GenRef
v1.00

MDOS Reference guide.

ANSI Library

(C) Copyright 2023
Beery W. Miller

_____
ANSI - CONTENTS
_____

Contents

_____

ANSI Overview

_____

All ANSI management routines in the self-installing MDOS XOP application is provided to aid a programmer in writing applications requiring ANSI operations beyond the immediate instruction set of the TMS 9995 microprocessor and V9938 video processor for the Geneve 9640.

Opcodes within the ANSI XOP Library 13 extend support access to the VIDEO routines in providing ANSI Video output in both 80-column text mode and 80 column 16-color graphic mode.

All ANSI code was written by InsaneMultitasker on AtariAge and Michael Maksimik and subsequently converted to a standalone driver by Beery Miller with the ability to self-install into the MDOS operating system.

_____
CALLING ANSI FUNCTIONS
_____

The MDOS ANSI Library must be called from within a machine code program running as a task under MDOS. You pass arguments to the ANSI Library via the calling registers.

The MDOS ANSI Library is invoked from a machine code program when software trap number zero (XOP 0) is called with a library number of 13. The calling program's R0 must contain the 16-bit subprogram at the time of the XOP. The following code fragment will open a TCP connection to a hostname or IP address.

```
            LI     R0,>0001     CRU address opcode
            CLR    R1           Let's use a null terminated string
            XOP    @ANSI,0      Access subprogram

ANSI    DATA   13           ANSI Library
STR1    TEXT   'Hello'      String
            BYTE   0            null terminated
```


In the preceding example, two hidden assumptions were made. First it is assumed the STR1 label address is located on a page which is currently mapped into a memory page which has the same 16-bit address page number as its Virtual address page number (read the section on Memory Management). The second assumption is the ANSI label is currently mapped into a memory page which has the same 16-bit address page number as its Virtual address page number.

SET ANSI MODE

Function   Sets the screen and video mode for the ANSI screen.

Parameters   R0 = >0000 (opcode)
                      R1 = Text or Graphics Mode
                      R2 = Telnet handle port (MSB), optional

Results   N/A -

Parameter Description

Text or Graphics Mode   Specify 0 for 80 column text mode or <>0 for Graphics Mode 6 (512x192x16 colors).

Telnet handle Port   Specify the optional TIPI client or server byte for communication between a TCP
                        connection. This is generally only used when there is a request from a server to
                        autodetect an ANSI terminal or cursor position on the screen.

Example Code

```
        LI    R0,>0000    Set ANSI opcode
        CLR   R1          Set 80 column text mode configuration
        LI    R2,>0100    TIPI Client handle byte
        XOP   @ANSI,0     Access subprogram

ANSI    DATA  13          ANSI Library
```

Display ANSI characters

Function Passes the ANSI string to the ANSI interpreter for displaying the ANSI sequence to the screen. This function uses multiple Video XOP functions for cursor placement, color (if applicable) designation, window size and placement, etc.

Parameters R0 = >0001
R1 = String to pass
R2 = String length or 0 if null terminated.

Results Not Applicable

Parameter Description

String Any string, up to 8K in length, can be passed for display purposes.

String Length Length of string. Designate >0000 for a null terminated string. When reading ANSI sequences from a file, it is suggested the record length be used for the R2 value and the location of the buffer be used for the R1 string pointer.

It should also be noted when processing ANSI sequences, care must be taken during an ANSI terminal mode to not break up input and output character/string sequences during the display.

Example Code

```
        LI    R0,>0001    CRU address opcode
        CLR   R1          Let's use a null terminated string
        XOP   @ANSI,0     Access subprogram

ANSI    DATA  13          ANSI Library
STR1    TEXT  'Hello'     String
        BYTE  0           null terminated
```

ANSI VERSION

Function   Returns the current version of the ANSI Interpreter and status if the ANSI XOP driver is loaded.

Parameters   R0 = >0002

Results   R0 = Result code
                      R1 = Version

Parameter Description

Result code   >FFFF if ANSI driver installed
                      >0000 for no ANSI driver installed

Version   Reports version number of current ANSI driver. Version will return >0100 for the current version if XOP driver is installed. The presence of the ANSI driver should be tested first in R0 prior to testing R1.

Example Code

```
          LI    R0,>0002    Open Host opcode
          XOP   @ANSI,0     Access subprogram
          MOV   R0,R0       Test for ANSI Driver
          JEQ   NOCONCT     No driver
          CI    R1,>0100    Version test
          JHE   GOODANSI    ANSI present, version >0100
          JMP   NOANSI      Go to no ANSI driver mismatch and evaluate

  ANSI    DATA  13          ANSI Library
```