fb5 00             ??          00h
0fb6 00             ??          00h
0fb7 00             ??          00h
0fb8 00             ??          00h
0fb9 00             ??          00h
0fba 00             ??          00h
0fbb 00             ??          00h
0fbc 00             ??          00h
0fbd 00             ??          00h
0fbe 00             ??          00h
0fbf 00             ??          00h
0fc0 00             ??          00h
0fc1 00             ??          00h
0fc2 00             ??          00h
0fc3 00             ??          00h
0fc4 00             ??          00h
0fc5 00             ??          00h
0fc6 00             ??          00h
0fc7 00             ??          00h
0fc8 00             ??          00h
0fc9 00             ??          00h
0fca 00             ??          00h
0fcb 00             ??          00h
0fcc 00             ??          00h
0fcd 00             ??          00h
0fce 00             ??          00h
0fcf 00             ??          00h
0fd0 00             ??          00h
0fd1 00             ??          00h
0fd2 00             ??          00h
0fd3 00             ??          00h
0fd4 00             ??          00h
0fd5 00             ??          00h
0fd6 00             ??          00h
0fd7 00             ??          00h
0fd8 00             ??          00h
0fd9 00             ??          00h
0fda 00             ??          00h
0fdb 00             ??          00h
0fdc 00             ??          00h
0fdd 00             ??          00h
0fde 00             ??          00h
0fdf 00             ??          00h
0fe0 00             ??          00h
0fe1 00             ??          00h
0fe2 00             ??          00h
0fe3 00             ??          00h
0fe4 00             ??          00h
0fe5 00             ??          00h
0fe6 00             ??          00h
```

```
0fe7 00          ??        00h
0fe8 00          ??        00h
0fe9 00          ??        00h
0fea 00          ??        00h
0feb 00          ??        00h
0fec 00          ??        00h
0fed 00          ??        00h
0fee 00          ??        00h
0fef 00          ??        00h
0ff0 00          ??        00h
0ff1 00          ??        00h
0ff2 00          ??        00h
0ff3 00          ??        00h
0ff4 00          ??        00h
0ff5 00          ??        00h
0ff6 00 00       addr      0000                                 BOOTSTRAP Vector (Not used in th…
0ff8 08 c8       addr      Timer_1200Hz_CPU_Performance_Monitor   Timer Interrupt Vector
0ffa 00 00       addr      0000                                 External Interrupt Vector
0ffc 09 ba       addr      SWI_INTERRUPT                        SWI Vector
0ffe 01 10       addr      Wait_For_FIFO_Commands_And_Process   RESET Vector
```