Original and scanned by Bifi
Converted to PDF by HansO, 2002

Inhoudsopgave
Inleiding ..... 3
Overzicht control-codes ..... 5
VT-52 codes ..... 6
ANSI codes ..... 7
Overzicht van alle jANSI codes: ..... 8
jANSI-codes ..... 13
TSR-Call routine ..... 20

## Inleiding

jANSI is een TSR-programma voor MSX2 computers. Het maakt het mogelijk om scherm stuurcodes volgens de ANSI-standaard te verwerken. Deze door het American National Standards Institute vastgelegde standaard wordt veel toegepast op IBM compatible computers en in vele Bulletin Board Systems, of wel BBS'en.

Het is dan ook waarschijnlijk dat er op korte termijn allerlei toepassingsprogramma's geschreven worden, die gebruik maken van de mogelijkheden van de jANSI TSR. Hierbij kan gedacht worden aan communicatieprogramma's en editors om ANSI-schermen mee te ontwerpen. Maar ook bestaande programma's die georiënteerd zijn op het standaard tekstscherm, kunnen vaak door enkele simpele toevoegingen gebruik maken van de speciale ANSItekstmogelijkheden zoals kleurgebruik en speciale effecten zoals knipperende en onderstreepte tekst.

De ANSI-standaard gaat uit van een tekstscherm van 80 tekens bij 25 regels. De tekst kan weergegeven worden in acht verschillende kleuren, die zowel gewoon als extra helder high lighted weergegeven kunnen worden. Verder zijn er onder andere codes beschikbaar om tekst te laten knipperen of te onderstrepen.

Wanneer jANSI geactiveerd wordt, wordt de video-processor in een grafische schermmode geschakeld. De tekstmode van de MSX2 video-processor is namelijk niet in staat om aan de ANSI specificaties te voldoen. Het grafische SCREEN 7 biedt echter meer dan genoeg mogelijkheden.

## Scherm stuurcodes

De jANSI schermmode is volledig compatible met het tachtig kolommen tellende MSX2 tekstscherm, voor wat betreft de besturingskarakters. Dit houdt in dat alle control-codes en VT- 52 stuurcodes ondersteund worden.

Karakters en stuurcodes kunnen naar het scherm worden geschreven middels de standaard MSX-BASIC statements zoals PRINT, de schermfuncties van de MSX-DOS, de BIOS routine CHPUT of de snelle TSR-Call functie 'DmpStr'.

De stuurcodes worden in deze handleiding als volgt omschreven. De code 'Esc' of escape, dient gelezen te worden als "het karakter met ASCII nummer $27^{\prime \prime}$. Na het escape-karakter volgen én of meerdere karakters die de functic van het commando bepalen. Zo'n code wordt een 'escape-reeks' of 'escape sequence' genoemd.

De linkerbovenhoek van het scherm wordt aangeduid door het cobrdinaat 1,1 en de rechteronderhoek door $\mathbf{8 0 , 2 5}$.

In escape-reeksen wordt onderscheid gemaakt tussen hoofd en kleine letters. Hier en daar zijn spaties toegevoegd om de leesbaarheid te verhogen, wees erop attent dat spaties niet in een escape-reeks mogen voorkomen.

## Control-codes

Onder 'control-codes' worden de karaktercodes verstaan, die een schermcommando voorstellen en dus niet als een teken op het scherm weergegeven kunnen worden. In de bijgaande tabel staat de functie van de door jANSI ondersteunde control-codes weergegeven.

Indien de MSX-karakterset actief is, zullen karakters met een ASCII-waarde vanaf 0 tot en met 31 genegeerd worden, tenzij het een control-code betreft. In de IBM-karakterset worden de door de control-codes ongebruikte ASCIIwaarden gebruikt voor het weergeven van grafische symbolen.

## Overzicht control-codes

|  | Naam | Functie |
| :---: | :---: | :---: |
| 0 | NUL | Wordt genegeerd |
| 1 | SOH | MSX karakterset: Voorloopteken van een grafische karaktercode. Het grafische vervolgteken moet een ASCII-waarde hebben tussen 64 en 96. |
| 7 | BEL | Laat een pieptoontje horen. |
| 8 | BS | Verplaatst de cursor één positie naar links. Indien de cursor zich aan het begin van de regel bevindt, wordt hij naar het einde van de bovenliggende regel verplaatst. |
| 9 | HT | Verplaatst de cursor naar de volgende horizontale tab-stop. |
| 10 | LF | Verplaatst de cursor én regel omlaag. De verticale positie van de cursor blijft ongewijzigd. Het scherm wordt omhoog gescrolled indien de cursor zich op de onderste regel bevindt. |
| 11 | VT | Verplaatst de cursor naar positie 1,1. |
| 12 | FF | Wist het scherm en verplaatst de cursor naar positie 1,1. |
| 13 | CR | Verplaatst de cursor naar positie én van de huidige regel. |
| 27 | ESC | Voorloopteken voor een reeks stuurcodes. Dit kunnen ANSI, VT52 of jANSI codes zijn. |
| 28 | FS | Verplaatst de cursor שen positie naar rechts. Gaat eventueel verder aan het begin van de volgende regel, stopt in de rechter onderhoek van het scherm. |
| 29 | GS | Verplaatst de cursor één positie naar links. Gaat eventueel verder aan het einde van de vorige regel, stopt in de linker bovenhoek van het scherm. |
| 30 | RS | Verplaatst de cursor én regel omhoog. Stopt op de bovenste regel. |
| 31 | US | Verplaatst de cursor én regel omlaag. Stopt op de onderste regel. |
|  | DEL | Wist het karakter links van cursor. Verplaatst de cursor één positie naar links. |

## VT-52 codes

De VT-52 schermstuurcodes vormen een onderdeel van de MSX-standaard. jANSI herkent zowel ANSI als VT- 52 codes. De nu volgende tabel toont een overzicht van de functie van alle VT- 52 codes.

## Code Functie

Esc A Verplaatst de cursor énn regel omhoog. Stopt bij de bovenste regel.
Esc B Verplaatst de cursor één regel omlaag. Stopt bij de onderste regel.
Esc C Cursor rechts, stopt op het einde van de regel.
Esc D Cursor links, stopt op het begin van de regel.
Esc E Clear screen. Het scherm wordt gevuld met spaties. De cursor wordt verplaatst naar positie 1,1 .
Esc H Verplaatst de cursor naar positie 1,1.
Esc J Wist het scherm vanaf de cursor, verplaatst de cursor niet.
Esc $\mathbf{j} \quad$ Clear screen. Het scherm wordt gevuld met spaties. De cursor wordt verplaatst naar positie 1,1 .
Esc K Wist de regel vanaf de cursor, verplaatst de cursor niet.
Esc $L \quad$ Voegt een lege regel in boven de regel waar de cursor zich bevindt. De onderliggende regels worden omlaag gescrolled. De cursor wordt verplaatst naar het begin van de nieuwe lege regel.
Esc $1 \quad$ Wist de gehele regel, verplaatst de cursor niet.
Esc M Verwijdert de gehele regel, scrolled de onderliggende regels omhoog. Verplaatst de cursor naar het begin van de volgende regel.
Esc $\times 4$ Schakelt een blokvormige cursor in.
Esc $\times 5$ Maakt de cursor onzichtbaar.
Esc Y n m Verplaatst de cursor naar positie naar regel <n>en kolom <m>. De linkerbovenhoek van het scherm is $n=m=20 h$ (ASCII space).
Esc y 4 Schakelt een lijnvormige cursor in.
Esc y 5 Maakt de cursor zichtbaar.

## ANSI codes

ANSI schermstuurcodes beginnen altijd met de karakters escape (ASCII 27) en de rechte haak openen ("[", ASCII 91). Vervolgens volgt eventueel een reeks getallen, gescheiden door puntcomma's. Deze getallen zijn parameters van de betreffende ANSI-functie. De ASCI-cijfers '0'-9' worden gebruikt om de getallen vormen. Een ANSI-code eindigt bij het eerst volgende karakter dat geen ASCII-cijfer of puntcomma is. Dit afsluitende karakter bepaalt welk ANSI-commando uitgevoerd moet worden.

Op sommige details wijkt jANSI af van de ANSI standaard. Zo zijn er slechts 15 in plaats van 16 tekskleuren beschikbaar: de kleur 'wit' is namelijk gelijk aan 'helder wit'. Het zestiende kleurnummer wordt gebruikt om knipperende tekst weer te geven. Alle knipperende tekst op het scherm heeft dus dezelfde kleur.
jANSI ondersteunt alle in de ANSI-standaard gedefineerde commando's, alsmede enkele functies uit de semi-standaard 'PingAnsi'. Sommige 'ANSI device drivers' van MS-DOS computers ondersteunen deze extra commando's niet. Zie het commentaar bij de betreffende functies in het nu volgende overzicht.

## Overzicht van alle jANSI codes:

| Naam: | Cursor position (CUP) |
| :---: | :---: |
| String: | Esc[<regel>;<kolom>H |
| Voorbeeld: | PRINT CHR\$(27);"[25;70H"; |
| Functie: | Plaatst de cursor op de opgegeven schermcoördinaten. <regel> = De regel van de cursor. <br> Indien <regel> wordt weggelaten, wordt de waarde ' 1 ' aangenomen. <br> <kolom> = De horizontale positie van de cursor. <br> Indien <kolom> wordt weggelaten, wordt de waarde ' 1 ' genomen. |
| Naam: | Cursor position (HVP) |
| String: | Esc[<regel>;<kolom>f |
| Voorbeeld: | PRINT CHR\$(27);"[;10f"; |
| Functie: | Zelfde als CUP, het wordt aangeraden om alleen de CUP functie te gebruiken. |
| Naam: | Cursor position |
| String: | Esc[<regel>;<kolom>R |
| Voorbeeld: | PRINT CHR\$(27);"[2R"; |
| Functie: | Zelfde als CUP, het wordt aangeraden om alleen de CUP functie te gebruiken. |
| Naam: | Cursor up (CUU) |
| String: | Esc[<aantal>A |
| Voorbeeld: | PRINT CHR\$(27);"[2A"; |
| Functie: | Verplaatst de cursor omhoog. Stopt bovenaan het scherm. <aantal> = Aantal regels. <br> Default voor <aantal> is de de waarde ' 1 '. |

Naam: Cursor down (CUD)
String: Esc[<aantal>B
Voorbeeld: PRINT CHR (27); ${ }^{\left[1 B^{n} \text {; }\right.}$
Functie: Verplaatst de cursor omlaag. Stopt onderaan het scherm. <aantal> = Aantal regels.
Indien <aantal> wordt weggelaten, wordt de waarde '1' genomen.

## Naam: Cursor forward (CUF)

String: Esc[<aantal>C
Voorbeeld: PRINT CHR \$(27);"[10C";
Functie: Verplaatst de cursor naar rechts. Stopt wanneer de rechter zijkant van het scherm bereikt is.
<aantal> = Aantal posities.
Default voor <aantal> is de de waarde ' 1 '.

Naam: Cursor backward (CUB)
String: Esc[<aantal>D
Voorbeeld: PRINT CHR\$(27);"[3D";
Functie: Verplaatst de cursor naar links. Stopt wanneer de linker zijde van het scherm bereikt is.
<aantal> = Aantal posities.
Indien <aantal> wordt weggelaten, wordt de waarde ' 1 ' aangenomen.

Naam: $\quad$ Save cursor position (SCP)
String: Esc[s
Voorbeeld: PRINT CHR\$(27);"[s";
Functie: Bewaart de huidige cursorpositie, zodat deze door de RCP functie weer hersteld kan worden.

Naam: Restore cursor position (RCP)
String: Esc[u
Voorbeeld: PRINT CHR\$(27);"[u";
Functie: Herstelt de cursorpositie, die door de laatste aanroep van SCP is opgeslagen.

Naam: $\quad$ Erase in display (ED)
String: Esc[<mode>]
Voorbeeld: PRINT CHR\$(27);"[2]";
Functie: Wist (een gedeelte van) het scherm.
<mode> $=0$ : Wist vanaf de linker bovenhoek tot aan de cursorpositie. Laat de cursorpositie onververanderd.
1: Wist vanaf de cursor tot en met de rechter onderhoek. Laat de cursorpositie onveranderd.
2: Wist het gehele scherm. Plaatst de cursor in de linker bovenhoek.

De parameter <mode> mag niet worden weggelaten. Sommige ANSI-drivers ondersteunen alleen mode 2.

Naam: Erase in line (EL)
String: $\quad$ Esc[<mode>K
Voorbeeld: PRINT CHR\$(27);"[K";
Functie: Wist (een gedeelte van) de huidige regel. Laat de cursor op de huidige positie staan.
<mode> $=0$ : Wist vanaf de cursorpositie tot en met het einde van de regel.
1: Wist vanaf het begin van de regel tot aan de cursorpositie.
2: Wist de gehele huidige regel.
Indien de paramter <mode> wordt weggelaten, wordt de waarde ' 0 ' aangenomen. Sommige ANSI-drivers herkennen alleen mode 0 .

Naam: Insert lines (1L)
String: Esc[<aantal>L
Voorbeeld: PRINT CHR $\$(27)$;"[3L";
Functie: $\quad$ Voegt lege regels in boven de regel waar de cursor zich bevindt. De onderliggende regels worden omlaag gescrolled. De cursor wordt verplaatst naar het begin van de bovenste nieuwe lege regel.
<aantal> = Aantal regels dat ingevoegd moet worden.
Indien <aantal> wordt weggelaten, wordt de waarde ' 1 ' aangenomen.

Naam: Delete lines (DL)
String: Esc[<aantal>M
Voorbeeld: PRINT CHR\$(27);"[M";
Functie: Verwijdert gehele regels, te beginnen bij de cursorregel. De onderliggende regels worden omhoog gescrolled. Verplaatst de cursor naar het begin van de volgende regel.
<aantal> = Aantal regels dat verwijderd moet worden.
Indien <aantal> wordt weggelaten, wordt de waarde ' 1 ' aangenomen.

Naam: $\quad$ Set graphics rendition (SGR)
String: Esc|<code>;<code>;...<code>m
Voorbeeld: PRINT CHR\$(27);"[0;1;32;42m";
Functie: $\quad$ Stelt de huidige voor- en achtergrondkleuren in, of de attributen. Wanneer meerdere parameters worden opgegeven, worden ze đén voor één uitgevoerd.
<code> $=0$ : Zet alle attributen uit, stelt de voorgrondkleur in op 7 (wit) en de achtergrond op kleur 0 (zwart).
1: High lighted aan. Maakt de voorgrondkleur extra helder.
4: Onderstrepen aan.
5: Knipperen aan.
7: Inverse Video aan. Wisselt de vooren achtergrondkleuren om.
8: Onzichtbaar. Maakt de voorgrondkleur gelijk aan de achtergrondkleur.
30-37: Stelt de voorgrondkleur in volgens de onderstaande tabel.
40-47: Stelt de achtergrondkleur in volgens de volgende tabel.

Code Kleur

30/40 Zwart
31/41 Rood
32/42 Groen
33/43 Bruin/geel
34/44 Blauw
35/45 Magenta
36/46 Cyaan
37/47 Wit

Ingeschakelde attributen blijven net zo lang actief totdat de code 0 is gegeven.
Wanneer er geen <code> wordt opgegeven, wordt 'code 0' uitgevoerd.

## jANSI-codes

jANSI bevat diverse extra commando's, welke niet door andere ANSI-drivers herkend worden. Ze kunnen worden gebruikt om 'vaste' instellingen te wijzigen, zoals paletkleuren en de gebruikte karakterset. Deze commando's kunnen worden gegeven vanuit een toepassingsprogramma dat speciaal voor jANSI is geschreven, zoals een communicatieprogramma.

Het is niet verstandig om jANSI-codes in gewone tekstbestanden te plaatsen, bijvoorbeeld in een menu van een bulletin-board. Ten eerste is dan niet meer zeker wat het effect zal zijn wanneer de tekst wordt afgedrukt door andere ANSI-driver dan jANSI. Ten tweede zal de standaardinstelling van jANSI na het weergeven van zo'n tekst gewijzigd zijn, wat door de gebruiker meestal niet op prijs gesteld zal worden.

De structuur van de jANSI-codes is als volgt:
Naam: Init
Nummer: 0
String: Esc[0. of Esc[.
Voorbeeld: PRINT CHR\$(27);"[.";
Functie: Initialiseert jANSI. Dit commando moet zijn uitgevoerd voordat de overige ANSI en jANSI-codes herkend worden. De grafische schermmode SCREEN 7 wordt ingeschakeld en het scherm wordt schoongemaakt. De standaard tekstkleur is wit op een zwarte achtergrond. De IBM-karakterset wordt ingeschakeld, en de optie om na een Carriage Return automatisch een Line Feed af te drukken wordt uitgeschakeld.

De jANSI-mode wordt weer verlaten door door de 'Quit' (1)
functie. Wanneer SCREEN 7 wordt verlaten door een toepassingsprogramma, zal de jANSI-mode automatisch worden uitgeschakeld. Dit is bijvoorbeeld het geval in de BASIC-werkomgeving. Iedere keer wanneer een programma wordt verlaten, wordt door de BASIC-ROM het standaard tekstscherm ingeschakeld. Hierdoor wordt de jANSI-mode uitgeschakeld.

## M

Naam: Quit
Nummer: 1
String: Esc[1.
Voorbeeld: PRINT CHR\$(27);"[1.";
Functie: Verlaat de jANSI-mode. De standaard MSX2-tekstmode wordt weer ingeschakeld. Gebruik de functie Init (0) om jANSI weer actief $t e$ maken.

Naam: MSXChr
Nummer: 2
String: Esc[2.
Voorbeeld: PRINT CHR\$(27);"[2.";
Functie: Activeert de MSX karakterset. Ook grafische karakters die vooraf worden gegaan door het header-karakter SOH kunnen worden afgedrukt.

Naam: IBMChr
Nummer: 3
String: Esc[3.
Voorbeeld: PRINT CHR\$(27);"[3.";
Functie: Activeert de IBM karakterset. Grafische karakters met een ASCII-code tussen 0 en 16 kunnen rechtstreeks worden afgedrukt, tenzij het een stuurcode betreft zoals CR, LF, en dergelijke.

## Naam: SetLF

Nummer: 4
String: Esc[4;<mode>.
Voorbeeld: PRINT CHR\$(27);"[4;1.";
Functie: <mode> $=0$ : Voer geen regelopvoer (Line Feed) uit nadat een CR-karakter (ASCII-code 13)
is afgedrukt. Dit is de standaard instelling na initialisatie van jANSI.
1: Voer wel regelopvoer uit nadat het CR is afgedrukt.
Indien <mode> wordt weggelaten, wordt de waarde ' 0 ' aangenomen.

Naam: SavScr
Nummer: 5
String: Esc[5.
Voorbeeld: PRINT CHR\$(27);"[5.";
Functie: Kopieert het huidige scherm naar cen tweede pagina in het Video-RAM. Ook de huidige tekstkleuren en de attributen die met "Esc[m" zijn ingesteld, worden opgeslagen.

Naam: GetScr
Nummer: 6
String: Escl6.
Voorbeeld: PRINT CHR\$(27):"[6.";
Functie: Haalt een schermpagina op, die door het SavScr commando is opgeslagen.

Naam: LinMov
Nummer: 7
String: Esc[7;<bron>;<bestemming>.
Voorbeeld: PRINT CHR $\$(27$ );" $[7 ; 1 ; 2 . " ;$
Functie: Kopieert een schermregel.
<bron> = Nummer van de te kopiëren regel
<bestemming> = Nummer van de bestemmingsregel

Naam: jBeep
Nummer: 8
String: Esc[8;<tijd>.
Voorbeeld: PRINT CHR\$(27);"[8;5.";
Functie: Geeft een geluidssignaal <tijd> = Duur van het signaal in eenheden van éen vijfde seconde.
Indien <tijd> wordt weggelaten, wordt de waarde ' 1 ' genomen.

## Naam: StBlnk

Nummer: 9
String: Esc[9;<voorgrondtijd>;<achtergrondtijd>.
Voorbeeld: PRINT CHR\$(27);"[9;20;10.";
Functie: Instellen van de knippersnelheid. Tekst kan knipperend worden gemaakt door middel van het ANSI-commando "Esc[5m".
<voorgrondtijd> = Aantal eenheden van éen vijftigste seconde dat de tekst in voorgrondkleur getoond wordt. <achtergrondtijd> = Aantal eenheden van én vijftigste seconde dat de tekst in achtergrondkleur getoond wordt. De maximumwaarde voor <voorgrondtijd> en <achtergrondtijd> is 255.

Naam: ChgPlt
Nummer: 10
String: Esc[10;<kleur>;<rood>;<groen>;<blauw>.
Voorbeeld: PRINTCHR\$(27);"[10;1;6;2;1."
Functie: Wijzigt van het palet waaruit <kleur> is opgebouwd.
<kleur> = Nummer ( $0-14$ ) van de in te stellen schermkleur.
De kleurnummers 0-7 zijn de acht standaard ANSI-kleuren, de nummers 8-14 stellen de zogenaamde 'high lighted' kleuren voor.
<rood> Intensiteit van de paletkleur rood (0-7)
<groen> Intensiteit van de paletkleur groen (0-7)
<blauw> Intensiteit van de paletkleur blauw (0-7)
Aan de kleumummers wordt na de installatie van jANSI de volgende paletinstelling toegekend.

| Nr | Kleur | R | G B |  |
| :--- | :--- | :--- | :--- | :--- | :--- |
|  |  |  |  |  |
| 0: | zwart | 0 | 0 | 0 |
| 1: | rood | 5 | 0 | 0 |
| 2: | groen | 1 | 4 | 1 |
| 3: | brin | 3 | 2 | 0 |
| 4: | blauw | 1 | 2 | 5 |
| 5: | magenta | 5 | 1 | 4 |
| 6: | cyaan | 1 | 4 | 5 |
| 7: | wit | 7 | 7 | 7 |
| 8: | grijs | 4 | 4 | 4 |
| 9: | helder rood | 7 | 2 | 2 |
| 10: | licht groen | 3 | 7 | 3 |
| 11: | licht geel | 6 | 6 | 0 |
| 12: | licht blauw | 2 | 3 | 7 |
| 13: | helder magenta | 7 | 2 | 5 |
| 14: | helder cyaan | 3 | 7 | 7 |

Naam: $\quad$ SetBdr
Nummer: 11
String: Esc[11;<kleur>.
Voorbeeld: PRINT CHR\$(27);"[11;4.";
Functie: Stelt de borderkleur in.
<kleur> = Nummer van 0-14. Zie voor de betekenis van de kleurnummers de tabel bij functie 10 (ChgPlt).
Indien <kleur> wordt weggelaten, wordt de waarde ' 0 ' genomen.

Naam: Pause
Nummer: 12
String: Esc[12;<tijd>.
Voorbeeld: PRINT CHR\$(27);"[12;5.";
Functie: Pauseert de schermuitvoer.
<tijd> = Aantal eenheden van én vijfde van een seconde dat er gewacht moet worden.
Indien <tijd> wordt weggelaten, wordt de waarde ' 1 ' genomen.

Naam: StScrl
Nummer: 13
String: Esc[13;<mode>.
Voorbeeld: PRINT CHR\$(13);"13;1.";
Functie: $\quad$ Stelt de methode in waarop de schermregels omhoog worden verplaatst, of wel gescrolled.
<mode> $=0$ : Hard scroll. Het beeld schuift in éen stap een regel omhoog.
1: Smooth scroll. Het beeld schuift beeldlijn voor beeldlijn omhoog. De smooth scroll wordt via een interruptroutine afgehandeld, tijdens het scrollen kan de computer gewoon doorwerken.
Indien <mode> wordt weggelaten, wordt de waarde ' 0 ' genomen.

Naam: CsrCol
Nummer: 14
String: Esc[14;<kleur>.
Voorbeeld: PRINT CHR\$(27);"[14;13.";
Functie: Wijzigt de kleur van de cursor.
<kleur> = Nummer (0-14) van de in te stellen schermkleur.
Zie voor een overzicht van de beschikbare kleurnummers de tabel bij de functie ChgPlt (10). Om te voorkomen dat de cursor weg zou vallen tegen de achtergrondkleur, is het verstandig om étn van de high-lighted kleuren (8-14) te kiezen.

## TSR-Call routine

De TSR 'jANSI' kan vanuit machinetaal worden aanroepen via de Mem-Man-routine 'TSR-Call' - met functienummer 63. Onder MemMan 2.3 kan ook gebruik worden gemaakt van de functie 'XTsrCall' (61).

Om een TsrCall te kunnen uitvoeren, moet eerst de TSR ID-code worden opgevraagd middels de functie 62 'GetTsrID'. Hierbij dient de identificatienaam van jANSI te worden opgegeven. De ID-naam van jANSI is: "MST jANSI".

De interactieroutine van jAnsi versie 1.0 bevat vier functies. De functiecode moet in het ' $A$ '-register worden geplaatst. De werking van deze functies is als volgt.

Naam: GetVer
Nummer: 0
Functie: Bepaal jANSI versienummer
In: Niets
Uit: $\quad \mathbf{H}=$ Versienummer
$\mathrm{L}=$ Update nummer
$B=$ Debug nummer
Opm: Het versienummer heeft volgende formaat: H.LB.
Bijvoorbeeld, Voor jANSI versie 4.51 geldt dat ' H ' de waarde 4 heeft, registers 'L' de waarde 5 en ' $B$ ' de waarde 1 .

Naam: GetSta
Nummer: I
Functie: Bepaal de status van jANSI
In: $\quad$ Niets
Uit: $\quad H L=$ Statusvlaggen.
Opm: De betekenis van de bits in register HL is als volgt:

## Bit Functie

$0 \quad 1=j$ ANSI is geïnitialiseend
1-15 Gereserveerd, moeten altijd 0 zijn
Naam: IniDmp
Nummer: 2
Functie: Initialiseer de 'DmpStr’ functie
In: $\quad H L=$ Statusvlaggen voor 'DmpStr'.
Uit: Niets
Opm: De betekenis van de bits in register HL is als volgt:

## Bit Functie

$0 \quad 1=$ Keer terug bij ieder vol scherm
1 1 = Keer terug voor ieder CLS-commando
2 1 = Negeer CLS-commando's
$3 \quad 1=$ Keer terug bij toetsdruk
4-15 Gereserveerd, moeten altijd 0 zijn

Naam: DmpStr
Nummer: 3
Functie: Druk een tekststring af
In: $\quad$ HL $=$ Startadres van de test (boven adres \&H7FFF)
DE = Aantal af te drukken karakters
Uit: $\quad A=$ Statuscode
DE = Aantal nog af te drukken karakters
HL = Adres van cerst volgende karakter
Opm: Deze functie kan worden gebruikt om tekstblokken sneller af te drukken, dan mogelijk is door middel van de standaard BIOS routines.

Het afdrukken kan door de 'DmpStr' routine onder bepaalde voorwaarden worden afgebroken. Deze voorwaarden kunnen middels de functie 'IniDmp' gesteld worden.
De conditie waarop het afdrukken werd onderbroken, wordt aangegeven door de statuscode in register ' $A$ '. De betekenis hiervan is als volgt:

## Code Betekenis

0 De tekst is helemaal afgedrukt
1 Eerst volgende karakter is een CLS-commando
2 Scherm is vol, er zijn 25 regels afgedrukt
3 Er is een toets ingedrukt. Deze toetscode kan middels de BIOS-routine CHGET opgehaald worden

## Index

| ANSI codes | 8 | ID-naam | 20 |
| :--- | ---: | :--- | ---: |
| ANSI Standaard | 3 | IniDmp | 21 |
| BBS | 3 | Init | 13 |
| Bulletin Board System | 3 | Insert lines (IL) | 11 |
| ChgPlt | 17 | jANSI codes | 13 |
| Coordinaten | 4 | jBeep | 16 |
| CsrCol | 19 | LinMov | 15 |
| Cursor backward (CUB) | 9 | MSXChr | 14 |
| Cursor down (CUD) | 9 | Pause | 18 |
| Cursor forward (CUF) | 9 | Quit | 14 |
| Cursor position | 8 | Restore cursor position (RCP) | 10 |
| Cursor position (CUP) | 8 | Save cursor position (SCP) | 9 |
| Cursor position (HVP) | 8 | SavScr | 15 |
| Cursor up (CUU) | 8 | SCREEN 7 | 3 |
| DmpStr | 22 | Set graphics rendition (SGR) | 12 |
| Erase in display (ED) | 10 | SetBdr | 18 |
| Erase in line (EL) | 10 | SetLF | 15 |
| GetScr | 15 | StBInk | 16 |
| GetSta | 21 | StScrl | 18 |
| GetVer | 20 | versienummer | 20 |
| IBMChr | 14 | VT52 | 6 |

