```
                    //
                    // ROM
                    // RAM:0000-RAM:07ff
                    //

                    PORTA (0x0000) - INPUT PORT - MC68705P3 Pins PA0-PA7
                    Data Direction: Input (DDRA = 0x00) - ALL INPUTS
                    Primary Function: Command Reception and Button Input

                    PIN ASSIGNMENTS (28-pin DIP package):
                    Pin 23 - PA0: Serial Data Channel 1 (CPU command data bit 0, ...
                    Pin 22 - PA1: Serial Data Channel 2 (CPU command data bit 1, ...
                    Pin 21 - PA2: Serial Data Channel 3 (CPU command data bit 2, ...
                    Pin 20 - PA3: Panel Lock Key Sense (0=LOCK, 1=ON/STANDBY posi...
                    Pin 19 - PA4: Command/Button Data (additional command or butt...
                    Pin 18 - PA5: Command/Button Data (additional command or butt...
                    Pin 17 - PA6: Button Change Flag (button state change detecti...
                    Pin 16 - PA7: Status/Control (additional control or status in...

                    HARDWARE INTERFACE:
                    - ND-120 CPU → PANC Register → CY7C401 FIFO → External Logi...
                    - Button Matrix → External Encoding Logic → PA4-PA7
                    - Panel Lock Key → Direct Connection → PA3

                    FIRMWARE USAGE:
                    - Command extraction: cmd = PORTA & 0x3F (extract 6-bit comma...
                    - Serial reception: if (!(PORTA & 1)) ShiftRegister1 |= 0x80 ...
                    - Lock key sensing: if (PORTA & 8) panel_unlocked else panel_...
                    - Button processing: ButtonChangeFlags = PORTA ^ PreviousButt...
                    - Stability checking: if (PORTA == ButtonStateBuffer) for deb...
                    PORTA                                   XREF[7]:     RESET:00ea(R),
                                                                         ProcessData:014d(R),
                                                                         ProcessData:0157(R),
                                                                         cmd_handler_10:01b7(R),
                                                                         cmd_handler_6c:0213(R),
                                                                         WaitForData:0268(R),
                                                                         WaitForData:0297(R)
0000 00             db          0h                                       Port A is READ (Input) ONLY

                    PORTB (0x0001) - OUTPUT PORT - MC68705P3 Pins PB0-PB7
                    Data Direction: Output (DDRB = 0xFF) - ALL OUTPUTS
                    Primary Function: Display Data and CPU Response

                    PIN ASSIGNMENTS (28-pin DIP package):
                    Pin  6 - PB0: Response Data Bit 0 (CPU PANS register bit 0)
                    Pin  7 - PB1: Response Data Bit 1 (CPU PANS register bit 1)
                    Pin  8 - PB2: Response Data Bit 2 (CPU PANS register bit 2)
                    Pin  9 - PB3: Response Data Bit 3 (CPU PANS register bit 3)
                    Pin 10 - PB4: Response Data Bit 4 (CPU PANS register bit 4)
                    Pin 11 - PB5: Response Data Bit 5 (CPU PANS register bit 5)
                    Pin 12 - PB6: Response Data Bit 6 (CPU PANS register bit 6)
                    Pin 13 - PB7: Response Data Bit 7 (CPU PANS register bit 7)

                    HARDWARE INTERFACE:
                    - PB0-PB7 → HD44100H LCD Driver Data Input (8-bit parallel)
                    - PB0-PB7 → CD4035 Shift Register Serial Data Input
                    - PB0-PB7 → External Logic → CPU PANS Register (response data)

                    FIRMWARE USAGE:
                    - Combined response: PORTB = (data | DisplayControlFlags) & 0...
                    - Direct data: PORTB = data (8-bit direct output)
                    - Display commands: PORTB = display_command (to HD44100H)
                    - Character output: PORTB = character_code (display data)
                    - Status output: PORTB = DisplayControlFlags (status to CPU)

                    PROTOCOL: Always used with PORTC bit 0 strobe for data valid ...
                    PORTB                                   XREF[4]:     WriteToDisplayPort:023e(W),
                                                                         WaitForData:0260(W),
                                                                         OutputCharacterToDisplay:05ed(W),
                                                                         SendDisplayCommand:05fc(W)
0001 ff             db          FFh
```

```
                    PORTC (0x0002) - CONTROL OUTPUT PORT - MC68705P3 Pins PC0-PC7
                    Data Direction: Output (DDRC = 0xFF) - ALL OUTPUTS
                    Primary Function: Control Signals and Timing

                    PIN ASSIGNMENTS (28-pin DIP package):
                    Pin  2 - PC0: Display Data Strobe (HD44100H/CD4035 data valid...
                    Pin 27 - PC1: Serial Clock (serial data sampling clock for PA...
                    Pin  3 - PC2: Display Command Mode (HD44100H command/data sel...
                    Pin 26 - PC3: Display Control 1 (additional display control s...
                    Pin  4 - PC4: Display Control 2 (additional display control s...
                    Pin 25 - PC5: Display Control 3 (additional display control s...
                    Pin  5 - PC6: Display Control 4 (additional display control s...
                    Pin 24 - PC7: Display Control 5 (additional display control s...

                    HARDWARE INTERFACE:
                    - PC0 → HD44100H Enable/Strobe + CD4035 Latch Enable
                    - PC1 → CD4035 Clock (shift register timing)
                    - PC2 → HD44100H Command/Data Select
                    - PC3-PC7 → Additional Display System Control (LCD module se...

                    CRITICAL TIMING PROTOCOLS:
                    PC0 - Display Strobe: PORTC &= 0xFE; PORTB = data; PORTC |= 1...
                    PC1 - Serial Clock: PORTC &= 0xFD; PORTC |= 2; sample_data();...
                    PC2 - Command Mode: PORTC &= 0xF3; PORTC |= 4; (command mode ...

                    USAGE PATTERNS:
                    - All responses use PC0 strobe protocol for data validation
                    - Serial reception uses PC1 clock for timing synchronization
                    - Display operations use PC2 for command/data mode selection
                    - PC3-PC7 provide additional control for complex display syst...
                    PORTC                                           XREF[15]:    RESET:00e8(RW),
                                                                                 ProcessData:014b(RW),
                                                                                 WriteToDisplayPort:023d(RW),
                                                                                 WriteToDisplayPort:0240(RW),
                                                                                 WaitForData:0276(RW),
                                                                                 WaitForData:0278(RW),
                                                                                 WaitForData:027a(RW),
                                                                                 WaitForData:0299(RW),
                                                                                 WaitForData:029b(RW),
                                                                                 OutputCharacterToDisplay:05ee(RW...
                                                                                 OutputCharacterToDisplay:05f0(RW...
                                                                                 SendDisplayCommand:05fe(RW),
                                                                                 SendDisplayCommand:0600(RW),
                                                                                 SendDisplayCommand:0602(RW),
                                                                                 SendDisplayCommand:0604(RW)
0002 ff             db          FFh

                    Not_Used
0003 ff             db          FFh

                    DDRA (0x0004) - Data Direction Register A
                    MC68705P3 Port A Direction Control

                    CONFIGURATION: 0x00 (ALL INPUTS)
                    HARDWARE PINS: Controls direction of PA0-PA7 (Pins 16-23)

                    PIN DIRECTIONS SET BY FIRMWARE:
                    Pin 23 PA0 = Input (Serial Data Channel 1)
                    Pin 22 PA1 = Input (Serial Data Channel 2)
                    Pin 21 PA2 = Input (Serial Data Channel 3)
                    Pin 20 PA3 = Input (Panel Lock Key Sense)
                    Pin 19 PA4 = Input (Command/Button Data)
                    Pin 18 PA5 = Input (Command/Button Data)
                    Pin 17 PA6 = Input (Button Change Flag)
                    Pin 16 PA7 = Input (Status/Control)

                    HARDWARE IMPLICATIONS:
                    - All pins configured as high impedance inputs
                    - External pull-up resistors likely required for proper CMOS ...
                    - Input protection may be present for ESD/overvoltage
                    - Button matrix and serial data inputs require clean signal l...
                    DDRA                                            XREF[1]:     RESET:00de(W)
0004 ff             db          FFh
```

```
                    DDRB (0x0005) - Data Direction Register B
                    MC68705P3 Port B Direction Control

                    CONFIGURATION: 0xFF (ALL OUTPUTS)
                    HARDWARE PINS: Controls direction of PB0-PB7 (Pins 6-13)

                    PIN DIRECTIONS SET BY FIRMWARE:
                    Pin  6 PB0 = Output (Response Data Bit 0)
                    Pin  7 PB1 = Output (Response Data Bit 1)
                    Pin  8 PB2 = Output (Response Data Bit 2)
                    Pin  9 PB3 = Output (Response Data Bit 3)
                    Pin 10 PB4 = Output (Response Data Bit 4)
                    Pin 11 PB5 = Output (Response Data Bit 5)
                    Pin 12 PB6 = Output (Response Data Bit 6)
                    Pin 13 PB7 = Output (Response Data Bit 7)

                    HARDWARE IMPLICATIONS:
                    - All pins configured as CMOS push-pull outputs
                    - Must drive HD44100H LCD driver inputs (8-bit parallel)
                    - Must drive external logic for CPU PANS register interface
                    - Current capacity must support multiple parallel loads
                    - Signal integrity critical for display and CPU communication
                    DDRB                                            XREF[1]:     RESET:00d8(W)
0005 ff                 db          FFh

                    DDRC (0x0006) - Data Direction Register C
                    MC68705P3 Port C Direction Control

                    CONFIGURATION: 0xFF (ALL OUTPUTS)
                    HARDWARE PINS: Controls direction of PC0-PC7 (Pins 2-5, 24-27)

                    PIN DIRECTIONS SET BY FIRMWARE:
                    Pin  2 PC0 = Output (Display Data Strobe)
                    Pin 27 PC1 = Output (Serial Clock)
                    Pin  3 PC2 = Output (Display Command Mode)
                    Pin 26 PC3 = Output (Display Control 1)
                    Pin  4 PC4 = Output (Display Control 2)
                    Pin 25 PC5 = Output (Display Control 3)
                    Pin  5 PC6 = Output (Display Control 4)
                    Pin 24 PC7 = Output (Display Control 5)

                    HARDWARE IMPLICATIONS:
                    - All pins configured as CMOS push-pull outputs
                    - PC0/PC1 provide critical timing signals (strobe/clock)
                    - PC2 controls HD44100H command/data mode
                    - PC3-PC7 drive additional display system control
                    - Timing accuracy critical for proper display operation
                    - Signal rise/fall times must meet display driver requirements
                    DDRC                                            XREF[1]:     RESET:00da(W)
0006 ff                 db          FFh
0007 ff                 db          FFh

                    Timer_Data_Reg (0x0008) - MC68705P3 Internal Timer Data Regis…

                    HARDWARE: Internal 8-bit down counter for timing operations
                    CRYSTAL: Connected to Pin 15 (XTAL/EXTAL) - likely 2MHz cryst…
                    TIMING: Internal divide-by-4 → 500kHz instruction cycle

                    FIRMWARE USAGE:
                    - Reset to 0 in initialization and command processing
                    - Used for precise timing delays and synchronization
                    - Coordinates with 20ms CPU interrupt cycle
                    - Provides timing base for display refresh and communication

                    TIMER SYSTEM COORDINATION:
                    - Works with Timer_Control_Reg (0x78 configuration)
                    - Supports CountdownTimer1/Timer2 variables for software timi…
                    - Critical for maintaining panel processor ↔ CPU synchroniza…
                    - Used in InitDisplayClearPulse() for CPU interrupt generation
                    Timer_Data_Reg                                  XREF[3]:     RESET:0100(W),
                                                                                 ProcessData:017b(R),
                                                                                 WaitForData:0264(W)
0008 ff                 db          FFh
```

```
                    Timer_Control_Reg (0x0009) - MC68705P3 Internal Timer Control...

                    HARDWARE: Controls internal timer operation and prescaling
                    OBSERVED VALUE: 0x78 (set in RESET function)

                    CONFIGURATION ANALYSIS (0x78 = 01111000 binary):
                    Bit 7: Unknown function
                    Bit 6: Timer enable/disable control
                    Bit 5: Prescaler configuration
                    Bit 4: Prescaler configuration
                    Bit 3: Timer mode selection
                    Bits 2-0: Additional control bits

                    PURPOSE: Configures timer for:
                    - 20ms synchronization with ND-120 CPU interrupt cycle
                    - Precise timing for serial communication (PA0-PA2 sampling)
                    - Display refresh timing coordination
                    - Button debouncing timing base

                    CRITICAL SYSTEM TIMING: This configuration enables the panel ...
                    to maintain precise timing coordination with the CPU's 20ms i...
                    cycle, essential for reliable command/response communication ...
                    Timer_Control_Reg                               XREF[1]:     RESET:00e2(W)
0009 ff                 db          FFh

                    NOT_USED_0x0A
000a ff                 db          FFh

                    Programming_Control_Reg
000b ff                 db          FFh
000c ff                 ??          FFh
000d ff                 ??          FFh
000e ff                 ??          FFh
000f ff                 ??          FFh

                    CountdownTimer1 - Primary CPU Interrupt Timer
                    Decremented in UpdateTimersAndWait and ProcessData
                    When reaches 0 (along with CountdownTimer2): triggers CPU int...
                    Controls 20ms synchronization with CPU interrupt cycle
                    When both timers expire: calls InitDisplayClearPulse()
                    Resets CountdownTimer2 to 6 after interrupt generation
                    Part of panel processor to CPU interrupt signaling system
                    DAT_0010                                        XREF[1]:     UpdateTimersAndWait:012b(RW)
0010 ff                 ??          FFh

                    CountdownTimer2 - Secondary CPU Interrupt Timer
                    Works with CountdownTimer1 for interrupt timing
                    When both reach 0: CPU interrupt sequence initiated
                    Reset to 6 after interrupt generation
                    Set to 'P' (0x50) in some command sequences
                    Part of panel button interrupt signaling to CPU
                    Coordinates with 20ms CPU timer interrupt cycle
                    DAT_0011                                        XREF[3]:     UpdateTimersAndWait:012f(RW),
                                                                                 UpdateTimersAndWait:013d(W),
                                                                                 ProcessData:018b(W)
0011 ff                 ??          FFh

                    ButtonStateBuffer - Current Button/Command State
                    Holds current PORTA reading for stability checking
                    Used in ProcessData: if (PORTA == ButtonStateBuffer) for debo...
                    Masked with 0x3f to extract command bits
                    Masked with 0xc0 to extract button status bits
                    XOR with PreviousButtonState (0x15) detects changes
                    Bit 6 (0x40) used for button change detection
                    Updated when input is stable after debouncing
                    DAT_0012                                        XREF[8]:     RESET:00fe(W),
                                                                                 UpdateTimersAndWait:011f(R),
                                                                                 UpdateTimersAndWait:0125(W),
                                                                                 ProcessData:014f(R),
                                                                                 ProcessData:0160(R),
                                                                                 ProcessData:016c(R),
                                                                                 ProcessData:0175(R),
                                                                                 cmd_display_update:01db(R)
0012 ff                 ??          FFh
```

```
                    ButtonDebounceCounter - Input Stability Counter
                    Decremented each ProcessData cycle when input is stable
                    When reaches 0: input is considered valid and stable
                    Reset to 5 when input changes (new input detected)
                    Provides software debouncing for button inputs
                    Prevents false triggering from electrical noise
                    Part of reliable button state detection system
                    DAT_0013                                XREF[2]:     UpdateTimersAndWait:0129(W),
                                                                         ProcessData:0153(RW)
0013 ff             ??         FFh

                    DisplayControlFlags - Primary Control Register for Command Pr...

                    BIT LAYOUT:
                    Bit 7: CPU communication status (set/clear based on SerialInp...
                    Bit 6: Additional display control
                    Bit 5: Display enable flag (0x20) - affects display operations
                    Bit 4: Command table select (0=primary@0x80, 1=secondary@0x8B...
                    Bit 3-0: Display mode and addressing control

                    COMMAND PROCESSING ROLE:
                    - Bit 4 determines which lookup table is used for command dis...
                    - Combined with response data in OutputToDisplayDriver()
                    - Modified by various command handlers
                    - Affects command routing and display operation modes

                    STATUS INTEGRATION:
                    - Combined with data in response protocol: (data | DisplayCon...
                    - Provides CPU with panel processor state information
                    - Bit 7 reflects serial input status for communication coordi...

                    STATE MACHINE CONTROL:
                    - Changes to this register affect subsequent command interpre...
                    - Enables dual-mode operation (64 base commands × 2 modes = 1...
                    - Critical for maintaining communication protocol state
                    DAT_0014                                XREF[25]:    RESET:00e6(W), RESET:00ed(RW),
                                                                         RESET:00f1(RW), RESET:00f3(R),
                                                                         UpdateTimersAndWait:0133(RW),
                                                                         ProcessData:0145(RW),
                                                                         ProcessData:0149(RW),
                                                                         ProcessData:0171(R),
                                                                         ProcessData:0173(W),
                                                                         ProcessData:0181(RW),
                                                                         ProcessData:0185(RW),
                                                                         ProcessData:018f(R),
                                                                         CompleteCommandProcessing:01c6(R...
                                                                         cmd_conditional_update:01d1(R),
                                                                         cmd_handler_5e:0208(R),
                                                                         cmd_handler_5e:020b(RW),
                                                                         cmd_handler_5e:020f(RW),
                                                                         cmd_handler_6c:0222(R),
                                                                         cmd_handler_6c:0225(RW),
                                                                         OutputToDisplayDriver:0238(R),
                                                                         [more]
0014 ff             ??         FFh

                    DAT_0015                                XREF[5]:     UpdateTimersAndWait:0123(W),
                                                                         ProcessData:0159(R),
                                                                         ProcessData:0177(W),
                                                                         cmd_handler_5e:0205(R),
                                                                         cmd_handler_6c:021f(R)
0015 ff             ??         FFh
```

```
                    CommandParameter - Current Command Code Storage

                    USAGE: Stores extracted 6-bit command from PORTA (bits 5-0)
                    RANGE: 0x00-0x3F (64 possible commands)

                    SET BY: ProcessData() when stable command detected
                    FORMULA: CommandParameter = PORTA & 0x3F

                    COMMAND CATEGORIES:
                    0x00-0x02: Display update operations
                    0x08-0x0C: Button input polling
                    0x48-0x4A: Direct data output
                    0x51: Special status return
                    0x77-0x7F: Serial data reception

                    LOOKUP PROCESS:
                    1. Command extracted and stored here
                    2. Used to index into command_lookup_table_primary (0x80) or
                      command_lookup_table_secondary (0x8B)
                    3. Table selection based on DisplayControlFlags bit 4
                    4. Results in dispatch code for switch statement execution
                    DAT_0016                                XREF[2]:     ProcessData:0168(W),
                                                                         ProcessData:0195(R)
0016 ff                 ??          FFh

                    DAT_0017                                XREF[2]:     ProcessData:015b(W),
                                                                         ProcessData:015d(R)
0017 ff                 ??          FFh

                    DAT_0018                                XREF[4]:     RESET:00fc(W),
                                                                         WaitForData:02e0(R),
                                                                         WaitForData:02e4(W),
                                                                         WaitForData:0327(R)
0018 ff                 ??          FFh

                    DAT_0019                                XREF[11]:    UpdateTimersAndWait:0135(R),
                                                                         WaitForData:02de(R),
                                                                         WaitForData:02ef(W),
                                                                         WaitForData:033f(RW),
                                                                         WaitForData:034b(RW),
                                                                         WaitForData:03b0(RW),
                                                                         WaitForData:03b2(R),
                                                                         DisplayTimeData:03eb(RW),
                                                                         ShowMessageAndTime:03f9(R),
                                                                         ShowSystemStatusDisplay:041f(W),
                                                                         ShowSystemStatusDisplay:0507(RW)
0019 ff                 ??          FFh

                    DAT_001a                                XREF[4]:     WaitForData:0274(W),
                                                                         WaitForData:02c0(RW),
                                                                         WaitForData:02c2(R),
                                                                         WaitForData:02ca(W)
001a ff                 ??          FFh

                    DAT_001b                                XREF[21]:    WaitForData:02f5(W),
                                                                         WaitForData:0316(W),
                                                                         ShowSystemStatusDisplay:0436(R),
                                                                         ShowSystemStatusDisplay:043e(R),
                                                                         ShowSystemStatusDisplay:0446(R),
                                                                         ShowSystemStatusDisplay:044e(R),
                                                                         ShowSystemStatusDisplay:0473(W),
                                                                         ShowSystemStatusDisplay:0496(R),
                                                                         ShowSystemStatusDisplay:049e(R),
                                                                         ShowSystemStatusDisplay:04b6(W),
                                                                         ShowSystemStatusDisplay:04bf(R),
                                                                         ShowSystemStatusDisplay:04ce(R),
                                                                         ShowSystemStatusDisplay:056e(W),
                                                                         DisplayBinaryBars:057c(R),
                                                                         DisplayBinaryBars:058a(R),
                                                                         DisplayBinaryDigits:05a2(R),
                                                                         DisplayBinaryDigits:05bb(R),
                                                                         LookupCharacterCode:0626(W),
                                                                         LookupCharacterCode:0628(RW),
                                                                         LookupCharacterCode:062f(R),
                                                                         [more]
001b ff                 ??          FFh
```

```
                    DAT_001c                                XREF[11]:   WaitForData:02fa(W),
                                                                        WaitForData:0320(W),
                                                                        ShowSystemStatusDisplay:042e(R),
                                                                        ShowSystemStatusDisplay:0456(R),
                                                                        ShowSystemStatusDisplay:045e(R),
                                                                        ShowSystemStatusDisplay:0466(R),
                                                                        ShowSystemStatusDisplay:0478(W),
                                                                        ShowSystemStatusDisplay:048e(R),
                                                                        ShowSystemStatusDisplay:04a6(R),
                                                                        ShowSystemStatusDisplay:04bb(W),
                                                                        DecodeCharacterFromTable:064c(R)
001c ff             ??          FFh
                    DAT_001d                                XREF[9]:    WaitForData:02ff(W),
                                                                        WaitForData:030a(R),
                                                                        DecodeCharacterFromTable:0645(W),
                                                                        DecodeCharacterFromTable:065c(W),
                                                                        DecodeCharacterFromTable:065e(R),
                                                                        DecodeCharacterFromTable:067f(RW...
                                                                        DecodeCharacterFromTable:0681(R),
                                                                        DecodeCharacterFromTable:0689(W),
                                                                        DecodeCharacterFromTable:068d(R)
001d ff             ??          FFh
                    DAT_001e                                XREF[9]:    cmd_handler_a4:024d(W),
                                                                        cmd_handler_a4:0259(RW),
                                                                        WaitForData:027e(W),
                                                                        WaitForData:028a(RW),
                                                                        DisplayTimeData:03d3(W),
                                                                        DisplayTimeData:03d9(RW),
                                                                        ShowSystemStatusDisplay:0423(W),
                                                                        ShowSystemStatusDisplay:047a(RW),
                                                                        ShowSystemStatusDisplay:0481(R)
001e ff             ??          FFh
                    DAT_001f                                XREF[20]:   WaitForData:030c(W),
                                                                        WaitForData:0311(R),
                                                                        WaitForData:0369(W),
                                                                        WaitForData:036d(R),
                                                                        DisplayTimeData:03cf(W),
                                                                        DisplayTimeData:03de(R),
                                                                        ShowSystemStatusDisplay:042a(W),
                                                                        ShowSystemStatusDisplay:046e(R),
                                                                        ShowSystemStatusDisplay:04e9(W),
                                                                        ShowSystemStatusDisplay:04f3(R),
                                                                        ShowSystemStatusDisplay:050d(R),
                                                                        ShowSystemStatusDisplay:0516(R),
                                                                        ShowSystemStatusDisplay:0528(R),
                                                                        ShowSystemStatusDisplay:0533(R),
                                                                        ShowSystemStatusDisplay:053d(R),
                                                                        ShowSystemStatusDisplay:054d(R),
                                                                        ShowSystemStatusDisplay:0556(R),
                                                                        LookupCharacterCode:0624(W),
                                                                        DecodeCharacterFromTable:0641(W),
                                                                        DecodeCharacterFromTable:068b(R)
001f ff             ??          FFh

                    SerialInputData - Raw CPU Command Data
                    Direct copy of PORTA during serial reception: DAT_0020 = PORTA
                    Contains current CPU command bits and status
                    Bit 0-2: Serial data channels (copied to ShiftRegister1-3)
                    Bit 3: Control/status bit used in DisplayControlFlags updates
                    Bits 4-7: Additional command/status information
                    Used for command validation and processing
                    Source data for shift register operations
                    DAT_0020                                XREF[6]:    ProcessData:0142(R),
                                                                        cmd_conditional_update:01d4(R),
                                                                        WaitForData:029d(W),
                                                                        WaitForData:02a5(R),
                                                                        WaitForData:02ae(R),
                                                                        WaitForData:02b7(R)
0020 ff             ??          FFh
                    DAT_0021                                XREF[4]:    WaitForData:029f(RW),
                                                                        WaitForData:02a8(RW),
                                                                        WaitForData:02ac(RW),
                                                                        WaitForData:02cd(R)
0021 ff             ??          FFh
                    DAT_0022                                XREF[4]:    WaitForData:02a1(RW),
                                                                        WaitForData:02b1(RW),
                                                                        WaitForData:02b5(RW),
                                                                        WaitForData:02d1(R)
0022 ff             ??          FFh
```

```
                DAT_0023                                XREF[4]:    WaitForData:02a3(RW),
                                                                    WaitForData:02ba(RW),
                                                                    WaitForData:02be(RW),
                                                                    WaitForData:02d5(R)
0023 ff             ??          FFh

                DAT_0024                                XREF[6]:    DisplayTimeData:03d7(W),
                                                                    DisplayBinaryDigits:0597(RW),
                                                                    DisplayBinaryDigits:0599(R),
                                                                    DisplayBinaryDigits:05b0(RW),
                                                                    DisplayBinaryDigits:05b2(R),
                                                                    DisplayDecimalPoint:05cf(W)
0024 ff             ??          FFh

                MessageBuffer0                          XREF[5]:    WaitForData:030f(W),
                                                                    WaitForData:035d(R),
                                                                    WaitForData:0393(R),
                                                                    WaitForData:03b5(R),
                                                                    ShowMessageAndTime:03fe(R)
0025 ff             undefined1  FFh

                MessageBuffer1                          XREF[3]:    WaitForData:030f(W),
                                                                    WaitForData:0393(R),
                                                                    ShowMessageAndTime:03fe(R)
0026 ff             undefined1  FFh

                MessageBuffer2                          XREF[3]:    WaitForData:032c(R),
                                                                    WaitForData:034d(R),
                                                                    WaitForData:0386(R)
0027 ff             undefined1  FFh

                MessageBuffer3                          XREF[2]:    WaitForData:032c(R),
                                                                    WaitForData:034d(R)
0028 ff             undefined1  FFh

                StoredMessageBuffer0                    XREF[1]:    WaitForData:0395(W)
0029 ff             undefined1  FFh

                StoredMessageBuffer1                    XREF[2]:    WaitForData:035f(R),
                                                                    WaitForData:0395(W)
002a ff             undefined1  FFh

                StoredMessageBuffer2                    XREF[3]:    WaitForData:032e(W),
                                                                    WaitForData:034f(R),
                                                                    WaitForData:035f(R)
002b ff             undefined1  FFh

                StoredMessageBuffer3                    XREF[2]:    WaitForData:032e(W),
                                                                    WaitForData:034f(R)
002c ff             undefined1  FFh

                TimeDataBuffer0 - Raw Time/Date Data Reception Area
                First byte of 8-byte time data buffer (0x2D-0x34)
                Stores data from ShiftRegister1 (DAT_0021) during serial rece...
                Contains raw time/date information received from CPU
                Used by DecodeCharacterFromTable() for character conversion
                Part of 3-channel parallel data reception system
                TimeDataBuffer0                         XREF[1]:    WaitForData:02f3(R)
002d ff             undefined1  FFh

                TimeDataBuffer1                         XREF[2]:    WaitForData:02cf(W),
                                                                    WaitForData:02f8(R)
002e ff             undefined1  FFh

                time_data_buffer_2                      XREF[1]:    WaitForData:0314(R)
002f ff             undefined1  FFh

                time_data_buffer_3                      XREF[1]:    WaitForData:031e(R)
0030 ff             undefined1  FFh
0031 ff             ??          FFh
0032 ff             ??          FFh
0033 ff             ??          FFh

                time_data_buffer_7                      XREF[1]:    WaitForData:02cf(W)
0034 ff             undefined1  FFh
```

```
                    TimeDisplayBuffer0 - Formatted Time Display Data
                    First byte of 8-byte display buffer (0x35-0x3C)
                    Stores data from ShiftRegister2 (DAT_0022) during serial rece...
                    Contains formatted time/date for display output
                    Used by LookupCharacterCode() and display functions
                    Fed to LD-H7919 LCD modules for time display
                    TimeDisplayBuffer0                              XREF[8]:     ShowSystemStatusDisplay:04eb(R),
                                                                                 ShowSystemStatusDisplay:04f5(R),
                                                                                 ShowSystemStatusDisplay:050e(R),
                                                                                 ShowSystemStatusDisplay:0518(R),
                                                                                 ShowSystemStatusDisplay:0535(R),
                                                                                 ShowSystemStatusDisplay:053f(R),
                                                                                 ShowSystemStatusDisplay:054e(R),
                                                                                 ShowSystemStatusDisplay:0558(R)
0035 ff                 undefined1 FFh

                    time_display_buffer_1                           XREF[1]:     WaitForData:02d3(W)
0036 ff                 undefined1 FFh

                    time_display_buffer_2                           XREF[1]:     ShowSystemStatusDisplay:054e(R)
0037 ff                 undefined1 FFh

                    DAT_0038                                        XREF[3]:     ShowSystemStatusDisplay:04db(R),
                                                                                 ShowSystemStatusDisplay:04fd(R),
                                                                                 ShowSystemStatusDisplay:0520(R)
0038 ff                 ??         FFh
0039 ff                 ??         FFh

                    DAT_003a                                        XREF[3]:     WaitForData:0341(R),
                                                                                 WaitForData:0356(R),
                                                                                 WaitForData:039d(R)
003a ff                 ??         FFh
003b ff                 ??         FFh

                    time_display_buffer_7                           XREF[2]:     WaitForData:02d3(W),
                                                                                 ShowSystemStatusDisplay:0502(R)
003c ff                 undefined1 FFh

                    StatusDataBuffer0 - System Status Display Data
                    First byte of 8-byte status buffer (0x3D-0x44)
                    Stores data from ShiftRegister3 (DAT_0023) during serial rece...
                    Contains CPU status information (utilization, cache, protecti...
                    Used by DisplayBinaryBars() for status indicator display
                    Controls UTIL, HIT, RING, MODE function displays
                    StatusDataBuffer0                               XREF[1]:     ShowSystemStatusDisplay:056d(R)
003d ff                 undefined1 FFh

                    status_data_buffer_1                            XREF[2]:     WaitForData:02d7(W),
                                                                                 ShowSystemStatusDisplay:056d(R)
003e ff                 undefined1 FFh
003f ff                 ??         FFh
0040 ff                 ??         FFh
0041 ff                 ??         FFh
0042 ff                 ??         FFh
0043 ff                 ??         FFh

                    status_data_buffer_7                            XREF[1]:     WaitForData:02d7(W)
0044 ff                 undefined1 FFh

                    DAT_0045                                        XREF[4]:     RESET:00fa(W),
                                                                                 WaitForData:033d(W),
                                                                                 WaitForData:03a4(W),
                                                                                 WaitForData:03ab(R)
0045 ff                 ??         FFh

                    DAT_0046                                        XREF[6]:     CalculateDisplayPosition:0698(W),
                                                                                 CalculateDisplayPosition:06a2(W),
                                                                                 CalculateDisplayPosition:06a8(W),
                                                                                 SetDisplayAddressAndWrite:06b0(W...
                                                                                 SetDisplayAddressAndWrite:06b4(R...
                                                                                 SetDisplayAddressAndWrite:06b6(R...
0046 ff                 ??         FFh

                    DAT_0047                                        XREF[7]:     WaitForData:02eb(W),
                                                                                 WaitForData:0339(W),
                                                                                 WaitForData:0370(R),
                                                                                 WaitForData:0378(W),
                                                                                 WaitForData:037f(RW),
                                                                                 CalculateDisplayPosition:0696(R),
                                                                                 CalculateDisplayPosition:06a6(R)
0047 ff                 ??         FFh
```

```
                    DAT_0048                                XREF[3]:    OutputCharacterToDisplay:05d4(R),
                                                                        OutputCharacterToDisplay:05f2(RW...
                                                                        SetDisplayAddressAndWrite:06b2(W...
0048 ff             ??          FFh

                    DAT_0049                                XREF[2]:    OutputCharacterToDisplay:05d2(W),
                                                                        OutputCharacterToDisplay:05ea(R)
0049 ff             ??          FFh
004a ff             ??          FFh
004b ff             ??          FFh

                    DAT_004c                                XREF[3]:    cmd_handler_a4:0251(W),
                                                                        WaitForData:027c(R),
                                                                        WaitForData:0290(W)
004c ff             ??          FFh
004d ff             ??          FFh
004e ff             ??          FFh
004f ff             ??          FFh
0050 ff             ??          FFh
0051 ff             ??          FFh
0052 ff             ??          FFh
0053 ff             ??          FFh
0054 ff             ??          FFh
0055 ff             ??          FFh
0056 ff             ??          FFh
0057 ff             ??          FFh
0058 ff             ??          FFh
0059 ff             ??          FFh
005a ff             ??          FFh
005b ff             ??          FFh
005c ff             ??          FFh
005d ff             ??          FFh
005e ff             ??          FFh
005f ff             ??          FFh
0060 ff             ??          FFh
0061 ff             ??          FFh
0062 ff             ??          FFh
0063 ff             ??          FFh
0064 ff             ??          FFh
0065 ff             ??          FFh
0066 ff             ??          FFh
0067 ff             ??          FFh
0068 ff             ??          FFh
0069 ff             ??          FFh
006a ff             ??          FFh
006b ff             ??          FFh
006c ff             ??          FFh
006d ff             ??          FFh
006e ff             ??          FFh
006f ff             ??          FFh
0070 ff             ??          FFh
0071 ff             ??          FFh
0072 ff             ??          FFh
0073 ff             ??          FFh
0074 ff             ??          FFh
0075 ff             ??          FFh
0076 ff             ??          FFh
0077 ff             ??          FFh
0078 ff             ??          FFh
0079 ff             ??          FFh
007a ff             ??          FFh
007b ff             ??          FFh
007c ff             ??          FFh
007d ff             ??          FFh
007e ff             ??          FFh
007f ff             ??          FFh                         STACK START (Growing down max 31...
```

```
                    command_lookup_table_primary - Primary Command Dispatch Table

                    USAGE: Used when DisplayControlFlags bit 4 is clear (normal m…
                    SELECTION: if ((DisplayControlFlags & 0x10) == 0) table = 0x80

                    TABLE STRUCTURE:
                    [command_code_0][dispatch_flags_0]
                    [command_code_1][dispatch_flags_1]
                    ...
                    Each entry maps 6-bit command codes (0x00-0x3F) to dispatch i…

                    DISPATCH MECHANISM:
                    1. Extract command: cmd = PORTA & 0x3F
                    2. Lookup entry: table[cmd + 1]
                    3. Generate dispatch: dispatch = table[cmd + 1] << 1
                    4. Execute: switch(dispatch) with cases 0x00-0xFE (even numbe…

                    KNOWN MAPPINGS:
                    Command 0x00 → Dispatch 0x00 → cmd_display_update
                    Command 0x01 → Dispatch 0x02 → cmd_conditional_update
                    Command 0x02 → Dispatch 0x04 → cmd_multi_stage_update
                    command_lookup_table_primary                    XREF[1]:     ProcessData:0194(R)
0080 01                 undefined1 01h
0081 01                 ??         01h
0082 02                 ??         02h
0083 00                 ??         00h
0084 10                 ??         10h
0085 05                 ??         05h
0086 20                 ??         20h
0087 06                 ??         06h
0088 05                 ??         05h
0089 07                 ??         07h
008a ff                 ??         FFh

                    command_lookup_table_secondary - Secondary Command Dispatch T…

                    USAGE: Used when DisplayControlFlags bit 4 is set (extended m…
                    SELECTION: if ((DisplayControlFlags & 0x10) != 0) table = 0x8B

                    PURPOSE: Provides alternate command interpretation mode for e…
                    Same dispatch mechanism as primary table but different comman…

                    DISPATCH MECHANISM:
                    1. Extract command: cmd = PORTA & 0x3F
                    2. Lookup entry: table[cmd + 1]
                    3. Generate dispatch: dispatch = table[cmd + 1] << 1
                    4. Execute: switch(dispatch) with same case handlers

                    EXTENDED FUNCTIONALITY: Allows same 6-bit command space to ac…
                    different handlers based on system state, effectively doublin…
                    available command set from 64 to 128 total commands.
                    command_lookup_table_secondary                  XREF[1]:     ProcessData:0194(R)
008b 01                 undefined1 01h

                    command_table_entry_1                           XREF[1]:     ProcessData:01a2(R)
008c 01                 undefined1 01h

                    command_table_entry_2                           XREF[1]:     ProcessData:0194(R)
008d 02                 undefined1 02h
008e 02                 ??         02h
008f 04                 ??         04h
0090 03                 ??         03h
0091 08                 ??         08h
0092 04                 ??         04h
0093 10                 ??         10h
0094 05                 ??         05h
0095 20                 ??         20h
0096 06                 ??         06h
0097 ff                 ??         FFh

                    STRING_ON
0098 4f 4e 20           ds         "ON "
009b 22                 char       '"'

                    STRING_OFF
009c 4f 46 46           ds         "OFF"
009f 22                 char       '"'

                    STRING_DAY
00a0 44 41 59 3a        ds         "DAY:"
00a4 22                 char       '"'
```

```
                    STRING_TIME
00a5 20 20 54           ds          "  TIME:"
     49 4d 45 3a
00ac 22                 char        '"'

                    STRING_UTC
00ad 20 20 20           ds          "   UTC:"
     55 54 43 3a
00b4 22                 char        '"'

                    STRING_ADDRESS
00b5 41 44 44           ds          "ADDRESS:"
     52 45 53
     53 3a
00bd 22                 char        '"'

                    STRING_COUNT
00be 50 20 43           ds          "P COUNT:"
     4f 55 4e
     54 3a
00c6 22                 char        '"'

                    STRING_YEAR
00c7 59 45 41           ds          "YEAR:"
     52 3a
00cc 22                 char        '"'
00cd 20 20 4d           ds          "  MONTH:"
     4f 4e 54
     48 3a
00d5 22                 char        '"'

                    ****************************************************************
                    *                          FUNCTION                            *
                    ****************************************************************
                    undefined RESET()
undefined           <UNASSIGNED>  <RETURN>
                    RESET - MC68705P3 Hardware Initialization and System Startup

                    MC68705P3 PACKAGE: 28-pin DIP (Dual In-Line Package)
                    POWER SUPPLY PINS:
                    Pin  1 VBB = -5V (Negative supply for EPROM programming)
                    Pin 14 VSS = 0V (Ground reference)
                    Pin 28 VDD = +5V (Main logic power supply)
                    Pin 15 XTAL/EXTAL = Crystal/External Clock (likely 2MHz)

                    COMPLETE PIN CONFIGURATION SUMMARY:
                    PORTA (PA0-PA7, Pins 16-23): ALL INPUTS - Command reception, ...
                    PORTB (PB0-PB7, Pins 6-13):  ALL OUTPUTS - Display data, CPU ...
                    PORTC (PC0-PC7, Pins 2-5,24-27): ALL OUTPUTS - Control signal...

                    HARDWARE INITIALIZATION SEQUENCE:
                    1. Configure all port directions (DDRA=0x00, DDRB=0xFF, DDRC=...
                    2. Set timer control register (Timer_Control_Reg = 0x78)
                    3. Initialize display control signals (PORTC |= 2)
                    4. Detect panel lock key position (PORTA & 8)
                    5. Set initial display control flags based on lock key
                    6. Initialize display system (cmd_handler_a4, SendDisplayComm...
                    7. Synchronize with CPU communication (readIRQ polling)
                    8. Begin main command processing loop

                    SYSTEM INTEGRATION: This initialization establishes the 68705...
                    intelligent peripheral controller managing complete operator ...
                    for ND-120 minicomputer system with real-time display, button...
                    and bidirectional CPU communication capabilities.
                    RESET                                           XREF[4]:     07f8(*), 07fa(*), 07fc(*),
                                                                                 07fe(*)
00d6 a6 ff              LDA         #0xff
00d8 b7 05              STA         DDRB                                        = FFh
00da b7 06              STA         DDRC                                        = FFh
00dc a6 00              LDA         #0x0
00de b7 04              STA         DDRA                                        = FFh
00e0 a6 78              LDA         #0x78
00e2 b7 09              STA         Timer_Control_Reg                           = FFh
00e4 a6 60              LDA         #0x60
00e6 b7 14              STA         DAT_0014                                    = FFh
00e8 12 02              BSET        0x1,PORTC                                   = FFh
00ea 06 00 04           BRSET       0x3,PORTA,LAB_00f1
00ed 1f 14              BCLR        0x7,DAT_0014                                = FFh
00ef 20 02              BRA         LAB_00f3

                    LAB_00f1                                        XREF[1]:     00ea(j)
00f1 1e 14              BSET        0x7,DAT_0014                                = FFh
```

```
                LAB_00f3                                XREF[1]:    00ef(j)
00f3 b6 14          LDA         DAT_0014                            = FFh
00f5 cd 02 3c       JSR         WriteToDisplayPort                  undefinedWriteToDisplayPort()
00f8 a6 00          LDA         #0x0
00fa b7 45          STA         DAT_0045                            = FFh
00fc b7 18          STA         DAT_0018                            = FFh
00fe b7 12          STA         DAT_0012                            = FFh
0100 b7 08          STA         Timer_Data_Reg                      = FFh
0102 cd 02 4b       JSR         switchD_01a4::cmd_handler_a4        undefinedcmd_handler_a4()
0105 a6 38          LDA         #0x38
0107 cd 05 fc       JSR         SendDisplayCommand                  undefinedSendDisplayCommand()
010a a6 0c          LDA         #0xc
010c cd 05 fc       JSR         SendDisplayCommand                  undefinedSendDisplayCommand()
010f a6 06          LDA         #0x6
0111 cd 05 fc       JSR         SendDisplayCommand                  undefinedSendDisplayCommand()
0114 cd 06 18       JSR         InitDisplayClearPulse               undefinedInitDisplayClearPulse()

                LAB_0117                                XREF[1]:    0119(j)
0117 2f 02          BIH         LAB_011b
0119 20 fc          BRA         LAB_0117

                LAB_011b                                XREF[2]:    0117(j), 011d(j)
011b 2e 02          BIL         UpdateTimersAndWait
011d 20 fc          BRA         LAB_011b

                ****************************************************************
                *                          FUNCTION                            *
                ****************************************************************
                undefined UpdateTimersAndWait()
 undefined          <UNASSIGNED>  <RETURN>
                UpdateTimersAndWait                     XREF[2]:    RESET:011b(j),
                                                                    CompleteCommandProcessing:01ce(c...
011f b6 12          LDA         DAT_0012                            = FFh
0121 a4 c0          AND         #0xc0
0123 b7 15          STA         DAT_0015                            = FFh

                LAB_0125                                XREF[1]:    ProcessData:0151(j)
0125 b7 12          STA         DAT_0012                            = FFh
0127 a6 05          LDA         #0x5
0129 b7 13          STA         DAT_0013                            = FFh

                LAB_012b                                XREF[2]:    ProcessData:0155(j),
                                                                    ProcessData:0179(j)
012b 3a 10          DEC         DAT_0010                            = FFh
012d 26 10          BNE         LAB_013f
012f 3a 11          DEC         DAT_0011                            = FFh
0131 26 0c          BNE         LAB_013f
0133 19 14          BCLR        0x4,DAT_0014                        = FFh
0135 0a 19 07       BRSET       0x5,DAT_0019,LAB_013f               = FFh
0138 cd 06 18       JSR         InitDisplayClearPulse               undefinedInitDisplayClearPulse()
013b a6 06          LDA         #0x6
013d b7 11          STA         DAT_0011                            = FFh

                LAB_013f                                XREF[3]:    012d(j), 0131(j), 0135(j)
013f cc 02 5e       JMP         WaitForData                         undefinedWaitForData()
                -- Flow Override: CALL_RETURN (CALL_TERMINATOR)
```

```
                    ************************************************************
                    *                          FUNCTION                        *
                    ************************************************************
                    undefined ProcessData()
undefined           <UNASSIGNED>  <RETURN>
                    ProcessData - Main Command Processing State Machine

                    COMMAND PROCESSING OVERVIEW:
                    - Extracts 6-bit commands from PORTA (bits 5-0): range 0x00-0…
                    - Uses dual lookup table system: 0x80 (primary) or 0x8B (seco…
                    - Table selection based on DisplayControlFlags bit 4
                    - Executes massive switch statement with 128+ cases (dispatch…

                    COMMAND FORMAT (PORTA):
                    Bit 7: Status/control bit
                    Bit 6: Button change flag (0x40)
                    Bits 5-0: 6-bit command code (0x00-0x3F)

                    RESPONSE PROTOCOL:
                    - 7-bit data output via PORTB (bit 7 masked)
                    - Status integration combines data with DisplayControlFlags
                    - Strobe signaling via PORTC bit 0 indicates valid data
                    - Fixed timing delays ensure signal stability

                    IDENTIFIED COMMAND CATEGORIES:
                    0x00-0x02: Display update operations (complex multi-stage)
                    0x08-0x0C: Button input polling (waits for release)
                    0x48-0x4A: Direct data output (raw transmission)
                    0x77-0x7F: Serial data reception (192-bit packets)
                    ProcessData                                     XREF[2]:    WaitForData:0292(c),
                                                                                ShowSystemStatusDisplay:0579(c)
0142 06 20 04           BRSET       0x3,DAT_0020,LAB_0149                       = FFh
0145 1f 14              BCLR        0x7,DAT_0014                                = FFh
0147 20 02              BRA         LAB_014b

                    LAB_0149                                        XREF[1]:    0142(j)
0149 1e 14              BSET        0x7,DAT_0014                                = FFh

                    LAB_014b                                        XREF[1]:    0147(j)
014b 13 02              BCLR        0x1,PORTC                                   = FFh
014d b6 00              LDA         PORTA
014f b1 12              CMP         DAT_0012                                    = FFh
0151 26 d2              BNE         LAB_0125
0153 3a 13              DEC         DAT_0013                                    = FFh
0155 26 d4              BNE         LAB_012b
0157 b6 00              LDA         PORTA
0159 b8 15              EOR         DAT_0015                                    = FFh
015b b7 17              STA         DAT_0017                                    = FFh
015d 0c 17 0c           BRSET       0x6,DAT_0017,LAB_016c                       = FFh
0160 b6 12              LDA         DAT_0012                                    = FFh
0162 a4 3f              AND         #0x3f
0164 a1 00              CMP         #0x0
0166 27 0d              BEQ         LAB_0175
0168 b7 16              STA         DAT_0016                                    = FFh
016a 20 1d              BRA         LAB_0189

                    LAB_016c                                        XREF[1]:    015d(j)
016c 0c 12 0c           BRSET       0x6,DAT_0012,LAB_017b                       = FFh
016f a6 60              LDA         #0x60
0171 ba 14              ORA         DAT_0014                                    = FFh
0173 b7 14              STA         DAT_0014                                    = FFh

                    LAB_0175                                        XREF[3]:    0166(j), 0183(j), 0187(j)
0175 b6 12              LDA         DAT_0012                                    = FFh
0177 b7 15              STA         DAT_0015                                    = FFh
0179 20 b0              BRA         LAB_012b

                    LAB_017b                                        XREF[1]:    016c(j)
017b b6 08              LDA         Timer_Data_Reg                              = FFh
017d a1 00              CMP         #0x0
017f 26 04              BNE         LAB_0185
0181 1a 14              BSET        0x5,DAT_0014                                = FFh
0183 20 f0              BRA         LAB_0175

                    LAB_0185                                        XREF[1]:    017f(j)
0185 1b 14              BCLR        0x5,DAT_0014                                = FFh
0187 20 ec              BRA         LAB_0175

                    LAB_0189                                        XREF[1]:    016a(j)
0189 a6 50              LDA         #0x50
018b b7 11              STA         DAT_0011                                    = FFh
018d ae 80              LDX         #0x80
018f 09 14 02           BRCLR       0x4,DAT_0014,LAB_0194                       = FFh
0192 ae 8b              LDX         #0x8b
```

```
                    LAB_0194                                        XREF[2]:     018f(j), 019f(j)
0194 f6                 LDA         X=>command_lookup_table_secondary               = 01h
                                                                                    = 02h
0195 b1 16              CMP         DAT_0016                                        = FFh
0197 27 08              BEQ         LAB_01a1
0199 a1 ff              CMP         #0xff
019b 27 31              BEQ         LAB_01ce
019d 5c                 INCX
019e 5c                 INCX
019f 20 f3              BRA         LAB_0194

                    LAB_01a1                                        XREF[1]:     0197(j)
01a1 5c                 INCX
                    Special Status Return Handler - Command 0x51 (Dispatch 0xA2)

                    FUNCTION: Enhanced status return with timer and command state
                    USAGE: Special command completion with extended status inform...

                    PROCESSING:
                    CountdownTimer2 = 0x50;           // Set timer to specific v...
                    CommandParameter = command;       // Store current command f...
                    return input | lookup_table_entry; // Return combined status

                    RETURN FORMAT: Combines current input with lookup table data
                    providing enhanced status information to CPU

                    PURPOSE: Special command used for:
                    - Timer coordination (sets specific countdown value)
                    - Command state preservation (stores current command)
                    - Enhanced status reporting (combined data return)

                    TIMER SIGNIFICANCE: Value 0x50 (80) likely coordinates with
                    20ms CPU interrupt cycle for synchronized communication.

                    USAGE CONTEXT: Called for specific commands requiring enhanced
                    status feedback and timer coordination with CPU operations.
01a2 fe                 LDX         X=>command_table_entry_1                        = 01h
01a3 58                 ASLX

                    switchD_01a4::switchD
01a4 dc 01 a7           JMP         0x1a7,X

                    switchD_01a4::caseD_0                           XREF[1]:     01a4(j)
01a7 20 32              BRA         switchD_01a4::cmd_display_update                undefined cmd_display_update()
                    -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                    switchD_01a4::caseD_2                           XREF[1]:     01a4(j)
01a9 20 26              BRA         switchD_01a4::cmd_conditional_update            undefined cmd_conditional_update()
                    -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                    switchD_01a4::caseD_4                           XREF[1]:     01a4(j)
01ab 20 40              BRA         switchD_01a4::cmd_multi_stage_update            undefined cmd_multi_stage_update()
                    -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                    switchD_01a4::caseD_6                           XREF[1]:     01a4(j)
01ad 20 4e              BRA         switchD_01a4::cmd_handler_56                    undefined cmd_handler_56()
                    -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                    switchD_01a4::caseD_8                           XREF[1]:     01a4(j)
01af 20 50              BRA         switchD_01a4::cmd_handler_5a                    undefined cmd_handler_5a()
                    -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                    switchD_01a4::caseD_a                           XREF[1]:     01a4(j)
01b1 20 52              BRA         switchD_01a4::cmd_handler_5e                    undefined cmd_handler_5e()
                    -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                    switchD_01a4::caseD_c                           XREF[1]:     01a4(j)
01b3 20 5e              BRA         switchD_01a4::cmd_handler_6c                    undefined cmd_handler_6c()
                    -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                    switchD_01a4::caseD_e                           XREF[1]:     01a4(j)
01b5 20 20              BRA         switchD_01a4::caseD_30
```

```
                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined cmd_handler_10()
undefined           <UNASSIGNED>   <RETURN>
                    Button Input Polling Handler - Commands 0x08-0x0C (Dispatch 0...

                    FUNCTION: Waits for button release and returns final button s...
                    USAGE: Part of button processing sequence for debouncing

                    POLLING LOGIC:
                    do {
                        input = PORTA;                // Read current input
                        input = input & 0x3F;         // Mask to command bits (6...
                        is_zero = (input == 0);       // Check if all buttons re...
                    } while (!is_zero);               // Continue until buttons ...

                    RETURN: Final button state value after release detected

                    PURPOSE: Ensures clean button state transitions by waiting fo...
                    button release before proceeding. Used in conjunction with so...
                    debouncing (ButtonDebounceCounter) for reliable button input ...

                    BUTTON ENCODING: Uses same 6-bit field as commands (PORTA bit...
                    allowing button presses to be processed as command-like input...
                    proper debouncing and state management.
                    switchD_01a4::cmd_handler_10                    XREF[4]:     ProcessData:01a4(j), 01bd(j),
                                                                                 CompleteCommandProcessing:01cb(c...
                                                                                 cmd_multi_stage_update:01f2(c)
01b7 b6 00              LDA         PORTA

                    switchD_01a4::caseD_12                          XREF[1]:     ProcessData:01a4(j)
01b9 a4 3f              AND         #0x3f

                    switchD_01a4::caseD_14                          XREF[1]:     ProcessData:01a4(j)
01bb a1 00              CMP         #0x0

                    switchD_01a4::caseD_16                          XREF[1]:     ProcessData:01a4(j)
01bd 26 f8              BNE         switchD_01a4::cmd_handler_10

                    switchD_01a4::caseD_18                          XREF[1]:     ProcessData:01a4(j)
01bf 81                 RTS

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined CompleteCommandProcessing()
undefined           <UNASSIGNED>   <RETURN>
                    CompleteCommandProcessing - Standard Command Completion Seque...

                    FUNCTION: Standard multi-stage command completion with status...
                    USAGE: Called by command handlers to finalize operations

                    COMPLETION SEQUENCE:
                    OutputToDisplayDriver();          // Send current status wit...
                    OutputToDisplayDriver();          // Send status again (conf...
                    WriteToDisplayPort(DisplayControlFlags); // Send final status...
                    cmd_handler_10();                // Execute standard cleanup...
                    UpdateTimersAndWait();           // Update timers and wait f...

                    PURPOSE: Ensures CPU receives consistent status information a...
                    timing synchronization via timer coordination. Multi-stage ou...
                    robust communication protocol for command acknowledgment.
                    caseD_1a (01c0+1)                               XREF[4,1]:   cmd_conditional_update:01d9(c),
                    CompleteCommandProcessing                                    cmd_display_update:01eb(c),
                                                                                 cmd_handler_56:01ff(c),
                                                                                 cmd_handler_5a:0203(c),
                                                                                 ProcessData:01a4(j)
01c0 cd 02 38           JSR         OutputToDisplayDriver                          undefinedOutputToDisplayDriver()

                    caseD_1e (01c3+2)                               XREF[1,1]:   ProcessData:01a4(j),
                    switchD_01a4::caseD_1c                                       ProcessData:01a4(j)
01c3 cd 02 38           JSR         OutputToDisplayDriver                          undefinedOutputToDisplayDriver()
```

```
                caseD_20 (01c6+1)                               XREF[11,1]:  cmd_conditional_update:01d4(j),
                LAB_01c6                                                     cmd_display_update:01db(j),
                                                                             cmd_multi_stage_update:01fb(j),
                                                                             cmd_handler_5e:0205(j),
                                                                             cmd_handler_5e:020d(j),
                                                                             cmd_handler_5e:0211(j),
                                                                             cmd_handler_6c:021f(j),
                                                                             cmd_handler_6c:0227(j),
                                                                             cmd_handler_6c:022b(j),
                                                                             cmd_handler_6c:0232(j),
                                                                             cmd_handler_6c:0236(j),
                                                                             ProcessData:01a4(j)
01c6 b6 14          LDA         DAT_0014                                        = FFh
                caseD_22 (01c8+1)                               XREF[0,1]:   ProcessData:01a4(j)
01c8 cd 02 3c       JSR         WriteToDisplayPort                              undefinedWriteToDisplayPort()

                caseD_26 (01cb+2)                               XREF[1,1]:   ProcessData:01a4(j),
                switchD_01a4::caseD_24                                       ProcessData:01a4(j)
01cb cd 01 b7       JSR         switchD_01a4::cmd_handler_10                    undefinedcmd_handler_10()

                caseD_28 (01ce+1)                               XREF[1,1]:   ProcessData:019b(j),
                LAB_01ce                                                     ProcessData:01a4(j)
01ce cc 01 1f       JMP         UpdateTimersAndWait                             undefinedUpdateTimersAndWait()
                -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                *************************************************************
                *                          FUNCTION                         *
                *************************************************************
                undefined cmd_conditional_update()
 undefined          <UNASSIGNED>  <RETURN>
                cmd_conditional_update - Command 0x01 (Dispatch 0x02)

                FUNCTION: Conditional display update based on control flags
                RESPONSE: Status byte via CompleteCommandProcessing(1) or dir...

                PROCESSING LOGIC:
                if ((DisplayControlFlags & 0x10 == 0) && (SerialInputData & 8...
                   WriteToDisplayPort(DisplayControlFlags); // Output status
                    cmd_handler_10();              // Execute standard handl...
                    UpdateTimersAndWait();         // Update and wait
                } else {
                   CompleteCommandProcessing(1);    // Complete with respons...
                }

                INTERNAL STATE CHANGES: Conditional based on DisplayControlFl...
                SUBROUTINES: WriteToDisplayPort, cmd_handler_10, CompleteComm...
                caseD_2c (01d1+2)                               XREF[2,1]:   ProcessData:01a4(j),
                switchD_01a4::cmd_conditional_update                         ProcessData:01a9(c),
                                                                             ProcessData:01a4(j)
01d1 08 14 03       BRSET       0x4,DAT_0014,switchD_01a4::caseD_30              = FFh
                caseD_2e (01d4+1)                               XREF[0,1]:   ProcessData:01a4(j)
01d4 06 20 ef       BRSET       0x3,DAT_0020,LAB_01c6                            = FFh

                switchD_01a4::caseD_30                          XREF[3]:     ProcessData:01a4(j),
                                                                             ProcessData:01b5(j), 01d1(j)
01d7 a6 01          LDA         #0x1

                switchD_01a4::caseD_32                          XREF[1]:     ProcessData:01a4(j)
01d9 20 e5          BRA         CompleteCommandProcessing                       undefinedCompleteCommandProcess...
                -- Flow Override: CALL_RETURN (CALL_TERMINATOR)
```

```
                    ************************************************************
                    *                          FUNCTION                        *
                    ************************************************************
                    undefined cmd_display_update()
undefined           <UNASSIGNED>  <RETURN>
                    cmd_display_update - Command 0x00 (Dispatch 0x00)

                    FUNCTION: Primary display update operation
                    RESPONSE: Status byte via CompleteCommandProcessing(4)

                    PROCESSING LOGIC:
                    if (ButtonStateBuffer & 0x80 == 0) {
                        OutputToDisplayDriver(2);       // Send data=2 with stat…
                        OutputToDisplayDriver();        // Send status only
                        cmd_handler_a4();              // Execute display update…
                        CompleteCommandProcessing(4);   // Complete with respons…
                    } else {
                       WriteToDisplayPort(DisplayControlFlags); // Direct status…
                        cmd_handler_10();              // Execute standard handl…
                        UpdateTimersAndWait();         // Update timers and wait
                    }

                    INTERNAL STATE CHANGES: Updates DisplayControlFlags, executes…
                    SUBROUTINES: OutputToDisplayDriver, cmd_handler_a4, CompleteC…
                    caseD_36 (01db+2)                               XREF[2,1]:   ProcessData:01a4(j),
                    switchD_01a4::cmd_display_update                             ProcessData:01a7(c),
                                                                                 ProcessData:01a4(j)
01db 0e 12 e8           BRSET       0x7,DAT_0012,LAB_01c6                        = FFh
                    caseD_38 (01de+1)                               XREF[0,1]:   ProcessData:01a4(j)
01de a6 02              LDA         #0x2
                    caseD_3a (01e0+1)                               XREF[0,1]:   ProcessData:01a4(j)
01e0 cd 02 38           JSR         OutputToDisplayDriver                        undefinedOutputToDisplayDriver()

                    caseD_3e (01e3+2)                               XREF[1,1]:   ProcessData:01a4(j),
                    switchD_01a4::caseD_3c                                       ProcessData:01a4(j)
01e3 cd 02 38           JSR         OutputToDisplayDriver                        undefinedOutputToDisplayDriver()
                    caseD_40 (01e6+1)                               XREF[0,1]:   ProcessData:01a4(j)
01e6 cd 02 4b           JSR         switchD_01a4::cmd_handler_a4                 undefinedcmd_handler_a4()

                    switchD_01a4::caseD_42                          XREF[1]:     ProcessData:01a4(j)
01e9 a6 04              LDA         #0x4

                    switchD_01a4::caseD_44                          XREF[1]:     ProcessData:01a4(j)
01eb 20 d3              BRA         CompleteCommandProcessing                    undefinedCompleteCommandProcess…
                    -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                    ************************************************************
                    *                          FUNCTION                        *
                    ************************************************************
                    undefined cmd_multi_stage_update()
undefined           <UNASSIGNED>  <RETURN>
                    cmd_multi_stage_update - Command 0x02 (Dispatch 0x04)

                    FUNCTION: Complex multi-stage display operation
                    RESPONSE: Multiple outputs via OutputToDisplayDriver and Writ…

                    PROCESSING LOGIC:
                    OutputToDisplayDriver(2);          // Send data=2 with status
                    cmd_handler_10();                 // Execute standard handler
                    OutputToDisplayDriver();          // Send status only
                    cmd_handler_a4();                // Execute display sequence…
                    WriteToDisplayPort(DisplayControlFlags); // Output final stat…
                    cmd_handler_10();                // Execute handler again
                    UpdateTimersAndWait();           // Complete operation

                    INTERNAL STATE CHANGES: Complex multi-stage display operation…
                    SUBROUTINES: OutputToDisplayDriver, cmd_handler_10, cmd_handl…
                    switchD_01a4::cmd_multi_stage_update            XREF[2]:     ProcessData:01a4(j),
                                                                                 ProcessData:01ab(c)
01ed a6 02              LDA         #0x2

                    caseD_4a (01ef+2)                               XREF[1,1]:   ProcessData:01a4(j),
                    switchD_01a4::caseD_48                                       ProcessData:01a4(j)
01ef cd 02 38           JSR         OutputToDisplayDriver                        undefinedOutputToDisplayDriver()
                    caseD_4c (01f2+1)                               XREF[0,1]:   ProcessData:01a4(j)
01f2 cd 01 b7           JSR         switchD_01a4::cmd_handler_10                 undefinedcmd_handler_10()

                    caseD_50 (01f5+2)                               XREF[1,1]:   ProcessData:01a4(j),
                    switchD_01a4::caseD_4e                                       ProcessData:01a4(j)
01f5 cd 02 38           JSR         OutputToDisplayDriver                        undefinedOutputToDisplayDriver()
                    caseD_52 (01f8+1)                               XREF[0,1]:   ProcessData:01a4(j)
01f8 cd 02 4b           JSR         switchD_01a4::cmd_handler_a4                 undefinedcmd_handler_a4()
```

```
                    switchD_01a4::caseD_54                          XREF[1]:     ProcessData:01a4(j)
01fb 20 c9              BRA         LAB_01c6

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined cmd_handler_56()
     undefined      <UNASSIGNED>   <RETURN>
                    switchD_01a4::cmd_handler_56                    XREF[2]:     ProcessData:01a4(j),
                                                                                 ProcessData:01ad(c)
01fd a6 04              LDA         #0x4

                    switchD_01a4::caseD_58                          XREF[1]:     ProcessData:01a4(j)
01ff 20 bf              BRA         CompleteCommandProcessing                    undefinedCompleteCommandProcess...
                    -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined cmd_handler_5a()
     undefined      <UNASSIGNED>   <RETURN>
                    switchD_01a4::cmd_handler_5a                    XREF[2]:     ProcessData:01a4(j),
                                                                                 ProcessData:01af(c)
0201 a6 08              LDA         #0x8

                    switchD_01a4::caseD_5c                          XREF[1]:     ProcessData:01a4(j)
0203 20 bb              BRA         CompleteCommandProcessing                    undefinedCompleteCommandProcess...
                    -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined cmd_handler_5e()
     undefined      <UNASSIGNED>   <RETURN>
                    caseD_60 (0205+2)                               XREF[2,1]:   ProcessData:01a4(j),
                    switchD_01a4::cmd_handler_5e                                 ProcessData:01b1(c),
                                                                                 ProcessData:01a4(j)
0205 0d 15 be           BRCLR       0x6,DAT_0015,LAB_01c6                        = FFh
                    caseD_62 (0208+1)                               XREF[0,1]:   ProcessData:01a4(j)
0208 0a 14 04           BRSET       0x5,DAT_0014,switchD_01a4::caseD_68           = FFh

                    switchD_01a4::caseD_64                          XREF[1]:     ProcessData:01a4(j)
020b 1a 14              BSET        0x5,DAT_0014                                 = FFh

                    switchD_01a4::caseD_66                          XREF[1]:     ProcessData:01a4(j)
020d 20 b7              BRA         LAB_01c6

                    switchD_01a4::caseD_68                          XREF[2]:     ProcessData:01a4(j), 0208(j)
020f 1b 14              BCLR        0x5,DAT_0014                                 = FFh

                    switchD_01a4::caseD_6a                          XREF[1]:     ProcessData:01a4(j)
0211 20 b3              BRA         LAB_01c6

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined cmd_handler_6c()
     undefined      <UNASSIGNED>   <RETURN>
                    switchD_01a4::cmd_handler_6c                    XREF[3]:     ProcessData:01a4(j),
                                                                                 ProcessData:01b3(c), 021d(j)
0213 b6 00              LDA         PORTA

                    switchD_01a4::caseD_6e                          XREF[1]:     ProcessData:01a4(j)
0215 a4 3f              AND         #0x3f

                    switchD_01a4::caseD_70                          XREF[1]:     ProcessData:01a4(j)
0217 a1 21              CMP         #0x21

                    switchD_01a4::caseD_72                          XREF[1]:     ProcessData:01a4(j)
0219 27 12              BEQ         switchD_01a4::caseD_86

                    switchD_01a4::caseD_74                          XREF[1]:     ProcessData:01a4(j)
021b a1 00              CMP         #0x0

                    switchD_01a4::caseD_76                          XREF[1]:     ProcessData:01a4(j)
021d 26 f4              BNE         switchD_01a4::cmd_handler_6c

                    caseD_7a (021f+2)                               XREF[1,1]:   ProcessData:01a4(j),
                    switchD_01a4::caseD_78                                       ProcessData:01a4(j)
021f 0d 15 a4           BRCLR       0x6,DAT_0015,LAB_01c6                        = FFh
                    caseD_7c (0222+1)                               XREF[0,1]:   ProcessData:01a4(j)
0222 0c 14 04           BRSET       0x6,DAT_0014,switchD_01a4::caseD_82           = FFh
```

```
                    switchD_01a4::caseD_7e                          XREF[1]:     ProcessData:01a4(j)
0225 1c 14              BSET        0x6,DAT_0014                                    = FFh

                    switchD_01a4::caseD_80                          XREF[1]:     ProcessData:01a4(j)
0227 20 9d              BRA         LAB_01c6

                    switchD_01a4::caseD_82                          XREF[2]:     ProcessData:01a4(j), 0222(j)
0229 1d 14              BCLR        0x6,DAT_0014                                    = FFh

                    switchD_01a4::caseD_84                          XREF[1]:     ProcessData:01a4(j)
022b 20 99              BRA         LAB_01c6

                    caseD_88 (022d+2)                               XREF[2,1]:   ProcessData:01a4(j), 0219(j),
                    switchD_01a4::caseD_86                                       ProcessData:01a4(j)
022d 08 14 04           BRSET       0x4,DAT_0014,LAB_0234                           = FFh
                    caseD_8a (0230+1)                               XREF[0,1]:   ProcessData:01a4(j)
0230 18 14              BSET        0x4,DAT_0014                                    = FFh
                    caseD_8c (0232+1)                               XREF[0,1]:   ProcessData:01a4(j)
0232 20 92              BRA         LAB_01c6

                    caseD_8e (0234+1)                               XREF[1,1]:   022d(j), ProcessData:01a4(j)
                    LAB_0234
0234 19 14              BCLR        0x4,DAT_0014                                    = FFh
                    caseD_90 (0236+1)                               XREF[0,1]:   ProcessData:01a4(j)
0236 20 8e              BRA         LAB_01c6

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined OutputToDisplayDriver()
 undefined              <UNASSIGNED>   <RETURN>
                    OutputToDisplayDriver - Hardware Response Generation Protocol

                    HARDWARE INTERFACE DETAILS:
                    Response Path: 68705P3 PB0-PB7 + PC0 → External Logic → CPU ...
                    Display Path: 68705P3 PB0-PB7 + PC0 → HD44100H LCD Driver + ...

                    STROBE PROTOCOL TIMING (PC0 - Pin 2):
                    1. PORTC &= 0xFE;             // Clear PC0 strobe (setup pha...
                    2. PORTB = (data | status);     // Set 8-bit data on PB0-PB7 ...
                    3. PORTC |= 1;                 // Set PC0 strobe (data valid...
                    4. Delay loop (8192 cycles);    // Hold time for signal stabi...

                    HARDWARE TIMING REQUIREMENTS:
                    - Setup time: PC0 low before data change (step 1)
                    - Data valid time: PB0-PB7 stable before PC0 high (step 2)
                    - Strobe width: PC0 high duration (32×256 cycle delay)
                    - Hold time: Data stable after PC0 transition

                    SIGNAL INTEGRITY CONSIDERATIONS:
                    - PB0-PB7 must drive HD44100H inputs (CMOS levels)
                    - PC0 strobe must meet LCD driver setup/hold requirements
                    - Rise/fall times critical for proper data latching
                    - Current drive capability must support parallel loads

                    DUAL INTERFACE OPERATION:
                    - Same signals simultaneously drive display system AND CPU in...
                    - HD44100H: Interprets as display commands/data with PC0 enab...
                    - CPU Logic: Interprets as PANS register data with PC0 strobe
                    - Status integration: (data | DisplayControlFlags) & 0x7F pro...
                      combined response data and panel processor state information
                    caseD_92 (0238+1)                               XREF[6,1]:   CompleteCommandProcessing:01c0(c...
                    OutputToDisplayDriver                                        CompleteCommandProcessing:01c3(c...
                                                                                 cmd_display_update:01e0(c),
                                                                                 cmd_display_update:01e3(c),
                                                                                 cmd_multi_stage_update:01ef(c),
                                                                                 cmd_multi_stage_update:01f5(c),
                                                                                 ProcessData:01a4(j)
0238 ba 14              ORA         DAT_0014                                        = FFh
                    caseD_94 (023a+1)                               XREF[0,1]:   ProcessData:01a4(j)
023a a4 7f              AND         #0x7f
```

```
                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined WriteToDisplayPort()
undefined           <UNASSIGNED>  <RETURN>
                    Direct Data Output Handler - Commands 0x48-0x4A (Dispatch 0x9...

                    FUNCTION: Raw 7-bit data transmission to CPU without processi...
                    USAGE: Direct communication channel for immediate data transf...

                    OUTPUT PROTOCOL:
                    data = (input | DisplayControlFlags) & 0x7F; // Combine input...
                    PORTC = PORTC & 0xFE;             // Clear strobe (setup)
                    PORTB = data;                     // Set output data
                    PORTC = PORTC | 1;                // Set strobe (valid)
                    // Fixed timing delay for signal stability

                    DATA FORMAT:
                    Bit 7: Always 0 (masked)
                    Bits 6-0: Combined input data and status flags

                    CHARACTERISTICS:
                    - No internal state changes (pure data pass-through)
                    - Immediate response (no command processing delay)
                    - Status integration maintains communication protocol
                    - Used for real-time data transmission requirements

                    APPLICATION: Provides fast path for CPU to read panel process...
                    data without complex command processing overhead.
                    caseD_96 (023c+1)                       XREF[2,1]:   RESET:00f5(c),
                    WriteToDisplayPort                                   CompleteCommandProcessing:01c8(c...
                                                                         ProcessData:01a4(j)
023c 11 02              BCLR        0x0,PORTC                               = FFh
                    caseD_98 (023e+1)                       XREF[0,1]:   ProcessData:01a4(j)
023e b7 01              STA         PORTB                                   = FFh
                    caseD_9a (0240+1)                       XREF[0,1]:   ProcessData:01a4(j)
0240 10 02              BSET        0x0,PORTC                               = FFh
                    caseD_9c (0242+1)                       XREF[0,1]:   ProcessData:01a4(j)
0242 a6 20              LDA         #0x20

                    LAB_0244                                XREF[2]:     0245(j), 0248(j)
0244 5a                 DECX

                    switchD_01a4::caseD_9e                  XREF[1]:     ProcessData:01a4(j)
0245 26 fd              BNE         LAB_0244

                    switchD_01a4::caseD_a0                  XREF[1]:     ProcessData:01a4(j)
0247 4a                 DECA
                    caseD_a2 (0248+1)                       XREF[0,1]:   ProcessData:01a4(j)
0248 26 fa              BNE         LAB_0244
024a 81                 RTS

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined cmd_handler_a4()
undefined           <UNASSIGNED>  <RETURN>
                    switchD_01a4::cmd_handler_a4            XREF[4]:     RESET:0102(c),
                                                                         ProcessData:01a4(j),
                                                                         cmd_display_update:01e6(c),
                                                                         cmd_multi_stage_update:01f8(c)
024b a6 06              LDA         #0x6

                    switchD_01a4::caseD_a6                  XREF[1]:     ProcessData:01a4(j)
024d b7 1e              STA         DAT_001e                                = FFh

                    switchD_01a4::caseD_a8                  XREF[1]:     ProcessData:01a4(j)
024f a6 30              LDA         #0x30

                    switchD_01a4::caseD_aa                  XREF[1]:     ProcessData:01a4(j)
0251 b7 4c              STA         DAT_004c                                = FFh

                    switchD_01a4::caseD_ac                  XREF[4]:     ProcessData:01a4(j), 0254(j),
                                                                         0257(j), 025b(j)
0253 5a                 DECX
                    caseD_ae (0254+1)                       XREF[0,1]:   ProcessData:01a4(j)
0254 26 fd              BNE         switchD_01a4::caseD_ac
0256 4a                 DECA

                    switchD_01a4::caseD_b0                  XREF[1]:     ProcessData:01a4(j)
0257 26 fa              BNE         switchD_01a4::caseD_ac

                    switchD_01a4::caseD_b2                  XREF[1]:     ProcessData:01a4(j)
0259 3a 1e              DEC         DAT_001e                                = FFh
```

```
                        switchD_01a4::caseD_b4                          XREF[1]:     ProcessData:01a4(j)
025b 26 f6                  BNE         switchD_01a4::caseD_ac

                        switchD_01a4::caseD_b6                          XREF[1]:     ProcessData:01a4(j)
025d 81                     RTS

                        *************************************************************
                        *                          FUNCTION                         *
                        *************************************************************
                        undefined WaitForData()
 undefined                <UNASSIGNED>   <RETURN>
                        WaitForData - CPU Command Reception via Hardware Interface

                        HARDWARE INTERFACE DETAILS:
                        CPU Command Path: ND-120 CPU → PANC Register → CY7C401 FIFO …
                        - readIRQ() monitors FIFO status (likely external hardware si…
                        - PA0-PA2 receive 3-channel serial data (active low inputs)
                        - PC1 generates serial sampling clock (PORTC bit 1)

                        SERIAL HARDWARE PROTOCOL (192-bit packets):
                        - 8 bytes × 3 channels × 8 bits = 192 total bits received
                        - Clock generation: PORTC &= 0xFD; PORTC |= 2; (toggle PC1)
                        - Data sampling: PA0→ShiftRegister1, PA1→ShiftRegister2, PA2…
                        - Active-low inputs: if (!(PORTA & bit)) register |= 0x80;

                        TIMING COORDINATION:
                        - Synchronized with ND-120 20ms interrupt cycle
                        - CY7C401 FIFO (512×9-bit) buffers CPU microprogram commands
                        - readIRQ() polling ensures no command loss
                        - Timer coordination maintains communication protocol timing

                        DATA STORAGE:
                        - TimeDataBuffer (0x2D-0x34): Raw time data from ShiftRegiste…
                        - TimeDisplayBuffer (0x35-0x3C): Formatted display data from …
                        - StatusDataBuffer (0x3D-0x44): System status data from Shift…

                        HARDWARE REQUIREMENTS: Clean 2MHz clock, proper CMOS levels o…
                        reliable readIRQ() status signal, adequate setup/hold times f…
                        caseD_b8 (025e+1)                               XREF[1,1]:   UpdateTimersAndWait:013f(c),
                        WaitForData                                                  ProcessData:01a4(j)
025e b6 14                  LDA         DAT_0014                                       = FFh
                        caseD_ba (0260+1)                               XREF[0,1]:   ProcessData:01a4(j)
0260 b7 01                  STA         PORTB                                          = FFh
                        caseD_bc (0262+1)                               XREF[0,1]:   ProcessData:01a4(j)
0262 a6 00                  LDA         #0x0
                        caseD_be (0264+1)                               XREF[0,1]:   ProcessData:01a4(j)
0264 b7 08                  STA         Timer_Data_Reg                                 = FFh

                        caseD_c0 (0266+1)                               XREF[1,1]:   026e(j), ProcessData:01a4(j)
                        LAB_0266
0266 2f 0a                  BIH         LAB_0272
                        caseD_c2 (0268+1)                               XREF[0,1]:   ProcessData:01a4(j)
0268 b6 00                  LDA         PORTA
                        caseD_c4 (026a+1)                               XREF[0,1]:   ProcessData:01a4(j)
026a a4 3f                  AND         #0x3f
                        caseD_c6 (026c+1)                               XREF[0,1]:   ProcessData:01a4(j)
026c a1 00                  CMP         #0x0
                        caseD_c8 (026e+1)                               XREF[0,1]:   ProcessData:01a4(j)
026e 27 f6                  BEQ         LAB_0266
                        caseD_ca (0270+1)                               XREF[0,1]:   ProcessData:01a4(j)
0270 20 20                  BRA         LAB_0292

                        caseD_cc (0272+1)                               XREF[1,1]:   0266(j), ProcessData:01a4(j)
                        LAB_0272
0272 a6 00                  LDA         #0x0
                        caseD_ce (0274+1)                               XREF[0,1]:   ProcessData:01a4(j)
0274 b7 1a                  STA         DAT_001a                                       = FFh
                        caseD_d0 (0276+1)                               XREF[0,1]:   ProcessData:01a4(j)
0276 12 02                  BSET        0x1,PORTC                                      = FFh
                        caseD_d2 (0278+1)                               XREF[0,1]:   ProcessData:01a4(j)
0278 11 02                  BCLR        0x0,PORTC                                      = FFh
                        caseD_d4 (027a+1)                               XREF[0,1]:   ProcessData:01a4(j)
027a 10 02                  BSET        0x0,PORTC                                      = FFh
                        caseD_d6 (027c+1)                               XREF[0,1]:   ProcessData:01a4(j)
027c b6 4c                  LDA         DAT_004c                                       = FFh
                        caseD_d8 (027e+1)                               XREF[0,1]:   ProcessData:01a4(j)
027e b7 1e                  STA         DAT_001e                                       = FFh
                        caseD_da (0280+1)                               XREF[0,1]:   ProcessData:01a4(j)
0280 a6 05                  LDA         #0x5

                        caseD_dc (0282+1)                               XREF[3,1]:   0285(j), 0288(j), 028c(j),
                        LAB_0282                                                     ProcessData:01a4(j)
0282 2e 11                  BIL         switchD_01a4::caseD_ee
0284 5a                     DECX
```

```
                    switchD_01a4::caseD_de                          XREF[1]:    ProcessData:01a4(j)
0285 26 fb              BNE         LAB_0282

                    switchD_01a4::caseD_e0                          XREF[1]:    ProcessData:01a4(j)
0287 4a                 DECA
                    caseD_e2 (0288+1)                               XREF[0,1]:  ProcessData:01a4(j)
0288 26 f8              BNE         LAB_0282
                    caseD_e4 (028a+1)                               XREF[0,1]:  ProcessData:01a4(j)
028a 3a 1e              DEC         DAT_001e                                    = FFh
                    caseD_e6 (028c+1)                               XREF[0,1]:  ProcessData:01a4(j)
028c 26 f4              BNE         LAB_0282
                    caseD_e8 (028e+1)                               XREF[0,1]:  ProcessData:01a4(j)
028e a6 01              LDA         #0x1
                    caseD_ea (0290+1)                               XREF[0,1]:  ProcessData:01a4(j)
0290 b7 4c              STA         DAT_004c                                    = FFh

                    caseD_ec (0292+1)                               XREF[1,1]:  0270(j), ProcessData:01a4(j)
                    LAB_0292
0292 cc 01 42           JMP         ProcessData                                 undefined ProcessData()
                    -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                    switchD_01a4::caseD_ee                          XREF[2]:    ProcessData:01a4(j), 0282(j)
0295 ae 08              LDX         #0x8

                    switchD_01a4::caseD_f0                          XREF[3]:    ProcessData:01a4(j), 02c6(j),
                                                                                02dc(j)
0297 b6 00              LDA         PORTA

                    switchD_01a4::caseD_f2                          XREF[1]:    ProcessData:01a4(j)
0299 13 02              BCLR        0x1,PORTC                                   = FFh

                    switchD_01a4::caseD_f4                          XREF[1]:    ProcessData:01a4(j)
029b 12 02              BSET        0x1,PORTC                                   = FFh

                    switchD_01a4::caseD_f6                          XREF[1]:    ProcessData:01a4(j)
029d b7 20              STA         DAT_0020                                    = FFh

                    switchD_01a4::caseD_f8                          XREF[1]:    ProcessData:01a4(j)
029f 34 21              LSR         DAT_0021                                    = FFh

                    switchD_01a4::caseD_fa                          XREF[1]:    ProcessData:01a4(j)
02a1 34 22              LSR         DAT_0022                                    = FFh

                    switchD_01a4::caseD_fc                          XREF[1]:    ProcessData:01a4(j)
02a3 34 23              LSR         DAT_0023                                    = FFh

                    switchD_01a4::caseD_fe                          XREF[1]:    ProcessData:01a4(j)
02a5 00 20 04           BRSET       0x0,DAT_0020,LAB_02ac                       = FFh
02a8 1e 21              BSET        0x7,DAT_0021                                = FFh
02aa 20 02              BRA         LAB_02ae

                    LAB_02ac                                        XREF[1]:    02a5(j)
02ac 1f 21              BCLR        0x7,DAT_0021                                = FFh

                    LAB_02ae                                        XREF[1]:    02aa(j)
02ae 02 20 04           BRSET       0x1,DAT_0020,LAB_02b5                       = FFh
02b1 1e 22              BSET        0x7,DAT_0022                                = FFh
02b3 20 02              BRA         LAB_02b7

                    LAB_02b5                                        XREF[1]:    02ae(j)
02b5 1f 22              BCLR        0x7,DAT_0022                                = FFh

                    LAB_02b7                                        XREF[1]:    02b3(j)
02b7 04 20 04           BRSET       0x2,DAT_0020,LAB_02be                       = FFh
02ba 1e 23              BSET        0x7,DAT_0023                                = FFh
02bc 20 02              BRA         LAB_02c0

                    LAB_02be                                        XREF[1]:    02b7(j)
02be 1f 23              BCLR        0x7,DAT_0023                                = FFh

                    LAB_02c0                                        XREF[1]:    02bc(j)
02c0 3c 1a              INC         DAT_001a                                    = FFh
02c2 b6 1a              LDA         DAT_001a                                    = FFh
02c4 a1 08              CMP         #0x8
02c6 26 cf              BNE         switchD_01a4::caseD_f0
02c8 a6 00              LDA         #0x0
02ca b7 1a              STA         DAT_001a                                    = FFh
02cc 5a                 DECX
02cd b6 21              LDA         DAT_0021                                    = FFh
02cf e7 2d              STA         0x2d,X=>time_data_buffer_7                  = FFh
02d1 b6 22              LDA         DAT_0022                                    = FFh
02d3 e7 35              STA         0x35,X=>time_display_buffer_7               = FFh
02d5 b6 23              LDA         DAT_0023                                    = FFh
02d7 e7 3d              STA         0x3d,X=>status_data_buffer_7                = FFh
```

```
02d9 9f             TXA
02da a1 00          CMP         #0x0
02dc 26 b9          BNE         switchD_01a4::caseD_f0
                Serial Data Reception Handler - Commands 0x77-0x7F (Dispatch ...

                FUNCTION: Processes 192-bit serial data packets from CPU
                DATA STRUCTURE: 8 bytes × 3 channels × 8 bits = 192 bits total

                RECEPTION PROTOCOL:
                for (byte_count = 8; byte_count > 0; byte_count--) {
                    for (bit_count = 8; bit_count > 0; bit_count--) {
                        PORTC &= 0xFD;              // Clear clock
                        PORTC |= 2;                 // Set clock
                        SerialInputData = PORTA;    // Read input

                        // Shift data into 3 parallel registers (active-low i...
                        ShiftRegister1 >>= 1; if (!(PORTA & 1)) ShiftRegister...
                        ShiftRegister2 >>= 1; if (!(PORTA & 2)) ShiftRegister...
                        ShiftRegister3 >>= 1; if (!(PORTA & 4)) ShiftRegister...
                    }

                    // Store completed bytes in data buffers
                    TimeDataBuffer[byte_count-1] = ShiftRegister1;    // Raw ...
                    TimeDisplayBuffer[byte_count-1] = ShiftRegister2; // Disp...
                    StatusDataBuffer[byte_count-1] = ShiftRegister3;  // Stat...
                }

                POST-PROCESSING: Calls DecodeCharacterFromTable() and display...
                to process received data for panel display updates.
02de b6 19          LDA         DAT_0019                                = FFh
02e0 b1 18          CMP         DAT_0018                                = FFh
02e2 27 09          BEQ         LAB_02ed
02e4 b7 18          STA         DAT_0018                                = FFh
02e6 cd 06 18       JSR         InitDisplayClearPulse                   undefinedInitDisplayClearPulse()
02e9 a6 00          LDA         #0x0
02eb b7 47          STA         DAT_0047                                = FFh

                LAB_02ed                                    XREF[1]:    02e2(j)
02ed a6 00          LDA         #0x0
02ef b7 19          STA         DAT_0019                                = FFh
02f1 ae 00          LDX         #0x0
02f3 e6 2d          LDA         TimeDataBuffer0,X                       = FFh
02f5 b7 1b          STA         DAT_001b                                = FFh
02f7 5c             INCX
02f8 e6 2d          LDA         0x2d,X=>TimeDataBuffer1                 = FFh
02fa b7 1c          STA         DAT_001c                                = FFh
02fc cd 06 41       JSR         DecodeCharacterFromTable                undefinedDecodeCharacterFromTab...
02ff b7 1d          STA         DAT_001d                                = FFh
0301 a1 00          CMP         #0x0
0303 26 03          BNE         LAB_0308
0305 cc 04 1d       JMP         ShowSystemStatusDisplay                 undefinedShowSystemStatusDispla...
                -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                LAB_0308                                    XREF[1]:    0303(j)
0308 ae 01          LDX         #0x1
030a b6 1d          LDA         DAT_001d                                = FFh
030c bf 1f          STX         DAT_001f                                = FFh

                LAB_030e                                    XREF[1]:    0325(j)
030e 54             LSRX
030f e7 25          STA         MessageBuffer0,X                        = FFh
0311 be 1f          LDX         DAT_001f                                = FFh
0313 5c             INCX
0314 e6 2d          LDA         0x2d,X=>time_data_buffer_2              = FFh
0316 b7 1b          STA         DAT_001b                                = FFh
0318 5c             INCX
0319 9f             TXA
031a a1 09          CMP         #0x9
031c 27 09          BEQ         LAB_0327
031e e6 2d          LDA         0x2d,X=>time_data_buffer_3              = FFh
0320 b7 1c          STA         DAT_001c                                = FFh
0322 cd 06 41       JSR         DecodeCharacterFromTable                undefinedDecodeCharacterFromTab...
0325 20 e7          BRA         LAB_030e

                LAB_0327                                    XREF[1]:    031c(j)
0327 0a 18 1f       BRSET       0x5,DAT_0018,LAB_0349                   = FFh
032a ae 03          LDX         #0x3

                LAB_032c                                    XREF[1]:    0335(j)
032c e6 25          LDA         0x25,X=>MessageBuffer3                  = FFh
032e e7 29          STA         0x29,X=>StoredMessageBuffer3            = FFh
0330 a1 20          CMP         #0x20
0332 26 6e          BNE         LAB_03a2
0334 5a             DECX
0335 2a f5          BPL         LAB_032c
```

```
0337 a6 00          LDA       #0x0
0339 b7 47          STA       DAT_0047                                = FFh
033b a6 60          LDA       #0x60
033d b7 45          STA       DAT_0045                                = FFh
033f 1a 19          BSET      0x5,DAT_0019                            = FFh
0341 0e 3a 02       BRSET     0x7,DAT_003a,LAB_0346                   = FFh
0344 20 65          BRA       LAB_03ab

                    LAB_0346                                          XREF[3]:    0341(j), 0356(j), 039d(j)
0346 cc 04 d6       JMP       LAB_04d6

                    LAB_0349                                          XREF[1]:    0327(j)
0349 ae 03          LDX       #0x3
034b 1a 19          BSET      0x5,DAT_0019                            = FFh

                    LAB_034d                                          XREF[1]:    0354(j)
034d e6 25          LDA       0x25,X=>MessageBuffer3                  = FFh
034f e1 29          CMP       0x29,X=>StoredMessageBuffer3            = FFh
0351 26 08          BNE       LAB_035b
0353 5a             DECX
0354 2a f7          BPL       LAB_034d
0356 0e 3a ed       BRSET     0x7,DAT_003a,LAB_0346                   = FFh
0359 20 50          BRA       LAB_03ab

                    LAB_035b                                          XREF[1]:    0351(j)
035b ae 01          LDX       #0x1
035d b6 25          LDA       MessageBuffer0                          = FFh

                    LAB_035f                                          XREF[1]:    0367(j)
035f e1 29          CMP       0x29,X=>StoredMessageBuffer1            = FFh
0361 27 06          BEQ       LAB_0369
0363 5c             INCX
0364 9f             TXA
0365 a1 04          CMP       #0x4
0367 26 f6          BNE       LAB_035f

                    LAB_0369                                          XREF[1]:    0361(j)
0369 bf 1f          STX       DAT_001f                                = FFh
036b a6 04          LDA       #0x4
036d b0 1f          SUB       DAT_001f                                = FFh
036f 97             TAX

                    LAB_0370                                          XREF[1]:    038f(j)
0370 b6 47          LDA       DAT_0047                                = FFh
0372 a1 28          CMP       #0x28
0374 26 04          BNE       LAB_037a
0376 a6 00          LDA       #0x0
0378 b7 47          STA       DAT_0047                                = FFh

                    LAB_037a                                          XREF[1]:    0374(j)
037a a6 18          LDA       #0x18
037c cd 05 fc       JSR       SendDisplayCommand                      undefinedSendDisplayCommand()
037f 3c 47          INC       DAT_0047                                = FFh
0381 a6 27          LDA       #0x27
0383 cd 06 90       JSR       CalculateDisplayPosition                undefinedCalculateDisplayPositi...
0386 e6 25          LDA       0x25,X=>MessageBuffer2                  = FFh
0388 cd 05 d2       JSR       OutputCharacterToDisplay                undefinedOutputCharacterToDispl...
038b 5c             INCX
038c 9f             TXA
038d a1 04          CMP       #0x4
038f 26 df          BNE       LAB_0370
0391 ae 00          LDX       #0x0

                    LAB_0393                                          XREF[1]:    039b(j)
0393 e6 25          LDA       MessageBuffer0,X                        = FFh
0395 e7 29          STA       StoredMessageBuffer0,X                  = FFh
0397 5c             INCX
0398 9f             TXA
0399 a1 04          CMP       #0x4
039b 26 f6          BNE       LAB_0393
039d 0e 3a a6       BRSET     0x7,DAT_003a,LAB_0346                   = FFh
03a0 20 09          BRA       LAB_03ab

                    LAB_03a2                                          XREF[1]:    0332(j)
03a2 a6 00          LDA       #0x0
03a4 b7 45          STA       DAT_0045                                = FFh
03a6 a6 40          LDA       #0x40
03a8 cd 06 90       JSR       CalculateDisplayPosition                undefinedCalculateDisplayPositi...

                    LAB_03ab                                          XREF[3]:    0344(j), 0359(j), 03a0(j)
03ab b6 45          LDA       DAT_0045                                = FFh
03ad cd 06 90       JSR       CalculateDisplayPosition                undefinedCalculateDisplayPositi...
03b0 12 19          BSET      0x1,DAT_0019                            = FFh
03b2 0a 19 12       BRSET     0x5,DAT_0019,LAB_03c7                   = FFh
03b5 b6 25          LDA       MessageBuffer0                          = FFh
```

```
03b7 a1 20          CMP       #0x20
03b9 27 07          BEQ       LAB_03c2
03bb ae b5          LDX       #0xb5
03bd cd 06 0c       JSR       DisplayStringUntilQuote                undefinedDisplayStringUntilQuot...
03c0 20 05          BRA       LAB_03c7

                    LAB_03c2                                         XREF[1]:    03b9(j)
03c2 ae be          LDX       #0xbe
03c4 cd 06 0c       JSR       DisplayStringUntilQuote                undefinedDisplayStringUntilQuot...

                    LAB_03c7                                         XREF[2]:    03b2(j), 03c0(j)
03c7 a6 35          LDA       #0x35
03c9 97             TAX
03ca cd 03 cf       JSR       DisplayTimeData                        undefinedDisplayTimeData()
03cd 20 25          BRA       ShowMessageAndTime                     undefinedShowMessageAndTime()
                    -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined DisplayTimeData()
 undefined          <UNASSIGNED>  <RETURN>
                    DisplayTimeData                                  XREF[2]:    WaitForData:03ca(c),
                                                                                 ShowMessageAndTime:0417(c)
03cf bf 1f          STX       DAT_001f                                 = FFh
03d1 a6 09          LDA       #0x9
03d3 b7 1e          STA       DAT_001e                                 = FFh
03d5 a6 01          LDA       #0x1
03d7 b7 24          STA       DAT_0024                                 = FFh

                    LAB_03d9                                         XREF[2]:    03ed(j), 03f2(j)
03d9 3a 1e          DEC       DAT_001e                                 = FFh
03db 26 01          BNE       LAB_03de
03dd 81             RTS

                    LAB_03de                                         XREF[1]:    03db(j)
03de be 1f          LDX       DAT_001f                                 = FFh
03e0 f6             LDA       X
03e1 cd 06 23       JSR       LookupCharacterCode                    undefinedLookupCharacterCode()
03e4 a1 ff          CMP       #0xff
03e6 26 07          BNE       LAB_03ef
03e8 cd 05 97       JSR       DisplayBinaryDigits                    undefinedDisplayBinaryDigits()
03eb 14 19          BSET      0x2,DAT_0019                             = FFh
03ed 20 ea          BRA       LAB_03d9

                    LAB_03ef                                         XREF[1]:    03e6(j)
03ef cd 05 d2       JSR       OutputCharacterToDisplay               undefinedOutputCharacterToDispl...
03f2 20 e5          BRA       LAB_03d9

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined ShowMessageAndTime()
 undefined          <UNASSIGNED>  <RETURN>
                    ShowMessageAndTime                               XREF[1]:    WaitForData:03cd(c)
03f4 a6 40          LDA       #0x40
03f6 cd 06 90       JSR       CalculateDisplayPosition               undefinedCalculateDisplayPositi...
03f9 0a 19 19       BRSET     0x5,DAT_0019,LAB_0415                    = FFh
03fc ae 00          LDX       #0x0

                    LAB_03fe                                         XREF[1]:    0407(j)
03fe e6 25          LDA       MessageBuffer0,X                         = FFh
0400 cd 05 d2       JSR       OutputCharacterToDisplay               undefinedOutputCharacterToDispl...
0403 5c             INCX
0404 9f             TXA
0405 a1 04          CMP       #0x4
0407 26 f5          BNE       LAB_03fe
0409 a6 40          LDA       #0x40
040b ab 07          ADD       #0x7
040d cd 06 90       JSR       CalculateDisplayPosition               undefinedCalculateDisplayPositi...
0410 a6 3a          LDA       #0x3a
0412 cd 05 d2       JSR       OutputCharacterToDisplay               undefinedOutputCharacterToDispl...

                    LAB_0415                                         XREF[1]:    03f9(j)
0415 ae 3d          LDX       #0x3d
0417 cd 03 cf       JSR       DisplayTimeData                        undefinedDisplayTimeData()
041a cc 05 79       JMP       LAB_0579

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined ShowSystemStatusDisplay()
 undefined          <UNASSIGNED>  <RETURN>
                    ShowSystemStatusDisplay                          XREF[1]:    WaitForData:0305(c)
041d a6 01          LDA       #0x1
```

```
041f b7 19          STA     DAT_0019                        = FFh
0421 a6 00          LDA     #0x0
0423 b7 1e          STA     DAT_001e                        = FFh
0425 a6 00          LDA     #0x0
0427 cd 06 90       JSR     CalculateDisplayPosition        undefinedCalculateDisplayPositi...

                    LAB_042a                                XREF[1]:    0485(j)
042a bf 1f          STX     DAT_001f                        = FFh
042c a6 5f          LDA     #0x5f
042e 03 1c 02       BRCLR   0x1,DAT_001c,LAB_0433           = FFh
0431 a6 db          LDA     #0xdb

                    LAB_0433                                XREF[1]:    042e(j)
0433 cd 05 d2       JSR     OutputCharacterToDisplay        undefinedOutputCharacterToDispl...
0436 01 1b 02       BRCLR   0x0,DAT_001b,LAB_043b           = FFh
0439 a6 db          LDA     #0xdb

                    LAB_043b                                XREF[1]:    0436(j)
043b cd 05 d2       JSR     OutputCharacterToDisplay        undefinedOutputCharacterToDispl...
043e 05 1b 02       BRCLR   0x2,DAT_001b,LAB_0443           = FFh
0441 a6 db          LDA     #0xdb

                    LAB_0443                                XREF[1]:    043e(j)
0443 cd 05 d2       JSR     OutputCharacterToDisplay        undefinedOutputCharacterToDispl...
0446 09 1b 02       BRCLR   0x4,DAT_001b,LAB_044b           = FFh
0449 a6 db          LDA     #0xdb

                    LAB_044b                                XREF[1]:    0446(j)
044b cd 05 d2       JSR     OutputCharacterToDisplay        undefinedOutputCharacterToDispl...
044e 0d 1b 02       BRCLR   0x6,DAT_001b,LAB_0453           = FFh
0451 a6 db          LDA     #0xdb

                    LAB_0453                                XREF[1]:    044e(j)
0453 cd 05 d2       JSR     OutputCharacterToDisplay        undefinedOutputCharacterToDispl...
0456 0f 1c 02       BRCLR   0x7,DAT_001c,LAB_045b           = FFh
0459 a6 db          LDA     #0xdb

                    LAB_045b                                XREF[1]:    0456(j)
045b cd 05 d2       JSR     OutputCharacterToDisplay        undefinedOutputCharacterToDispl...
045e 0b 1c 02       BRCLR   0x5,DAT_001c,LAB_0463           = FFh
0461 a6 db          LDA     #0xdb

                    LAB_0463                                XREF[1]:    045e(j)
0463 cd 05 d2       JSR     OutputCharacterToDisplay        undefinedOutputCharacterToDispl...
0466 09 1c 02       BRCLR   0x4,DAT_001c,LAB_046b           = FFh
0469 a6 db          LDA     #0xdb

                    LAB_046b                                XREF[1]:    0466(j)
046b cd 05 d2       JSR     OutputCharacterToDisplay        undefinedOutputCharacterToDispl...
046e be 1f          LDX     DAT_001f                        = FFh
0470 5c             INCX
0471 e6 2d          LDA     0x2d,X
0473 b7 1b          STA     DAT_001b                        = FFh
0475 5c             INCX
0476 e6 2d          LDA     0x2d,X
0478 b7 1c          STA     DAT_001c                        = FFh
047a 3c 1e          INC     DAT_001e                        = FFh
047c a6 0c          LDA     #0xc
047e cd 06 90       JSR     CalculateDisplayPosition        undefinedCalculateDisplayPositi...
0481 b6 1e          LDA     DAT_001e                        = FFh
0483 a1 01          CMP     #0x1
0485 23 a3          BLS     LAB_042a
0487 a6 18          LDA     #0x18
0489 cd 06 90       JSR     CalculateDisplayPosition        undefinedCalculateDisplayPositi...
048c a6 5f          LDA     #0x5f
048e 01 1c 02       BRCLR   0x0,DAT_001c,LAB_0493           = FFh
0491 a6 30          LDA     #0x30

                    LAB_0493                                XREF[1]:    048e(j)
0493 cd 05 d2       JSR     OutputCharacterToDisplay        undefinedOutputCharacterToDispl...
0496 03 1b 02       BRCLR   0x1,DAT_001b,LAB_049b           = FFh
0499 a6 31          LDA     #0x31

                    LAB_049b                                XREF[1]:    0496(j)
049b cd 05 d2       JSR     OutputCharacterToDisplay        undefinedOutputCharacterToDispl...
049e 0f 1b 02       BRCLR   0x7,DAT_001b,LAB_04a3           = FFh
04a1 a6 32          LDA     #0x32

                    LAB_04a3                                XREF[1]:    049e(j)
04a3 cd 05 d2       JSR     OutputCharacterToDisplay        undefinedOutputCharacterToDispl...
04a6 0d 1c 02       BRCLR   0x6,DAT_001c,LAB_04ab           = FFh
04a9 a6 33          LDA     #0x33

                    LAB_04ab                                XREF[1]:    04a6(j)
04ab cd 05 d2       JSR     OutputCharacterToDisplay        undefinedOutputCharacterToDispl...
```

```
04ae a6 1f          LDA       #0x1f
04b0 cd 06 90       JSR       CalculateDisplayPosition                undefinedCalculateDisplayPositi...
04b3 5c             INCX
04b4 e6 2d          LDA       0x2d,X
04b6 b7 1b          STA       DAT_001b                                = FFh
04b8 5c             INCX
04b9 e6 2d          LDA       0x2d,X
04bb b7 1c          STA       DAT_001c                                = FFh
04bd ae 9c          LDX       #0x9c
04bf 0f 1b 02       BRCLR     0x7,DAT_001b,LAB_04c4                   = FFh
04c2 ae 98          LDX       #0x98

                    LAB_04c4                                          XREF[1]:    04bf(j)
04c4 cd 06 0c       JSR       DisplayStringUntilQuote                 undefinedDisplayStringUntilQuot...
04c7 a6 24          LDA       #0x24
04c9 cd 06 90       JSR       CalculateDisplayPosition                undefinedCalculateDisplayPositi...
04cc ae 9c          LDX       #0x9c
04ce 09 1b 02       BRCLR     0x4,DAT_001b,LAB_04d3                   = FFh
04d1 ae 98          LDX       #0x98

                    LAB_04d3                                          XREF[1]:    04ce(j)
04d3 cd 06 0c       JSR       DisplayStringUntilQuote                 undefinedDisplayStringUntilQuot...

                    LAB_04d6                                          XREF[1]:    WaitForData:0346(j)
04d6 a6 40          LDA       #0x40
04d8 cd 06 90       JSR       CalculateDisplayPosition                undefinedCalculateDisplayPositi...
04db 0e 38 04       BRSET     0x7,DAT_0038,LAB_04e2                   = FFh
04de ae c7          LDX       #0xc7
04e0 20 02          BRA       LAB_04e4

                    LAB_04e2                                          XREF[1]:    04db(j)
04e2 ae a0          LDX       #0xa0

                    LAB_04e4                                          XREF[1]:    04e0(j)
04e4 cd 06 0c       JSR       DisplayStringUntilQuote                 undefinedDisplayStringUntilQuot...
04e7 ae 00          LDX       #0x0
04e9 bf 1f          STX       DAT_001f                                = FFh
04eb e6 35          LDA       TimeDisplayBuffer0,X                    = FFh
04ed cd 06 23       JSR       LookupCharacterCode                     undefinedLookupCharacterCode()
04f0 cd 05 d2       JSR       OutputCharacterToDisplay                undefinedOutputCharacterToDispl...
04f3 be 1f          LDX       DAT_001f                                = FFh
04f5 e6 35          LDA       TimeDisplayBuffer0,X                    = FFh
04f7 cd 06 23       JSR       LookupCharacterCode                     undefinedLookupCharacterCode()
04fa cd 05 d2       JSR       OutputCharacterToDisplay                undefinedOutputCharacterToDispl...
04fd 0f 38 0c       BRCLR     0x7,DAT_0038,LAB_050c                   = FFh
0500 ae a5          LDX       #0xa5
0502 0e 3c 02       BRSET     0x7,time_display_buffer_7,LAB_0507      = FFh
0505 ae ad          LDX       #0xad

                    LAB_0507                                          XREF[1]:    0502(j)
0507 18 19          BSET      0x4,DAT_0019                            = FFh
0509 cd 06 0c       JSR       DisplayStringUntilQuote                 undefinedDisplayStringUntilQuot...

                    LAB_050c                                          XREF[1]:    04fd(j)
050c be 1f          LDX       DAT_001f                                = FFh
050e e6 35          LDA       TimeDisplayBuffer0,X                    = FFh
0510 cd 06 23       JSR       LookupCharacterCode                     undefinedLookupCharacterCode()
0513 cd 05 d2       JSR       OutputCharacterToDisplay                undefinedOutputCharacterToDispl...
0516 be 1f          LDX       DAT_001f                                = FFh
0518 e6 35          LDA       TimeDisplayBuffer0,X                    = FFh
051a cd 06 23       JSR       LookupCharacterCode                     undefinedLookupCharacterCode()
051d cd 05 d2       JSR       OutputCharacterToDisplay                undefinedOutputCharacterToDispl...
0520 0e 38 0b       BRSET     0x7,DAT_0038,LAB_052e                   = FFh
0523 ae cd          LDX       #0xcd
0525 cd 06 0c       JSR       DisplayStringUntilQuote                 undefinedDisplayStringUntilQuot...
0528 be 1f          LDX       DAT_001f                                = FFh
052a 5c             INCX
052b 5c             INCX
052c 20 20          BRA       LAB_054e

                    LAB_052e                                          XREF[1]:    0520(j)
052e a6 3a          LDA       #0x3a
0530 cd 05 d2       JSR       OutputCharacterToDisplay                undefinedOutputCharacterToDispl...
0533 be 1f          LDX       DAT_001f                                = FFh
0535 e6 35          LDA       TimeDisplayBuffer0,X                    = FFh
0537 cd 06 23       JSR       LookupCharacterCode                     undefinedLookupCharacterCode()
053a cd 05 d2       JSR       OutputCharacterToDisplay                undefinedOutputCharacterToDispl...
053d be 1f          LDX       DAT_001f                                = FFh
053f e6 35          LDA       TimeDisplayBuffer0,X                    = FFh
0541 cd 06 23       JSR       LookupCharacterCode                     undefinedLookupCharacterCode()
0544 cd 05 d2       JSR       OutputCharacterToDisplay                undefinedOutputCharacterToDispl...
0547 a6 3a          LDA       #0x3a
0549 cd 05 d2       JSR       OutputCharacterToDisplay                undefinedOutputCharacterToDispl...
054c be 1f          LDX       DAT_001f                                = FFh
```

```
                LAB_054e                                XREF[1]:    052c(j)
054e e6 35          LDA         0x35,X=>time_display_buffer_2       = FFh
0550 cd 06 23       JSR         LookupCharacterCode                 undefined LookupCharacterCode()
0553 cd 05 d2       JSR         OutputCharacterToDisplay            undefined OutputCharacterToDispl...
0556 be 1f          LDX         DAT_001f                            = FFh
0558 e6 35          LDA         TimeDisplayBuffer0,X                = FFh
055a cd 06 23       JSR         LookupCharacterCode                 undefined LookupCharacterCode()
055d cd 05 d2       JSR         OutputCharacterToDisplay            undefined OutputCharacterToDispl...
0560 a6 57          LDA         #0x57
0562 cd 06 90       JSR         CalculateDisplayPosition            undefined CalculateDisplayPositi...
0565 a6 20          LDA         #0x20
0567 cd 05 d2       JSR         OutputCharacterToDisplay            undefined OutputCharacterToDispl...
056a ae 00          LDX         #0x0

                LAB_056c                                XREF[1]:    0577(j)
056c e6 3d          LDA         StatusDataBuffer0,X                 = FFh
056e b7 1b          STA         DAT_001b                            = FFh
0570 cd 05 7c       JSR         DisplayBinaryBars                   undefined DisplayBinaryBars()
0573 5c             INCX
0574 9f             TXA
0575 a1 08          CMP         #0x8
0577 25 f3          BCS         LAB_056c

                LAB_0579                                XREF[1]:    ShowMessageAndTime:041a(j)
0579 cc 01 42       JMP         ProcessData                         undefined ProcessData()
                -- Flow Override: CALL_RETURN (CALL_TERMINATOR)

                *************************************************************
                *                          FUNCTION                         *
                *************************************************************
                undefined DisplayBinaryBars()
 undefined          <UNASSIGNED>  <RETURN>
                DisplayBinaryBars                       XREF[1]:    ShowSystemStatusDisplay:057(c)
057c 04 1b 07       BRSET       0x2,DAT_001b,LAB_0586               = FFh
057f a6 2e          LDA         #0x2e

                LAB_0581                                XREF[1]:    0588(j)
0581 cd 05 d2       JSR         OutputCharacterToDisplay            undefined OutputCharacterToDispl...
0584 20 04          BRA         LAB_058a

                LAB_0586                                XREF[1]:    057c(j)
0586 a6 7c          LDA         #0x7c
0588 20 f7          BRA         LAB_0581

                LAB_058a                                XREF[1]:    0584(j)
058a 00 1b 06       BRSET       0x0,DAT_001b,LAB_0593               = FFh
058d a6 2e          LDA         #0x2e

                LAB_058f                                XREF[1]:    0595(j)
058f cd 05 d2       JSR         OutputCharacterToDisplay            undefined OutputCharacterToDispl...
0592 81             RTS

                LAB_0593                                XREF[1]:    058a(j)
0593 a6 7c          LDA         #0x7c
0595 20 f8          BRA         LAB_058f

                *************************************************************
                *                          FUNCTION                         *
                *************************************************************
                undefined DisplayBinaryDigits()
 undefined          <UNASSIGNED>  <RETURN>
                DisplayBinaryDigits                     XREF[1]:    DisplayTimeData:03e8(c)
0597 3c 24          INC         DAT_0024                            = FFh
0599 b6 24          LDA         DAT_0024                            = FFh
059b a1 03          CMP         #0x3
059d 26 03          BNE         LAB_05a2
059f cd 05 c8       JSR         DisplayDecimalPoint                 undefined DisplayDecimalPoint()

                LAB_05a2                                XREF[1]:    059d(j)
05a2 04 1b 07       BRSET       0x2,DAT_001b,LAB_05ac               = FFh
05a5 a6 30          LDA         #0x30

                LAB_05a7                                XREF[1]:    05ae(j)
05a7 cd 05 d2       JSR         OutputCharacterToDisplay            undefined OutputCharacterToDispl...
05aa 20 04          BRA         LAB_05b0

                LAB_05ac                                XREF[1]:    05a2(j)
05ac a6 31          LDA         #0x31
05ae 20 f7          BRA         LAB_05a7

                LAB_05b0                                XREF[1]:    05aa(j)
05b0 3c 24          INC         DAT_0024                            = FFh
05b2 b6 24          LDA         DAT_0024                            = FFh
05b4 a1 03          CMP         #0x3
```

```
05b6 26 03          BNE       LAB_05bb
05b8 cd 05 c8       JSR       DisplayDecimalPoint                    undefinedDisplayDecimalPoint()

                    LAB_05bb                                         XREF[1]:    05b6(j)
05bb 00 1b 06       BRSET     0x0,DAT_001b,LAB_05c4                    = FFh
05be a6 30          LDA       #0x30

                    LAB_05c0                                         XREF[1]:    05c6(j)
05c0 cd 05 d2       JSR       OutputCharacterToDisplay               undefinedOutputCharacterToDispl...
05c3 81             RTS

                    LAB_05c4                                         XREF[1]:    05bb(j)
05c4 a6 31          LDA       #0x31
05c6 20 f8          BRA       LAB_05c0

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined DisplayDecimalPoint()
 undefined          <UNASSIGNED>  <RETURN>
                    DisplayDecimalPoint                              XREF[2]:    DisplayBinaryDigits:059f(c),
                                                                                 DisplayBinaryDigits:05b8(c)
05c8 a6 2e          LDA       #0x2e
05ca cd 05 d2       JSR       OutputCharacterToDisplay               undefinedOutputCharacterToDispl...
05cd a6 00          LDA       #0x0
05cf b7 24          STA       DAT_0024                                 = FFh
05d1 81             RTS

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined OutputCharacterToDisplay()
 undefined          <UNASSIGNED>  <RETURN>
                    OutputCharacterToDisplay                         XREF[33]:   WaitForData:0388(c),
                                                                                 ShowMessageAndTime:0400(c),
                                                                                 ShowMessageAndTime:0412(c),
                                                                                 ShowSystemStatusDisplay:0433(c),
                                                                                 ShowSystemStatusDisplay:043b(c),
                                                                                 ShowSystemStatusDisplay:0443(c),
                                                                                 ShowSystemStatusDisplay:044b(c),
                                                                                 ShowSystemStatusDisplay:0453(c),
                                                                                 ShowSystemStatusDisplay:045b(c),
                                                                                 ShowSystemStatusDisplay:0463(c),
                                                                                 ShowSystemStatusDisplay:046b(c),
                                                                                 ShowSystemStatusDisplay:0493(c),
                                                                                 ShowSystemStatusDisplay:049b(c),
                                                                                 ShowSystemStatusDisplay:04a3(c),
                                                                                 ShowSystemStatusDisplay:04ab(c),
                                                                                 ShowSystemStatusDisplay:04f0(c),
                                                                                 ShowSystemStatusDisplay:04fa(c),
                                                                                 ShowSystemStatusDisplay:0513(c),
                                                                                 ShowSystemStatusDisplay:051d(c),
                                                                                 ShowSystemStatusDisplay:0530(c),
                                                                                 [more]
05d2 b7 49          STA       DAT_0049                                 = FFh
05d4 b6 48          LDA       DAT_0048                                 = FFh
05d6 a1 28          CMP       #0x28
05d8 26 07          BNE       LAB_05e1
05da a6 00          LDA       #0x0
05dc cd 06 b0       JSR       SetDisplayAddressAndWrite              undefinedSetDisplayAddressAndWr...
05df 20 09          BRA       LAB_05ea

                    LAB_05e1                                         XREF[1]:    05d8(j)
05e1 a1 68          CMP       #0x68
05e3 26 05          BNE       LAB_05ea
05e5 a6 40          LDA       #0x40
05e7 cd 06 b0       JSR       SetDisplayAddressAndWrite              undefinedSetDisplayAddressAndWr...

                    LAB_05ea                                         XREF[2]:    05df(j), 05e3(j)
05ea b6 49          LDA       DAT_0049                                 = FFh
05ec b7 01          STA       PORTB                                    = FFh
05ee 16 02          BSET      0x3,PORTC                                = FFh
05f0 17 02          BCLR      0x3,PORTC                                = FFh
05f2 3c 48          INC       DAT_0048                                 = FFh
05f4 a6 05          LDA       #0x5

                    LAB_05f6                                         XREF[1]:    05f7(j)
05f6 4a             DECA
05f7 26 fd          BNE       LAB_05f6
05f9 a6 5f          LDA       #0x5f
05fb 81             RTS
```

```
                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined SendDisplayCommand()
undefined           <UNASSIGNED>  <RETURN>
                    SendDisplayCommand - HD44100H LCD Driver Hardware Control

                    HARDWARE: HD44100H 40-Channel LCD Driver with Serial/Parallel...
                    DISPLAY MODULES: LD-H7919 (8-digit alphanumeric) + LCD SX 423...

                    HD44100H INTERFACE PROTOCOL:
                    - PB0-PB7 (Pins 6-13): 8-bit parallel data/command input
                    - PC2 (Pin 3): Command/Data mode select (1=command, 0=data)
                    - PC0 (Pin 2): Enable/Strobe signal (data valid)

                    COMMAND MODE SEQUENCE:
                    1. PORTB = command;            // Set command byte on parall...
                    2. PORTC &= 0xF3;              // Clear PC3-PC2 (mode control)
                    3. PORTC |= 4;                 // Set PC2 = command mode
                    4. Delay (5 cycles);           // Allow setup time
                    5. (PC0 strobe handled by caller)

                    DISPLAY SYSTEM ARCHITECTURE:
                    HD44100H → 40 LCD segment drivers → Multiple LCD modules
                    PB0-PB7 → HD44100H Data Input (8-bit parallel)
                    PC2 → HD44100H Command/Data Select
                    PC0 → HD44100H Enable (from strobe protocol)
                    PC3-PC7 → Additional display control (module select, etc.)

                    PARALLEL DISPLAY CHAIN:
                    HD44100H also connects to CD4035 shift register chain:
                    - Same PB0-PB7 data bus shared between HD44100H and CD4035 in...
                    - PC0 strobe latches data into both HD44100H and CD4035 simul...
                    - PC1 provides shift clock for CD4035 registers
                    - This enables coordinated updates across entire display syst...

                    TIMING REQUIREMENTS: HD44100H requires specific setup/hold ti...
                    for command/data transitions and enable pulse widths for reli...
                    SendDisplayCommand                              XREF[6]:     RESET:0107(c), RESET:010c(c),
                                                                                 RESET:0111(c),
                                                                                 WaitForData:037c(c),
                                                                                 InitDisplayClearPulse:061a(c),
                                                                                 SetDisplayAddressAndWrite:06b8(c...
05fc b7 01          STA         PORTB                                   = FFh
05fe 15 02          BCLR        0x2,PORTC                               = FFh
0600 16 02          BSET        0x3,PORTC                               = FFh
0602 17 02          BCLR        0x3,PORTC                               = FFh
0604 14 02          BSET        0x2,PORTC                               = FFh
0606 a6 05          LDA         #0x5

                    LAB_0608                                        XREF[1]:     0609(j)
0608 4a             DECA
0609 26 fd          BNE         LAB_0608
060b 81             RTS

                    *************************************************************
                    *                          FUNCTION                         *
                    *************************************************************
                    undefined DisplayStringUntilQuote()
undefined           <UNASSIGNED>  <RETURN>
                    DisplayStringUntilQuote                         XREF[8]:     WaitForData:03bd(c),
                                                                                 WaitForData:03c4(c),
                                                                                 ShowSystemStatusDisplay:04c4(c),
                                                                                 ShowSystemStatusDisplay:04d3(c),
                                                                                 ShowSystemStatusDisplay:04e4(c),
                                                                                 ShowSystemStatusDisplay:0509(c),
                                                                                 ShowSystemStatusDisplay:0525(c),
                                                                                 0615(j)
060c f6             LDA         X
060d a1 22          CMP         #0x22
060f 27 06          BEQ         LAB_0617
0611 cd 05 d2       JSR         OutputCharacterToDisplay                undefinedOutputCharacterToDispl...
0614 5c             INCX
0615 20 f5          BRA         DisplayStringUntilQuote

                    LAB_0617                                        XREF[1]:     060f(j)
0617 81             RTS
```

```
                    ************************************************************
                    *                          FUNCTION                        *
                    ************************************************************
                    undefined InitDisplayClearPulse()
undefined           <UNASSIGNED>  <RETURN>
                    InitDisplayClearPulse                   XREF[3]:    RESET:0114(c),
                                                                        UpdateTimersAndWait:0138(c),
                                                                        WaitForData:02e6(c)
0618 a6 01          LDA     #0x1
061a cd 05 fc       JSR     SendDisplayCommand                  undefinedSendDisplayCommand()
061d a6 b4          LDA     #0xb4

                    LAB_061f                                XREF[1]:    0620(j)
061f 4a             DECA
0620 26 fd          BNE     LAB_061f
0622 81             RTS

                    ************************************************************
                    *                          FUNCTION                        *
                    ************************************************************
                    undefined LookupCharacterCode()
undefined           <UNASSIGNED>  <RETURN>
                    LookupCharacterCode                     XREF[9]:    DisplayTimeData:03e1(c),
                                                                        ShowSystemStatusDisplay:04ed(c),
                                                                        ShowSystemStatusDisplay:04f7(c),
                                                                        ShowSystemStatusDisplay:0510(c),
                                                                        ShowSystemStatusDisplay:051a(c),
                                                                        ShowSystemStatusDisplay:0537(c),
                                                                        ShowSystemStatusDisplay:0541(c),
                                                                        ShowSystemStatusDisplay:0550(c),
                                                                        ShowSystemStatusDisplay:055a(c)
0623 5c             INCX
0624 bf 1f          STX     DAT_001f                            = FFh
0626 b7 1b          STA     DAT_001b                            = FFh
0628 1f 1b          BCLR    0x7,DAT_001b                        = FFh
062a ae 00          LDX     #0x0

                    LAB_062c                                XREF[1]:    0637(j)
062c d6 07 59       LDA     CharacterLookupTable,X              = 77h
062f b1 1b          CMP     DAT_001b                            = FFh
0631 27 09          BEQ     LAB_063c
0633 5c             INCX
0634 5c             INCX
0635 a1 ff          CMP     #0xff
0637 26 f3          BNE     LAB_062c
0639 a6 20          LDA     #0x20
063b 81             RTS

                    LAB_063c                                XREF[1]:    0631(j)
063c 5c             INCX
063d d6 07 59       LDA     0x759,X=>char_lookup_table_1        = 20h
0640 81             RTS

                    ************************************************************
                    *                          FUNCTION                        *
                    ************************************************************
                    undefined DecodeCharacterFromTable()
undefined           <UNASSIGNED>  <RETURN>
                    DecodeCharacterFromTable                XREF[2]:    WaitForData:02fc(c),
                                                                        WaitForData:0322(c)
0641 bf 1f          STX     DAT_001f                            = FFh
0643 a6 00          LDA     #0x0
0645 b7 1d          STA     DAT_001d                            = FFh
0647 ae 00          LDX     #0x0

                    LAB_0649                                XREF[1]:    0654(j)
0649 d6 06 bc       LDA     CharacterDecodeTable,X              = 08h
                                                                = 0Eh
064c b1 1c          CMP     DAT_001c                            = FFh
064e 27 08          BEQ     LAB_0658
0650 5c             INCX
0651 5c             INCX
0652 a1 fe          CMP     #0xfe
0654 26 f3          BNE     LAB_0649
0656 20 23          BRA     LAB_067b

                    LAB_0658                                XREF[1]:    064e(j)
0658 5c             INCX
0659 d6 06 bc       LDA     0x6bc,X=>char_decode_table_1        = 80h
065c b7 1d          STA     DAT_001d                            = FFh
065e 0e 1d 1e       BRSET   0x7,DAT_001d,LAB_067f               = FFh
0661 97             TAX
```

```
                LAB_0662                                XREF[1]:     0679(j)
0662 d6 06 fd       LDA         0x6fd,X=>char_lookup_table_end_marker      = 4Eh
                                                                       = 80h
0665 a1 ff          CMP         #0xff
0667 26 0a          BNE         LAB_0673
0669 5c             INCX
066a d6 06 fd       LDA         0x6fd,X=>char_lookup_table_final_1         = 09h
066d a1 ff          CMP         #0xff
066f 27 0a          BEQ         LAB_067b
0671 20 13          BRA         LAB_0686

                LAB_0673                                XREF[1]:     0667(j)
0673 b1 1b          CMP         DAT_001b                               = FFh
0675 27 0e          BEQ         LAB_0685
0677 5c             INCX
0678 5c             INCX
0679 20 e7          BRA         LAB_0662

                LAB_067b                                XREF[2]:     0656(j), 066f(j)
067b a6 20          LDA         #0x20
067d 20 0c          BRA         LAB_068b

                LAB_067f                                XREF[1]:     065e(j)
067f 1f 1d          BCLR        0x7,DAT_001d                           = FFh
0681 b6 1d          LDA         DAT_001d                               = FFh
0683 20 06          BRA         LAB_068b

                LAB_0685                                XREF[1]:     0675(j)
0685 5c             INCX

                LAB_0686                                XREF[1]:     0671(j)
0686 d6 06 fd       LDA         0x6fd,X=>char_lookup_table_final_1         = 09h
0689 b7 1d          STA         DAT_001d                               = FFh

                LAB_068b                                XREF[2]:     067d(j), 0683(j)
068b be 1f          LDX         DAT_001f                               = FFh
068d b1 1d          CMP         DAT_001d                               = FFh
068f 81             RTS

                *************************************************************
                *                          FUNCTION                         *
                *************************************************************
                undefined CalculateDisplayPosition()
undefined       <UNASSIGNED>  <RETURN>
                CalculateDisplayPosition                XREF[12]:    WaitForData:0383(c),
                                                                     WaitForData:03a8(c),
                                                                     WaitForData:03ad(c),
                                                                     ShowMessageAndTime:03f6(c),
                                                                     ShowMessageAndTime:040d(c),
                                                                     ShowSystemStatusDisplay:0427(c),
                                                                     ShowSystemStatusDisplay:047e(c),
                                                                     ShowSystemStatusDisplay:0489(c),
                                                                     ShowSystemStatusDisplay:04b0(c),
                                                                     ShowSystemStatusDisplay:04c9(c),
                                                                     ShowSystemStatusDisplay:04d8(c),
                                                                     ShowSystemStatusDisplay:0562(c)
0690 a1 28          CMP         #0x28
0692 2b 12          BMI         LAB_06a6
0694 a0 40          SUB         #0x40
0696 bb 47          ADD         DAT_0047                               = FFh
0698 b7 46          STA         DAT_0046                               = FFh
069a a1 28          CMP         #0x28
069c 2b 02          BMI         LAB_06a0
069e a0 28          SUB         #0x28

                LAB_06a0                                XREF[1]:     069c(j)
06a0 ab 40          ADD         #0x40
06a2 b7 46          STA         DAT_0046                               = FFh
06a4 20 0c          BRA         LAB_06b2

                LAB_06a6                                XREF[1]:     0692(j)
06a6 bb 47          ADD         DAT_0047                               = FFh
06a8 b7 46          STA         DAT_0046                               = FFh
06aa a1 28          CMP         #0x28
06ac 2b 04          BMI         LAB_06b2
06ae a0 28          SUB         #0x28
```

```
                    ************************************************************
                    *                          FUNCTION                        *
                    ************************************************************
                    undefined SetDisplayAddressAndWrite()
     undefined      <UNASSIGNED>  <RETURN>
                    SetDisplayAddressAndWrite                       XREF[2]:    OutputCharacterToDisplay:05dd(c),
                                                                                OutputCharacterToDisplay:05e7(c)
06b0 b7 46          STA         DAT_0046                                        = FFh

                    LAB_06b2                                        XREF[2]:    CalculateDisplayPosition:06a4(j),
                                                                                CalculateDisplayPosition:06ad(j)
06b2 b7 48          STA         DAT_0048                                        = FFh
06b4 1e 46          BSET        0x7,DAT_0046                                    = FFh
06b6 b6 46          LDA         DAT_0046                                        = FFh
06b8 cd 05 fc       JSR         SendDisplayCommand                              undefinedSendDisplayCommand()
06bb 81             RTS

                    CharacterDecodeTable                            XREF[1]:    DecodeCharacterFromTable:0649(R)
06bc 08             undefined1 08h

                    char_decode_table_1                             XREF[1]:    DecodeCharacterFromTable:0659(R)
06bd 80             undefined1 80h

                    char_decode_table_2                             XREF[1]:    DecodeCharacterFromTable:0649(R)
06be 0e             undefined1 0Eh
06bf 80             ??          80h
06c0 0f             ??          0Fh
06c1 80             ??          80h
06c2 8f             ??          8Fh
06c3 80             ??          80h
06c4 ef             ??          EFh
06c5 80             ??          80h
06c6 ff             ??          FFh
06c7 80             ??          80h
06c8 00             ??          00h
06c9 00             ??          00h
06ca d5             ??          D5h
06cb 54             ??          54h    T
06cc d4             ??          D4h
06cd 4c             ??          4Ch    L
06ce 22             ??          22h    "
06cf 2c             ??          2Ch    ,
06d0 40             ??          40h    @
06d1 cc             ??          CCh
06d2 63             ??          63h    c
06d3 cd             ??          CDh
06d4 1c             ??          1Ch
06d5 1e             ??          1Eh
06d6 1d             ??          1Dh
06d7 26             ??          26h    &
06d8 41             ??          41h    A
06d9 46             ??          46h    F
06da 0d             ??          0Dh
06db 16             ??          16h
06dc 03             ??          03h
06dd b1             ??          B1h
06de 05             ??          05h
06df 08             ??          08h
06e0 09             ??          09h
06e1 b4             ??          B4h
06e2 0c             ??          0Ch
06e3 10             ??          10h
06e4 02             ??          02h
06e5 af             ??          AFh
06e6 16             ??          16h
06e7 da             ??          DAh
06e8 54             ??          54h    T
06e9 32             ??          32h    2
06ea 55             ??          55h    U
06eb 38             ??          38h    8
06ec 61             ??          61h    a
06ed ce             ??          CEh
06ee 80             ??          80h
06ef ad             ??          ADh
06f0 42             ??          42h    B
06f1 d6             ??          D6h
06f2 88             ??          88h
06f3 ab             ??          ABh
06f4 95             ??          95h
06f5 c0             ??          C0h
06f6 aa             ??          AAh
06f7 aa             ??          AAh
06f8 c1             ??          C1h
06f9 c8             ??          C8h
```

```
06fa c2             ??         C2h
06fb cb             ??         CBh
06fc fe             ??         FEh
06fd 00             ??         00h
06fe 20             ??         20h
06ff 20             ??         20h
0700 2e             ??         2Eh    .
0701 80             ??         80h
0702 2c             ??         2Ch    ,
0703 ff             ??         FFh
0704 ff             ??         FFh
0705 02             ??         02h
0706 37             ??         37h    7
0707 0b             ??         0Bh
0708 33             ??         33h    3
0709 19             ??         19h
070a 32             ??         32h    2
070b ff             ??         FFh
070c ff             ??         FFh
070d 0b             ??         0Bh
070e 35             ??         35h    5
070f 1b             ??         1Bh
0710 36             ??         36h    6
0711 ff             ??         FFh
0712 ff             ??         FFh
0713 0b             ??         0Bh
0714 39             ??         39h    9
0715 1a             ??         1Ah
0716 30             ??         30h    0
0717 1b             ??         1Bh
0718 38             ??         38h    8
0719 ff             ??         FFh
071a ff             ??         FFh
071b 10             ??         10h
071c 54             ??         54h    T
071d 38             ??         38h    8
071e 49             ??         49h    I
071f b0             ??         B0h
0720 4a             ??         4Ah    J
0721 ff             ??         FFh
0722 ff             ??         FFh
0723 3a             ??         3Ah    :
0724 44             ??         44h    D
0725 3b             ??         3Bh    ;
0726 42             ??         42h    B
0727 ff             ??         FFh
0728 ff             ??         FFh
0729 10             ??         10h
072a 59             ??         59h    Y
072b 44             ??         44h    D
072c 58             ??         58h    X
072d ff             ??         FFh
072e ff             ??         FFh
072f a8             ??         A8h
0730 43             ??         43h    C
0731 ab             ??         ABh
0732 47             ??         47h    G
0733 ff             ??         FFh
0734 ff             ??         FFh
0735 11             ??         11h
0736 3f             ??         3Fh    ?
0737 aa             ??         AAh
0738 4f             ??         4Fh    O
0739 ae             ??         AEh
073a 51             ??         51h    Q
073b ff             ??         FFh
073c ff             ??         FFh
073d ff             ??         FFh
073e ff             ??         FFh
073f ff             ??         FFh
0740 ff             ??         FFh
0741 ff             ??         FFh
0742 ff             ??         FFh
0743 aa             ??         AAh
0744 55             ??         55h    U
0745 c6             ??         C6h
0746 57             ??         57h    W
0747 ff             ??         FFh
0748 ff             ??         FFh
0749 2b             ??         2Bh    +
074a 53             ??         53h    S
074b 80             ??         80h
074c 46             ??         46h    F
074d a8             ??         A8h
074e 45             ??         45h    E
```

```
074f ff             ??          FFh
0750 ff             ??          FFh
0751 81             ??          81h
0752 50             ??          50h    P
0753 83             ??          83h
0754 41             ??          41h    A
0755 85             ??          85h
0756 52             ??          52h    R
0757 ff             ??          FFh
0758 ff             ??          FFh

                    CharacterLookupTable                    XREF[1]:    LookupCharacterCode:062c(R)
0759 00             undefined1  00h

                    char_lookup_table_1                     XREF[1]:    LookupCharacterCode:063d(R)
075a 20             undefined1  20h

                    char_lookup_table_2                     XREF[1]:    LookupCharacterCode:062c(R)
075b 77             undefined1  77h
075c 30             ??          30h    0
075d 11             ??          11h
075e 31             ??          31h    1
075f 6b             ??          6Bh    k
0760 32             ??          32h    2
0761 3b             ??          3Bh    ;
0762 33             ??          33h    3
0763 1d             ??          1Dh
0764 34             ??          34h    4
0765 3e             ??          3Eh    >
0766 35             ??          35h    5
0767 7e             ??          7Eh    ~
0768 36             ??          36h    6
0769 13             ??          13h
076a 37             ??          37h    7
076b 7f             ??          7Fh
076c 38             ??          38h    8
076d 3f             ??          3Fh    ?
076e 39             ??          39h    9
076f 50             ??          50h    P
0770 ff             ??          FFh
0771 51             ??          51h    Q
0772 ff             ??          FFh
0773 54             ??          54h    T
0774 ff             ??          FFh
0775 55             ??          55h    U
0776 ff             ??          FFh
0777 4f             ??          4Fh    O
0778 20             ??          20h
0779 66             ??          66h    f
077a 20             ??          20h
077b ff             ??          FFh
077c cf             ??          CFh

                    char_lookup_table_end_marker            XREF[1]:    DecodeCharacterFromTable:0662(R)
077d 4e             undefined1  4Eh

                    char_lookup_table_final_1               XREF[2]:    DecodeCharacterFromTable:066a(R),
                                                                            DecodeCharacterFromTable:0686(R)
077e 09             undefined1  09h

                    char_lookup_table_final_2               XREF[1]:    DecodeCharacterFromTable:0662(R)
077f 80             undefined1  80h
0780 00             ??          00h
0781 c5             ??          C5h
0782 00             ??          00h
0783 00             ??          00h

                    MOR - Mask Option Register
                    Located at 0x784 on Px
                    MOR
0784 20             ??          20h
                    bootstrap ROM at 0x785-0x7f7
0785 ff             ??          FFh
0786 ff             ??          FFh
0787 ff             ??          FFh
0788 ff             ??          FFh
0789 ff             ??          FFh
078a ff             ??          FFh
078b ff             ??          FFh
078c ff             ??          FFh
078d ff             ??          FFh
078e ff             ??          FFh
078f ff             ??          FFh
0790 ff             ??          FFh
0791 ff             ??          FFh
```

```
0792 ff          ??         FFh
0793 ff          ??         FFh
0794 ff          ??         FFh
0795 ff          ??         FFh
0796 ff          ??         FFh
0797 ff          ??         FFh
0798 ff          ??         FFh
0799 ff          ??         FFh
079a ff          ??         FFh
079b ff          ??         FFh
079c ff          ??         FFh
079d ff          ??         FFh
079e ff          ??         FFh
079f ff          ??         FFh
07a0 ff          ??         FFh
07a1 ff          ??         FFh
07a2 ff          ??         FFh
07a3 ff          ??         FFh
07a4 ff          ??         FFh
07a5 ff          ??         FFh
07a6 ff          ??         FFh
07a7 ff          ??         FFh
07a8 ff          ??         FFh
07a9 ff          ??         FFh
07aa ff          ??         FFh
07ab ff          ??         FFh
07ac ff          ??         FFh
07ad ff          ??         FFh
07ae ff          ??         FFh
07af ff          ??         FFh
07b0 ff          ??         FFh
07b1 ff          ??         FFh
07b2 ff          ??         FFh
07b3 ff          ??         FFh
07b4 ff          ??         FFh
07b5 ff          ??         FFh
07b6 ff          ??         FFh
07b7 ff          ??         FFh
07b8 ff          ??         FFh
07b9 ff          ??         FFh
07ba ff          ??         FFh
07bb ff          ??         FFh
07bc ff          ??         FFh
07bd ff          ??         FFh
07be ff          ??         FFh
07bf ff          ??         FFh
07c0 ff          ??         FFh
07c1 ff          ??         FFh
07c2 ff          ??         FFh
07c3 ff          ??         FFh
07c4 ff          ??         FFh
07c5 ff          ??         FFh
07c6 ff          ??         FFh
07c7 ff          ??         FFh
07c8 ff          ??         FFh
07c9 ff          ??         FFh
07ca ff          ??         FFh
07cb ff          ??         FFh
07cc ff          ??         FFh
07cd ff          ??         FFh
07ce ff          ??         FFh
07cf ff          ??         FFh
07d0 ff          ??         FFh
07d1 ff          ??         FFh
07d2 ff          ??         FFh
07d3 ff          ??         FFh
07d4 ff          ??         FFh
07d5 ff          ??         FFh
07d6 ff          ??         FFh
07d7 ff          ??         FFh
07d8 ff          ??         FFh
07d9 ff          ??         FFh
07da ff          ??         FFh
07db ff          ??         FFh
07dc ff          ??         FFh
07dd ff          ??         FFh
07de ff          ??         FFh
07df ff          ??         FFh
07e0 ff          ??         FFh
07e1 ff          ??         FFh
07e2 ff          ??         FFh
07e3 ff          ??         FFh
07e4 ff          ??         FFh
07e5 ff          ??         FFh
07e6 ff          ??         FFh
```

```
07e7 ff          ??         FFh
07e8 ff          ??         FFh
07e9 ff          ??         FFh
07ea ff          ??         FFh
07eb ff          ??         FFh
07ec ff          ??         FFh
07ed ff          ??         FFh
07ee ff          ??         FFh
07ef ff          ??         FFh
07f0 ff          ??         FFh
07f1 ff          ??         FFh
07f2 ff          ??         FFh
07f3 ff          ??         FFh
07f4 ff          ??         FFh
07f5 ff          ??         FFh
07f6 ff          ??         FFh
07f7 ff          ??         FFh
07f8 00 d6       addr       RESET                    Timer Interrupt Vector
07fa 00 d6       addr       RESET                    External Interrupt Vector
07fc 00 d6       addr       RESET                    SWI Vector
07fe 00 d6       addr       RESET                    Reset vector
```