

N A S C O M

Betriebssystem

N A S - S Y S

NASCOM - Betriebssystem NAS - SYS

Inhalt

Das NASCOM - Betriebssystem NAS - SYS	1
Liste der Befehle	2
Eingabe eines Befehls	3
Fehler bei der Eingabe	3
Korrigieren eines Fehlers	3
Editiermöglichkeiten auf dem Bildschirm	4
Die Tastatur	5
Genaue Beschreibung der Systembefehle	6
Wie man ein Programm eingibt und testet	14
Anzeige der CPU - Register	15
Einfache Anweisungen für die Ein/Ausgabe	15
Wie man ein Programm beendet	16
NAS- SYS Restart- Befehle und Unterprogramme	17
Ein/Ausgabe	22
NAS-SYS Arbeitsspeicher	24
Adressierung des Bildschirmspeichers	25
Anhang	
NAS-SYS Assemblerausdruck	26

Das NASCOM - Betriebssystem N A S - S Y S

NAS - SYS ist ein Betriebssystem für den NASCOM 1 oder den NASCOM 2 Mikrocomputer. Es macht es leicht, Programme in Maschinensprache einzugeben, zu testen und ausführen zu lassen. Es enthält darüberhinaus eine ganze Reihe nützlicher Unterprogramme, die von Benutzerprogrammen aufgerufen werden können. Diese Unterprogramme werden auch von höheren Programmiersprachen, wie etwa dem 8K - Basic, benutzt.

Folgende Möglichkeiten sind gegeben:

Veränderung und Darstellung von Speicher- Inhalten.

Normale Ausführung von Programmen, Einzelschrittverarbeitung oder Ausführung bis zu einem bestimmbaren "Break-Point" (Programm-Unterbrechung). Während der Einzelschrittverarbeitung oder nach einem Break-Point werden alle CPU- Register angezeigt.

Komfortable Editiermöglichkeiten mit einem blinkenden, nicht zerstörenden Cursor. Dies macht es leicht, Befehle zu verbessern und neu einzugeben.

Das Tastatur- Programm erlaubt es, mit der erweiterten Tastatur des NASCOM 2 alle möglichen Zeichen einzugeben, alle Möglichkeiten sind auch mit der Tastatur des NASCOM 1 gegeben.

Mit einem Cassetten - Recorder ist es möglich, Programme zu speichern. Die benutzte Methode ist schnell und erkennt Lesefehler. Es ist möglich, Cassetten zu erstellen, die automatisch eingelesen werden, und das Maschinensprache- Programm automatisch ausführen, ohne daß dafür ein Befehl von der Tastatur gegeben werden muß.

Alle parallelen Ein/Ausgabekanäle können mit direkten Befehlen kontrolliert werden.

NAS - SYS beinhaltet die vollständige Unterstützung von Terminals oder Druckern mit Serienschnittstelle, sodaß der Computer beispielsweise auch mit einem Teletype betrieben werden kann. Die Speicherung von Programmen mit Lochstreifen ist möglich.

Wenn man den Computer mit einem Akustik- Koppler an einen Timesharing - Dienst anschließt, kann er als Terminal benutzt werden. Auch der Austausch von Programmen und Daten zwischen zwei NASCOMs ist möglich.

Da intern mit dem ASCII - Code (American Standard Code for Information Interchange) gearbeitet wird, ist es sehr einfach, andere Peripherie-Geräte, wie etwa einen Drucker mit Parallelschnittstelle zu betreiben.

Liste der Befehle

Axxxx yyyy	Hexadezimal - Arithmetik
Bxxxx	Setzen oder löschen des Breakpoints
Cxxxx yyyy zzzz	Daten von einem Bereich in einen anderen kopieren
Exxxx	Programm ausführen
Gxxxx yyyy zzzz	Selbstladende Cassette erzeugen
H	Terminal - Betrieb (Halb-Duplex)
Ixxxx yyyy zzzz	Kopieren ohne überlappende Bereiche zu zerstören
J	Springe nach FFFA - "Kaltstart" des BASIC
Kxx	Optionen der Tastatur auswählen
L	Code von Lochstreifen lesen
Mxxxx	Speicherinhalt verändern oder überprüfen
N	Optionen rücksetzen
Oxx yy	Ausgabe eines Datums an einen Port
Qxx	Datum von einem Port holen
R	Cassette lesen
Sxxxx	Einzelschrittverarbeitung
Txxxx yyyy zzzz	Anzeige eines Speicherbereichs oder Lochstreifen schreiben
U	Aktivierung der vom Benutzer bereitgestellten Ein/Ausgabe - Programme
V	Überprüfen einer Cassette
Wxxxx yyyy	Cassette schreiben
Xxx	Aktivierung der externen Geräte mit serieller Schnittstelle
Z	Springe nach FFFD - "Warmstart" des BASIC

(D,F,P und Y - Befehle existieren nicht!)

Eingabe eines Befehls an das Betriebssystem

Schreiben Sie den Buchstaben des Befehls in die erste Stelle einer Zeile, gefolgt von den erforderlichen Variablen. Jede Variable muß von einer anderen durch ein oder mehrere Leerzeichen getrennt werden. Dann drücken Sie die 'Enter' - ('New Line') Taste. Die Zeile, die eingegeben wird, ist immer die Zeile, in der der Cursor blinkt, wenn die 'Enter'- Taste gedrückt wird.

Wenn die erste Stelle einer Zeile leer ist, wird kein Befehl ausgeführt.

Fehler bei der Eingabe eines Befehls

Wenn ein ungültiger Befehl eingegeben wurde, wird in der nächsten Zeile 'Error' angezeigt.

Folgende Fehler werden erkannt:

- Ungültige Zeichen für Befehle. (Wenn das Befehls- Zeichen leer ist, wird die Zeile einfach ignoriert.)
- Mehr als zehn Variable nach einem Befehls- Zeichen. (Höchstens drei Variable werden vom Betriebsprogramm benötigt, aber von einem Benutzer- Programm können bis zu zehn Variable benutzt werden, die nach einem "E" - Befehl eingegeben wurden.)
- Jede Variable, die nicht eine Hexadezimal- Zahl ist. (Ziffern 0 - 9 und Buchstaben A - F)
- Jede Variable, die größer als FFFF (Hex) ist.

Das Korrigieren eines Fehlers

Drücken Sie einfach die 'Backspace' - Taste, wenn Sie einen Fehler beim Tippen bemerken.

Die komfortablen Editiermöglichkeiten auf dem Bildschirm ermöglichen es Ihnen, zurück zu gehen und eine fehlerhafte Zeile erneut einzugeben, vorausgesetzt, sie ist noch auf dem Bildschirm. Bewegen Sie einfach den Cursor über den Bildschirm und verbessern die Zeile mit den Tasten zur Cursor- Steuerung, einschließlich Löschen und Einfügen von Zeichen, falls erforderlich. Dann drücken Sie die 'Enter'-Taste, um die Zeile einzugeben.

Genaueres über die Editiermöglichkeiten finden Sie auf den folgenden Seiten.

Editiermöglichkeiten auf dem Bildschirm

Diese Möglichkeiten sind sowohl mit der erweiterten Tastatur als auch mit der Tastatur des NASCOM 1 verfügbar.

<u>NAME</u>	<u>HEX-Code</u>	<u>Gedrückte Tasten</u>	<u>Funktion</u>
NULL	ØØ	CTRL/SHIFT/	wird ignoriert
BS	Ø8 ↴	BACKSPACE	eine Stelle zurück, letztes Zeichen wird gelöscht
LF	ØA	LF oder CTRL/J	wird auf dem Bildschirm nicht dargestellt
FF	ØC	CS oder SHIFT/BS	löscht Bildschirm und setzt den Cursor nach oben links
CR	ØD	ENTER oder NEWLINE	Eingabe der Zeile, Cursor wird an den Anfang der nächsten Zeile gesetzt, "Scrolling" wird durchgeführt, wenn der Bildschirm voll ist
CUL	11 ←	Pfeil links oder CTRL/Q	Cursor nach links
CUR	12 →	Pfeil rechts oder CTRL/R	Cursor nach rechts
CUU	13 ↑	Pfeil oben oder CTRL/S	Cursor nach oben
CUD	14 ↓	Pfeil unten oder CTRL/T	Cursor nach unten
CSL	15	SHIFT/Pfeil links oder CTRL U	lösche das Zeichen auf der Cursor Position und bewege den Rest der Zeile nach links
CSR	16	SHIFT/Pfeil rechts oder CTRL/V	bewege den Rest der Zeile nach rechts
CH	17	CH oder CTRL/W	setze den Cursor auf die erste Stelle der Zeile
CCR	18	CTRL/X	wenn der Cursor am Anfang einer Zeile ist wird die Eingabe ignoriert, andernfalls wird CR ausgeführt
ESC	1B	ESC oder SHIFT/ENTER	lösche die Zeile, setze den Cursor an den Anfang der Zeile

Die Tastatur

Mit der erweiterten Tastatur ist es möglich, alle 256 verschiedenen 8 - Bit - Codes zu erzeugen:

Hex 20 bis 5F sind die normalen ASCII - Zeichen, die Tasten sind entsprechend gekennzeichnet.

Buchstaben werden als Großbuchstaben dargestellt. Wenn die Shift- Taste gleichzeitig gedrückt wird, werden Kleinbuchstaben ausgegeben (s.a. 'K' - Befehl). Um ein @ mit der @ - Taste zu erzeugen, muß die Shift- Taste gedrückt werden.

Sowohl die CTRL - Taste als auch die @ - Taste arbeiten als CTRL - Taste. Damit ist es möglich, auch mit der Tastatur die des NASCOM 1 die Editermöglichkeiten zu nutzen. Wenn eine dieser beiden Tasten gedrückt wird, wird Bit 6 verändert. Dies ergibt die Codes 00 bis 1F und 60 bis 7F.

Die Grafik- Taste setzt das Bit 7, damit ist es möglich, mit der NASCOM 2 Grafik - Option Grafikzeichen direkt auf den Bildschirm zu schreiben. Mit der Grafik- Taste zusammen mit den anderen Tasten der Tastatur können die Codes 80 bis FF erzeugt werden (s.a. 'K' - Befehl).

Die Reset- Taste

Die Reset- Taste unterscheidet sich stark von den anderen Tasten der Tastatur. Sie sendet ein Rückstell- Signal direkt an die CPU.

Betätigt man die Reset- Taste, so wird das Betriebssystem neu initialisiert. Oben links erscheint auf dem Bildschirm "NAS-SYS 1", und der Computer erwartet Befehle.

Das Scrolling

Wenn die Reset- Taste gedrückt wird oder der Bildspeicher gelöscht wird (Clear Screen), wird der Cursor automatisch in die linke obere Ecke gesetzt. Wenn nach Eingabe mehrerer Zeilen die unterste Zeile erreicht ist, wird der Inhalt des Bildschirmspeichers automatisch Zeile um Zeile nach oben geschoben. Es werden also immer die letzten 15 Zeilen angezeigt.

Die oberste Zeile wird niemals gescrollt, dadurch ist sie besonders für Überschriften geeignet.

Genaue Beschreibung der System-Befehle

Axxxx yyyy Hexadezimal-Arithmetik

Dieser Befehl verarbeitet einfache hexadezimale Arithmetik.
Drei Befehle werden angezeigt:

SSSS DDDD JJ

SSSS ist die Summe der beiden Werte xxxx und yyyy.

DDDD ist die Differenz dieser Werte.

JJ ist das Displacement für einen relativen Sprung von einem Befehl, dessen erstes Byte auf der Speicherstelle XXXX steht, zur Speicherstelle yyyy. Wenn ein relativer Sprung nicht möglich ist, weil beide Adressen zu weit auseinander liegen, wird anstatt JJ ?? angezeigt.

Bxxxx Breakpoint - Befehl

Mit dem Breakpoint - Befehl wird die Adresse eingegeben, an der der Programmablauf unterbrochen werden soll.

Anfangs ist der Breakpoint abgeschaltet, da er bei Reset gelöscht wird. Der Breakpoint - Befehl kann jederzeit gegeben werden. Die Breakpoint-Adresse wird abgespeichert. Wenn ein Befehl zur Programmausführung gegeben wird, wird in die Speicherstelle mit dieser Adresse der Code E7H geladen. Wenn dieser Befehl (E7H = RST 20) während der Ausführung des Programms erreicht wird, werden die Register der CPU automatisch gerettet und angezeigt. Der ursprüngliche Code wird dann wieder in die Speicherstelle geladen. Jetzt kann jeder Betriebssystem-Befehl gegeben werden, um beispielsweise das Programm zu verändern. Wenn ein Befehl zur Ausführung des Programms ('E' oder 'S') gegeben wird, wird die Adresse zunächst abgearbeitet, auf die der Breakpoint zeigt, und es wird erst dann der Programmablauf wieder unterbrochen, wenn diese Speicherzelle erneut erreicht wird.

Beachten Sie, daß der ursprüngliche Wert wieder eingefügt wird, gleichgültig, auf welche Weise das Programm beendet wird (s.a. "Wie ein Programm beendet werden kann").

Denken Sie daran, daß der Breakpoint immer auf das erste Byte eines Befehls zeigen muß.

Übrigens ist es nicht möglich, eine Programmunterbrechung in einem Programm durchführen zu lassen, die in ROM gespeichert ist.

Cxxxx yyyy zzzz Copy Befehl (Kopieren)

Der Kopierbefehl überträgt einen Datenblock der Länge zzzz von dem Adressbereich xxxx zum Adressbereich yyyy. Die Daten werden Byte - weise nacheinander übertragen, beginnend bei Adresse xxxx. Deshalb ist es möglich, daß Daten zerstört werden, wenn die Adressbereiche überlappen.

Der Befehl kann auch benutzt werden, um einen Adressbereich mit mit einem einzigen Wert zu füllen. Dieser Wert muß auf der Speicherstelle xxxx stehen, yyyy ist um eins größer als xxxx, zzzz gibt dann die Länge des Bereiches an.

Exxxx Execute Befehl (Ausführen)

Mit dem Execute - Befehl wird ein Programm, daß bei Adresse xxxx beginnt, ausgeführt.

Wenn xxxx nicht angegeben wird, wird der gespeicherte Programmzähler benutzt. Um zum Beispiel ein Programm nach einem Breakpoint fortzusetzen, muß einfach E eingegeben werden.

Gxxxx yyyy zzzz Generate Befehl (Cassette erzeugen)

Der Generate - Befehl dient dazu, eine Cassette zu erzeugen, die, wenn sie wieder eingelesen wird, selbstständig lädt und das Programm automatisch ausführt.

Daten von der Adresse xxxx werden bis zur Adresse yyyy (nicht einschließlich) auf die Cassette geladen, zzzz gibt die Adresse an, an der das Programm nach dem Einlesen gestartet werden soll.

Starten Sie den Cassetten- Recorder, bevor Sie diesen Befehl eingeben. Die Leuchtdiode ist während der Ausgabe des Programms eingeschaltet.

Wenn die Cassette wieder eingelesen wird, ist es nicht nötig, irgendeinen Befehl zu geben. Starten Sie einfach den Cassetten- Recorder und schalten Sie ihn wieder ab, wenn die Ausführung des eingelesenen Programms begonnen hat.

Die Daten werden im folgenden Format auf die Cassette geschrieben:

(CR) EØ (CR) R (CR)
dann folgen die Daten im gleichen Format wie mit dem Write- Befehl,
danach folgt Ezzzz (CR) .

H

Halb - Duplex Terminal

Der H - Befehl führt ein sehr einfaches Programm in NAS - SYS aus, das auf die Eingabe von Daten wartet und diese ausgibt. Das Programm kann nur mit der Reset - Taste verlassen werden.

Um aus dem NASCOM ein Halb- Duplex Computer- Terminal zu machen, das mit dem ASCII - Code arbeitet, geben Sie zunächst X 30 und danach den H - Befehl ein.

Ixxxx yyyy zzzz Intelligent Copy Befehl (Intelligentes Kopieren)

Dieser Befehl ist gleich dem Copy - Befehl, insoweit Datenblocks der Länge zzzz von xxxx nach yyyy übertragen werden.

Er ist jedoch "intelligent" genug, an dem Ende anzufangen, das erforderlich ist, um überlappende Speicherbereiche nicht zu zerstören.

J

Jump - Befehl (Springe zum BASIC)

Mit dem Jump - Befehl wird ein Programm gestartet, daß ab der Speicherstelle FFFAH steht. Normalerweise ist das die Adresse für den "Kaltstart" für das 8K - BASIC. "Kaltstart" heißt, daß ein bestehendes BASIC - Programm ~~████████~~ gelöscht wird.
evtl.

K xx

Keyboard - Befehl (Tastatur)

Mit dem Keyboard - Befehl wird die Arbeitsweise der Tastatur festgelegt.

K0 normale Arbeitsweise, die nach Reset eingeschaltet wird.

K1 kehrt die Funktion der Shift - Taste für Buchstaben um.

K4 kehrt die Funktion der Graphik - Taste um.

K5 funktioniert wie K 1 und K 4 zusammen.

L

Load - Befehl (Lochstreifen einlesen)

Der Load - Befehl wird benutzt, um Daten von einem Lochstreifeneser einzugeben. Die Daten müssen in dem Format sein, das benutzt wird, wenn mit dem Tabulate - Befehl Lochstreifen gestanzt werden.

Die eingelesenen Daten werden von NAS - SYS sorgfältig auf mögliche Übertragungsfehler überprüft. Jede Zeile, die einen Fehler enthält, wird auf dem Bildschirm nach oben geschoben, sodaß sie verbessert werden kann, und dann erneut eingegeben. Alle fehlerfreien Zeilen werden kurz auf dem Bildschirm angezeigt und verschwinden dann.

Mxxxx

Modify Befehl (Speicher verändern)

Mit dem Modify - Befehl können Speicherstellen, beginnend mit xxxx überprüft und verändert werden.

Die jeweilige Adresse wird angezeigt, gefolgt vom gegenwärtigen Wert. Dieser Wert kann verändert werden, und wenn die Zeile eingegeben wird, wird dieser neue Wert auf der Adresse eingefügt. Mehrere Daten können in einer Zeile eingegeben werden, damit werden die angezeigte und die entsprechend folgenden Adressen verändert.

Wenn Sie den Modify - Mode beenden wollen, geben Sie am Ende einer Zeile '..' ein.

Geben Sie ':' ein um zu der vorhergehenden Adresse zurück zu gehen.

Geben Sie '/yyyy' am Ende einer Zeile ein, um den Modify- Mode ab Adresse yyyy fortzusetzen.

Normalerweise müssen Hexadezimalzahlen eingegeben werden, es ist jedoch auch möglich, ASCII - Zeichen direkt in den Speicher zu schreiben. Dazu muß ein Komma, gefolgt von dem jeweiligen ASCII - Zeichen eingegeben werden. Zum Beispiel würde ',A,B,C' die Zeichen ABC im Speicher ablegen.

Fehlerhafte Ausdrücke werden erkannt und mit einer Fehlermeldung quittiert. In diesem Fall wird der Modify - Mode automatisch mit der ersten Adresse der letzten Zeile fortgesetzt.

N

Normal - Befehl

Mit diesem Befehl werden die Zeiger auf die Ein- und Ausgabetafeln auf ihre normalen Werte zurückgesetzt. Damit werden also die X- oder U- Befehle ausgeschaltet.

Alle Eingaben müssen dann also von der Tastatur oder der seriellen Schnittstelle kommen, alle Ausgaben werden auf dem Bildschirm dargestellt.

Oxx yy

Output - Befehl (Ausgabe)

Der Output - Befehl sendet Daten zu einer Schnittstelle (Port). Der Wert yy wird zu dem Port mit der Adresse xx gesendet. Mit 'OØ7 ØF' würde also der Wert ØFH zur Schnittstelle Ø7H geschickt werden. Über den Umgang mit den parallelen Schnittstellen informiert Sie das PIO Technical Manual.

Q

Query - Befehl (Eingabe)

Damit wird ein Datum von einer Schnittstelle eingegeben und hexadezimal auf dem Bildschirm angezeigt. Die Port- Adresse ist xx. (s.a. PIO Technical Manual).

R

Read - Befehl (Cassette lesen)

Der Read - Befehl liest eine Cassette, die mit dem Write- Befehl geschrieben wurde. Das verwendete Format finden sie in der Erläuterung des Write- Befehls.

Während die einzelnen Datenblöcke gelesen werden, wird auf dem Bildschirm SSSS BBLL angezeigt. SSSS bezeichnet die Startadresse der Daten, BB ist die Nummer des Blocks, LL die Blocklänge (Ø entspricht der Blocklänge 256).

Wenn der letzte Datenblock, Block ØØ, fehlerfrei gelesen wurde, wird der Read- Befehl automatisch beendet. Während der Ausführung des Befehls ist die LED eingeschaltet.

Der Start eines jeden Blocks wird durch vier Startzeichen erkannt. Alle Daten werden bis zum Lesen dieser Startzeichen ignoriert. Wenn die Prüfsumme für die führenden Daten nicht richtig ist, wird ein Fragezeichen angezeigt und das Programm wartet auf den Start des nächsten Blocks. Die folgenden Daten werden nicht eingelesen.

Wenn die Prüfsumme der gerade gelesenen Daten falsch ist, wird anstelle eines Punktes ein Fragezeichen angezeigt, und das Programm wartet auf den Start eines neuen Blocks. In diesem Falle wurden fehlerhafte Daten im Speicher abgelegt. Aber diese Technik erlaubt es, auf einen Daten - Puffer zu verzichten.

Während eine Cassette gelesen wird, dürfen keine Tasten auf der Tastatur gedrückt werden, was Fehler verursachen würde, die allerdings erkannt werden.

Es ist nötig, während dem Einlesen den Bildschirm zu beobachten, um festzustellen, ob alle Datenblöcke richtig gelesen wurden. Wenn Fehler festgestellt werden, spulen Sie einfach die Cassette um ca. zwei Blöcke zurück und starten erneut.

Um den Lesebefehl abzubrechen, drücken Sie viermal 'ESC' (Shift/Enter). Dies funktioniert nur zwischen zwei Blöcken, deshalb drücken Sie nötigenfalls die Tasten, bis das Ende eines Blocks erreicht ist.

Sxxxx

Single Step - Befehl (Einzelschritt)

Der Single Step - Befehl führt einen einzigen Befehl eines Maschinen - Programms aus, der ab Adresse xxxx steht.

Wenn xxxx nicht angegeben wird, wird der gespeicherte Programmzähler benutzt. Um nach einem Breakpoint einen einzigen Schritt auszuführen, ist es nur erforderlich, 'S' einzugeben.

Wenn der vorhergehende Befehl schon 'S' war, ist es noch nicht einmal nötig, 'S' einzugeben.

Wenn der Maschinen- Befehl ausgeführt wurde, werden die Register der CPU gerettet und angezeigt.

Beachten Sie, daß es nicht möglich ist, einzelne Teile von NAS - SYS im Einzelschritt- Betrieb abzuarbeiten, einschließlich der Unterprogramme zur Ein/Ausgabe - Steuerung. Es ist empfehlenswert, direkt nach dem Aufruf eines Unterprogramms von NAS - SYS einen Breakpoint zu setzen und die Unterprogramme

mit 'E' ausführen zu lassen. Man kann dann mit Single Step fortfahren, um das Benutzerprogramm auszuführen.

Sie können davon ausgehen, daß die Unterprogramme in NAS- SYS selbst keine Fehler enthalten.

Txxxxx yyyy zzzz Tabulate - Befehl (Datenausgabe)

Mit diesem Befehl wird ein Datenblock von der Adresse xxxx bis zur Adresse yyyy (nicht einschließlich) angezeigt. Jede Zeile zeigt die Adresse gefolgt von acht Daten- Bytes. zzzz gibt die Anzahl der Zeilen an, die auf dem Bildschirm dargestellt werden sollen. Drücken Sie irgend eine Taste, um die nächste Zeilen - Gruppe anzeigen zu lassen, oder drücken Sie 'ESC' oder 'Shift /Enter', um den Befehl abzubrechen.

Wenn zzzz gleich Ø ist, werden alle Daten ohne Pause angezeigt. Wenn Sie den X Ø - Befehl und danach den Tabulate-Befehl mit zzzz gleich Ø eingeben, wird ein fortlaufender Datenblock zur seriellen Schnittstelle gegeben, z.B. um einen Lochstreifen zu stanzen. Am Ende einer jeden Zeile wird eine Prüfsumme ausgegeben. Mit dem Load- Befehl kann der Lochstreifen wieder gelesen werden.

U User I/O - Befehl (Ein/Ausgabe für Benutzer)

Dieser Befehl aktiviert die vom Benutzer angegebenen Ein/Ausgabe- Programme, zusätzlich zur normalen Eingabe über die Tastatur und die Ausgabe über den Bildschirm. Damit ist es z.B. möglich, einen Drucker über die Parallel- Schnittstelle zu betreiben. Der 'N' - Befehl kann gegeben werden, um die Unterprogramme wieder abzuschalten.

Die Adresse ØC78H muß auf die Ausgabe- Programme zeigen, die Adresse ØC7BH auf die Eingabe- Programme. Bei Reset zeigen diese Zeiger auf einen Return - Befehl innerhalb NAS - SYS, deshalb hat die Eingabe des U - Befehls keine Folgen, wenn diese Adressen nicht geändert wurden.

Der U - Befehl wird während Load, Read, Write und Generate automatisch abgeschaltet und später wieder eingeschaltet.

V Verify - Befehl (Überprüfe)

Der Verify - Befehl ist mit dem Read - Befehl identisch, jedoch werden die Daten, die von der Cassette gelesen werden, nicht im Speicher abgelegt. Der Sinn dieses Befehls ist es, zu überprüfen, ob eine Cassette fehlerfrei eingelesen werden kann.

Dabei wird der Inhalt des eingelesenen Bandes mit dem entsprechenden Speicherinhalt des NASCOM - Systems verglichen.

Wxxxx_yyyy

Write - Befehl

(Schreibe Cassette)

Mit diesem Befehl werden Daten auf einer Cassette gespeichert. Daten von der Adresse xxxx bis zur Adresse yyyy (nicht einschließlich) werden an das Cassetten - Interface gegeben.

Die Daten werden in Blöcken zu je 256 Bytes ausgegeben, mit Ausnahme des letzten Blocks, der kürzer sein kann. Das Format ist wie folgt:

ØØ	NULL
FF FF FF FF	Vier Block - Start- Zeichen (FFH)
SS SS	Startadresse (niederwertiges Byte zuerst)
LL	Länge des Blocks ($\emptyset = 256$)
BB	Blocknummer, in absteigender Reihenfolge, ØØ ist der letzte Block
CC	Prüfsumme für die Präambel
DD DD ...	Daten
EE	Prüfsumme für die Daten
00 00 00 00	
00 00 00 00	
00 00	zehn mal NULL

Während ein Block geschrieben wird, werden die führenden Daten bestehend aus Start- Adresse, Blocknummer und Länge in folgendem Format angezeigt

SSSS BBLL .

Wenn der Befehl gegeben wird, wird die LED eingeschaltet und nach einer kurzen Verzögerung 256 mal NULL ausgegeben. Dann werden nacheinander die Blöcke gesendet. Wenn der Befehl abgeschlossen ist, wird die LED wieder abgeschaltet.

Die eingefügten NULLEN am Ende eines jeden Blocks stellen sicher, daß, auch wenn mehrere Daten im vorhergehenden Block verloren gegangen sind, der nächste Block wieder einwandfrei gelesen werden kann. Die zusätzliche NULL vor dem Start eines Blocks stellt sicher, daß ein zufälliges Block- Startzeichen ignoriert wird. Die 256 NULLEN am Anfang ermöglichen es, daß eine Korrektur fehlerhaft gelesener Blöcke durch Zurückspulen der Cassette und erneutes Einlesen möglich ist.

Xxx

External - Befehl (Externe Geräte)

Mit dem External - Befehl werden Ein/Ausgabe - Programme aktiviert, die in NAS - SYS enthalten sind. Damit sind komfortable Möglichkeiten gegeben, mit externen Geräten wie z.B. ASCII - Terminals oder anderen Computern über die serielle Schnittstelle zu verkehren. Mit dem Normal - Befehl werden diese Unterprogramme wieder abgeschaltet.

Mit der Eingabe des Wertes xx nach X können folgende Möglichkeiten gewählt werden:

- Ø Unterstützung eines Terminals im Voll - Duplex - Modus. Dabei wird jedes erhaltene Zeichen zurück an das Terminal gesendet. LF wird automatisch nach CR ausgeführt. Alle Ausgaben erfolgen mit gerader Parität.
 - 1Ø Gleich wie Ø, aber LF wird nicht unterstützt.
 - 2Ø Gleich wie Ø, aber Halb - Duplex - Modus. Die erhaltenen Zeichen werden nicht zurück gesendet. Jedoch wird LF immer nach CR ausgeführt
 - 3Ø Gleich wie 2Ø, aber LF wird nicht ausgeführt. Diese Option macht den NASCOM zu einem Halb - Duplex - Terminal.
- 1,11,21,31 Gleich wie Ø, 1Ø, 2Ø und 3Ø, aber die Parität der Ausgangssignale ist jetzt ungerade.

(Die Parität der Eingangsdaten wird immer ignoriert.)

Um Daten auf Lochstreifen zu speichern, benutzen Sie den X - Befehl (gewöhnlich Option Ø) und dann den Tabulate - Befehl. Benutzen Sie den L - Befehl, um die Daten wieder einzulesen.

Wenn ein Zeichen von einem Terminal empfangen wird (außer ESC (1BH), NULL (Ø), oder DEL (7FH), wird angenommen, daß das Programm dieses Zeichen ausgeben will. Deshalb wird ein Indikator gesetzt, der bewirkt, daß das nächste vom Programm ausgegebene Zeichen nicht an den seriellen Ausgang gegeben wird.

Um einem Terminal "Backspace" einzugeben, daß diese Taste nicht hat, benutzen Sie CTRL/H.
Bei 'L','R','W','G' wird der X- Befehl automatisch abgeschaltet und nach Abarbeitung des Befehls wieder aktiviert.
Mit dem X - Befehl werden auch die 'U' - Ausgabe- Programme aktiviert, sofern sie vorhanden sind. Damit ist es sehr einfach, den NASCOM, einen Fernschreiber und einen Drucker mit paralleler Schnittstelle gleichzeitig zu betreiben.

Z

Z - Befehl

Mit dem Z - Befehl wird ein Sprung zur Adresse FFDH durchgeführt. Dies ist die "Warmstart" - Adresse des 8K - BASIC. Beim "Warmstart" wird ein bereits eingelesenes BASIC - Programm nicht gelöscht.

Wie man ein Programm eingibt und testet

1. Schreiben Sie das Programm auf (in der Z80 - Assembler - Sprache). Dazu benötigen Sie das Z80 - Programmierhandbuch und die Teile dieser Beschreibung, die die Möglichkeiten der NAS - SYS - Unterprogramme zur Ein/Ausgabe und der Beendigung eines Programms erläutert. Es gibt dazu noch eine Reihe andere nützliche Unterprogramme in NAS - SYS, die Sie für Ihr Programm verwenden können.
2. Jetzt können Sie Ihr Programm von Hand assemblyieren. Das heißt, daß Sie jeden Befehl des Quelltextes in Maschinen-Code übersetzen müssen. Eine Tabelle dazu finden Sie auch im Z80 - Programmierhandbuch. Die Arithmetik in NAS - SYS erlaubt es, die benötigten Werte für relative Sprünge zu ermitteln.
3. Wenn Sie über ZEAP, das ist der Editor/Assembler von NASCOM, verfügen, können Sie Ihr Programm in der symbolischen Assembler- Sprache eingeben, ZEAP erzeugt dann den Maschinen-Code.
4. Wenn Sie ZEAP nicht haben, benutzen Sie den Modify- Befehl, um das Programm einzugeben.
5. Es ist immer möglich, daß Sie beim Eingeben oder Schreiben des Programms einen Fehler gemacht haben, der möglicherweise jeden Speicherbereich verändern kann. Sie sollten deshalb das Programm, bevor Sie es ausführen lassen, auf eine Cassette retten. Benutzen Sie dazu den Write - Befehl. Wenn Sie nicht ganz sicher sind, daß Cassette und Recorder fehlerfrei sind, benutzen Sie den Verify - Befehl, um zu testen, ob das Programm wieder eingelesen werden kann.
6. Lassen Sie das Programm jetzt mit dem Execute - Befehl ausführen. Um das Programm zu testen, können Sie einen Breakpoint setzen oder den Single Step - Befehl benutzen. Mit Hilfe des Tabulate- oder Modify- Befehls können Sie sich auch Speicherbereiche ansehen, die vom Programm benutzt werden, nachdem Sie einen Einzelschritt ausgeführt haben oder das Programm mit einem Breakpoint unterbrochen wurde.
7. Wenn das Programm "wild" läuft, können Sie jederzeit die Reset- Taste benutzen, um wieder Kontrolle über den Computer zu erhalten. Wenn dabei der Programmspeicher verändert wurde, können Sie das Programm wieder von der Cassette laden.
8. Auf den ersten Blick scheint Assembler - oder Maschinensprachen- Programmierung schwierig und zeitaufwendig zu sein, aber sie birgt doch eine gewisse Faszination und gibt die Möglichkeit, alle Vorteile der Maschine zu nutzen. Programme in Maschinensprache sind in der Regel außerordentlich schnell und tragen sehr zum Verständnis der Programmierung und des Computers bei.

Viel Glück!

Anzeige der CPU - Register

Nach einem Breakpoint oder im Einzelschrittbetrieb oder wenn der Code E7H (RST 20H) in einem Programm ausgeführt wurde, werden die Register der CPU wie folgt angezeigt:

-SP- -PC- -AF- -HL- -DE- -BC- -I- -IX- -IY- -Flags-

Die "Flags" sind eine dekodierte Darstellung des F- Registers. Folgende Buchstaben werden angezeigt, abhängig davon, welche Flags (Flaggen, Marken) gesetzt worden sind:

- S Z H P N C -

Die Anzeige der CPU - Register ist häufig eine wichtige Hilfe bei der Fehlersuche in einem Programm. Der Programmzähler (PC) zeigt die nächste Adresse an, die ausgeführt werden soll, der Stapelzeiger gibt die Adresse des Stapspeichers an. Alle anderen Register werden so angezeigt, wie sie vom Programm verändert worden sind. Wenn die Register angezeigt worden sind, ist es oftmals nötig, die Wirkungen des Programms noch genauer zu betrachten, z.B. dadurch, daß man sich mit dem Modify- oder Tabulate- Befehl verschiedene Speicherbereiche ansieht. So ist es möglich, den Inhalt des Stapel- Speichers (Stack) oder die Speicherstelle, auf die HL zeigt, anzusehen.

Einfache Anweisungen für die Ein/Ausgabe

NAS - SYS benutzt eine sehr effektive Methode für Ein/Ausgabeoperationen. Dies wird in dem Abschnitt 'Ein/Ausgabe' erläutert.

Es ist jedoch sehr einfach, den Inhalt des Register A auf dem Bildschirm anzeigen zu lassen. Fügen Sie dazu einfach den Befehl F7H (RST 30H) in Ihr Programm ein.

Wenn das Programm die Eingabe eines Zeichens auf der Tastatur abwarten soll, setzen Sie den Befehl CFH (RST 08) in das Programm. Das Unterprogramm in NAS - SYS übergibt das Zeichen von der Tastatur im Register A an Ihr Programm.

In beiden Fällen werden keine anderen Register beeinflußt.

Selbst wenn Sie nur diese beiden sehr einfachen Methoden für die Ein/Ausgabe verwenden, können Sie mit Hilfe der Monitor-Befehle 'X' oder 'U' schon ein Terminal oder einen Drucker steuern. (s.a. Beschreibung dieser Betriebssystem- Befehle)

Wie man ein Programm beendet

Eine der folgenden Methoden sollte für die Beendigung eines Programms verwendet werden - und keine andere!

1. Drücken der Reset - Taste, um das System wieder zu starten. Der HALT - Befehl (76H) kann an das Ende eines Programms eingefügt werden, die HALT - LED leuchtet dann, um anzudeuten, daß das Programm beendet wurde. Mit der Reset- Taste gewinnt man dann wieder Kontrolle über das System. Das ist allerdings eine etwas primitive Methode.
2. Ausführen von RST Ø in einem Programm. Der Befehl RST Ø (C7H) hat die gleiche Wirkung, wie das Drücken der Reset- Taste, es wird nämlich ein Sprung zur Adresse ØØØØ durchgeführt. Beide Methoden haben jedoch den Nachteil, daß der Bildschirm gelöscht wird, sodaß Sie nicht mehr sehen können, was das Programm zuletzt ausgegeben hat.
3. Die normale und empfehlenswerteste Methode ist die Einfügung von DF 5B (Hex) in das Programm. Mit diesem Befehl wird kontrolliert in das Betriebssystem gesprungen, sodaß Monitor-Befehle gegeben werden können. (Die Register werden nicht gerettet und der Stapelzeiger (SP) wird auf 1ØØØH zurückgesetzt.)
4. Wenn Sie sicherheitshalber ein Programm an einer Stelle beenden wollen, die das Programm eigentlich gar nicht erreichen dürfte, verwenden Sie den Befehl RST 2Ø (E7H). Dieser Befehl rettet die Programm - Register in den Stapspeicher und zeigt sie auf dem Bildschirm an, bevor die Kontrolle an NAS - SYS abgegeben wird. Dies kann sehr nützlich sein für die Fehlersuche in einem Programm. Das Programm kann dann mit dem Execute - oder dem Single Step - Befehl fortgesetzt werden, es muß dann jedoch angegeben werden, ab welcher Adresse fortgesetzt werden soll.
5. Mit entsprechender Hardware ist es möglich einen nicht-maskierbaren Interrupt (Programmunterbrechung, NMI) auszuführen. Dies hätte den gleichen Effekt wie 4., allerdings muß dann keine Adresse zur Fortsetzung des Programms angegeben werden.

NAS - SYS Restart - Befehle und Unterprogramme

NAS - SYS beinhaltet sehr viele nützliche Unterprogramme, die der Benutzer für eigene Programme verwenden kann.

1. Restart - Unterprogramme. Diese Funktionen sind alle mit den 1 - Byte - Z80 - Restart - Befehlen zugänglich.
2. Unterprogramme, die durch spezielle 2 - Byte - Befehle aufgerufen werden. Diese Befehle bestehen aus einem RST 18H (DFH) - Befehl, gefolgt von einer Nummer, die das Unterprogramm bezeichnet. NAS - SYS - Unterprogramme sollten immer mit dieser Methode aufgerufen werden und niemals mit einem CALL - Befehl, denn wenn NAS - SYS einmal verändert werden sollte, werden die Nummern der Unterprogramme bestehen bleiben, aber die absoluten Adressen wären möglicherweise verändert.
3. Es gibt eine Routine, die mit einem CALL - Befehl aufgerufen werden muß. Diese Routine erlaubt es, die NAS - SYS - Initialisierung vorzunehmen, ohne die Kontrolle an NAS - SYS abzugeben.

Die Unterprogramme jeder dieser Kategorien wird nun ausführlich beschrieben, mit Ausnahme der Ein/Ausgabe - Programme, die im nächsten Abschnitt erläutert werden.

NAS - SYS RESTART - Befehle

Code	Assembler	Name	Funktion
C7	RST Ø	START	Reset, initialisiere NAS - SYS
CF	RST 8	RIN	Übergebe ein eingegebenes Zeichen mit dem A - Register
D7	RST 1ØH	RCAL	Relativer Programmaufruf. Nach dem Code wird das Displacement (Offset) zum gegenwärtigen Stand des Befehlszähler angegeben. Das funktioniert also ähnlich wie die relativen Sprünge der Z80 - CPU, und erlaubt das Schreiben von verschiebbaren (Adress - unabhängigen) Programmen.
DF	RST 18H	SCAL	Unterprogramm - Aufruf, gefolgt von diesem Code wird die Nummer des Unterprogramms angegeben, das in NAS - SYS aufgerufen werden soll. (s. nächster Abschnitt)
E7	RST 2ØH	BRKPT	Rette die Programm- Register und zeige sie an, dann übergib die Kontrolle an NAS - SYS. Dies wird von dem Breakpoint - Befehl benutzt
EF	RST 28H	PRS	Gib die diesem Code folgende Zeichenkette aus bis ØØ gefunden wird, dann fahre mit dem folgenden Code fort. Dies ist eine sehr einfache Methode, Text auzugeben. Das A - Register wird verändert.
F7	RST 3ØH	ROUT	Gebe das Zeichen aus dem A - Register aus.
FF	RST 38H	RDEL	Warte eine Zeit, die abhängig vom Wert des A-Reg. ist. A wird ØØ.

NAS - SYS Unterprogramme

Diese Routinen werden einfach durch den angegebenen Code aufgerufen. Um z.B. den Text 'Error' (Fehlermeldung) auszugeben, setzen Sie einfach den Code DF 6B in Ihr Programm, womit das Unterprogramm ERRM aufgerufen wird.

Es ist ebenso möglich, die NAS - SYS - Betriebssystembefehle aufzurufen. Dazu dient der Code DF gefolgt von dem ASCII - Code des Buchstabens des Befehls. Die Register HL, DE und BC sollten die (bis zu drei) Variablen enthalten, die normalerweise nach dem Befehl eingegeben werden. Der Code DF 57 würde zum Beispiel den NAS - SYS - Write - Befehl aufrufen. Dazu müßte vorher die Startadresse des zu schreibenden Datenblocks in HL gegeben werden, DE müßte die auf die Daten folgende Adresse erhalten. Sie sollten den Assembler - Ausdruck von NAS - SYS im Anhang studieren, um zu verstehen, wie diese Befehle arbeiten.

Möglichlicherweise wollen Sie Ihren eigenen Satz von Befehlen und Unterprogrammen definieren oder Funktionen hinzufügen, die nicht in NAS - SYS enthalten sind. Dies ist möglich, da der Start der Tabelle, die die Unterprogramm - Adressen enthält, in den Adressen ØC71 - ØC72 gespeichert ist. Sie können den Inhalt dieser Adressen so verändern, daß er auf Ihre eigene Unterprogramm - Tabelle zeigt. Auch dazu sollten Sie sich den Assembler - Ausdruck anschauen.

Vor der Liste der Unterprogramme, die Sie aufrufen können, noch die Beschreibung eines besonderen Befehls, den Sie mit dem Code CD ØD ØØ (CALL ØØØDH). Dieses Unterprogramm, daß den Namen STMON trägt, initialisiert NAS - SYS und löscht den Bildschirm. Wie Sie wissen, ist es möglich, mit dem NASCOM 2 verschiedene Adressen einzustellen, die nach einem Reset angesprungen werden. Das Programm auf dieser Adresse kann also den Stapelzeiger setzen und dann STMON aufrufen. Es ist dann mögliche, alle NAS - SYS - Unterprogramme aufzurufen.

Nun folgt die Liste der NAS - SYS - Unterprogramme. Es wird leicht für Sie sein, diese Unterprogramme mit Ihren Programmen aufzurufen, was Ihnen eine Menge Arbeit bei der Programm-Entwicklung spart.

<u>Code</u>	<u>Name</u>	<u>Funktion</u>
DF 5B MRET		Dies ist kein normales Unterprogramm, sondern wird benutzt, um ein Programm zu beenden und die Kontrolle an NAS - SYS zu übergeben. (s.a. 'Wie man ein Programm beendet')
DF 5C SCALJ		Auch dies ist kein normales Unterprogramm, sondern wird benutzt, um ein Unterprogramm aufzurufen, dessen Adresse beim Programmieren noch nicht bekannt ist. Die Adresse dieses Unterprogramms muß auf den Adressen ARGC ($\emptyset C\emptyset A$) gespeichert sein, bevor der Code DF 5C ausgeführt wird. Das Unterprogramm wird dann aufgerufen.
DF 5D TDEL		Warte ca. 1 Sekunde (bei 4 MHz). Register A und E werden $\emptyset\emptyset$.
DF 5E FFLP		Schalte ein oder mehrere Bits im Ausgabekanal \emptyset , dann schalte sie sofort wieder zurück. Für jedes Bit, das geschaltet werden soll, muß in A ein entsprechendes Bit gesetzt sein. A wird verändert.
DF 5F MFLP		Verändere (schalte ein oder aus) den Zustand der DRIVE - LED. A wird verändert.
DF 60 ARGS		Lade den Inhalt von ARG1 in HL, ARG2 in DE und ARG3 in BC. ARG 1, 2 und drei sind die ersten drei Werte, die nach einem Betriebssystem - Befehl eingegeben werden. Sie können damit sehr einfach zusätzliche Variable als Teil Ihres Programms eingeben, wenn Sie das Programm mit dem 'E' - Befehl starten.
DF 62 IN		Abfrage auf Zeichen von der Tastatur. Es wird jedoch nicht wie mit dem Code CF (RST 8) auf eine Eingabe gewartet sondern einfach nachgesehen, ob eine Eingabe gemacht worden ist. Wenn dies der Fall ist, wird das Carry - Flag gesetzt und das Zeichen steht in Register A. Register A wird verändert.
DF 63 INLIN		Übernehme eine eingegebene Zeile. Diese Routine wird von NAS - SYS benutzt, um Befehle zu erhalten. Sie liefert den blinkenden Cursor und wartet auf die Eingabe von 'Enter' oder 'New Line'. Das DE - Register zeigt auf die Speicherstelle, an der der Cursor war, bevor die Zeile eingegeben wurde. Register A wird verändert.

<u>Code</u>	<u>Name</u>	<u>Funktion</u>
DF 64	NUM	Überprüfe eine eingegebene Zeile und forme einen hexadezimalen Wert von ASCII in binäre Darstellung. DE zeigt auf den Start der Zeile. Führende Leerzeichen werden ignoriert. Der Wert muß durch ein Leerzeichen oder eine Ø beendet sein. DE wird an das Programm übergeben und zeigt auf den nächsten Wert. Wenn der Wert ungültig ist (nicht 0-9,A-F oder größer als FFFFH), ist das Carry- Flag gesetzt, und DE zeigt auf das ungültige Zeichen. Der resultierende Binär- Wert ist in NUMV ($\varnothing C21 - \varnothing C22$) gespeichert. Die Anzahl der ASCII - Zeichen des Wertes ist in NUMN ($\varnothing C20$) gespeichert. Die Register HL und A werden verändert.
DF 66	TBCD3	Gib den Wert, der im HL - Register steht, im ASCII - Format aus, und addiere H und L in das C - Register. Das A - Register wird verändert.
DF 67	TBCD2	Gebe den Wert im A - Register in ASCII - Format aus. Addiere A in das C- Register. A wird verändert.
DF 68	B2HEX	Gebe den Wert des A - Registers im ASCII - Format aus. Das A - Register wird verändert.
DF 69	SPACE	Gib ein Leerzeichen aus. A enthält den Wert 20H.
DF 6A	CRLF	Gib CR/LF aus. A enthält CR.
DF 6B	ERRM	Gib den Text 'Error' aus, gefolgt von CR. A enthält CR.
DF 6C	TX1	Gib HL in ASCII aus, dann ein Leerzeichen, DE und noch ein Leerzeichen. H, L, D und E werden in das C - Register addiert. A wird verändert.
DF 6D	SOUT	Sende eine Zeichenkette direkt zur seriellen Schnittstelle. HL muß auf den Anfang der Zeichenkette gesetzt werden, in B muß die Länge der Zeichenkette stehen. Das Register C wird Ø und dann werden alle Zeichen in C addiert. Auch die Register HL, B und A werden verändert.

<u>Code</u>	<u>Name</u>	<u>Funktion</u>
DF 79	RLIN	Überprüfe eine Eingabezeile und forme bis zu zehn Hexadezimalzahlen von ASCII in hexadezimal um. DE muß auf den Anfang der Zeile zeigen. Die Zeile muß mit einer Ø beendet werden. RLIN benutzt die NUM - Routine für die Umformung der Werte. Wenn ein ungültiges Zeichen gefunden wird oder die Zeile mehr als zehn Werte enthält, wird das Carry- Flag gesetzt. ARGN ($\emptyset C\emptyset B$) enthält die Anzahl der gefundenen Werte. ARG1 ($\emptyset C\emptyset C - \emptyset C\emptyset D$) bis ARG1Ø ($\emptyset C1E - \emptyset C1F$) enthalten die binäre Form der gefundenen Werte. In NAS - SYS wird dieses Unterprogramm benutzt, um die Variablen nach der Eingabe eines Betriebssystem -Befehls zu erhalten. HL, DE, BC und A werden verändert.
DF 7A	B1HEX	Gib die untere (rechte) Hälfte von A in ASCII aus. Das A - Register wird verändert.
DF 7B	BLINK	Übergib ein eingegebenes Zeichen im A - Register. Während des Wartens auf die Eingabe blinkt der Cursor auf dem Bildschirm. Der Bildschirmspeicher wird nicht verändert. Die Register HL und DE werden verändert.
DF 7C	CPOS	Das HL - Register muß auf den Wert einer Adresse im Bildschirmspeicher gesetzt werden. Wenn CPOS dann aufgerufen wird, wird HL auf das erste Zeichen in eben dieser Zeile gesetzt. Register E und A werden verändert.

Ein/Ausgabe

Dieser Abschnitt beschreibt die sehr wirkungsvollen Methoden, mit denen NAS - SYS die Ein/Ausgabe kontrolliert, sodaß Sie davon einen möglichst sinnvollen Gebrauch machen können. Es ist jedoch nicht unbedingt erforderlich, daß Sie diesen Abschnitt verstehen, wenn Sie die Ein/Ausgabemöglichkeiten bietet, nutzen wollen.

Die Beschreibung der System-Befehle 'N', 'U', und 'X' zeigt Ihnen, wie Sie mit NAS - SYS ein Terminal oder einen Drucker anschließen können. Die NAS - SYS - Restart - Befehle RIN, ROUT und PRS ermöglichen Ihnen die Eingabe von Zeichen und die Ausgabe einzelner Zeichen oder Zeilen. Die NAS - SYS Routinen IN, BLINK und INLIN ermöglichen weitere Eingabe-Möglichkeiten, und TBCD3, TBCD2, B2HEX, B1HEX, SPACE, CRLF,ERRM, und TX1 ermöglichen die einfache Ausgabe von Daten. NUM und RLIN gehen noch einen Schritt weiter und helfen Ihnen, Daten aus einer eingegebenen Zeile zu gewinnen.

Sie sollten jetzt also wissen, wie Sie Ein/Ausgabe - Operationen in Ihr Programm einfügen können, und wie Sie kontrollieren, wohin diese Daten gesendet werden, wenn Sie die N, - U - und X - Befehle verwenden.

Sie können jedoch die Ein/Ausgabe noch wesentlich flexibler gestalten, wenn Sie die folgenden Erläuterungen von NAS - SYS verstehen.

Sowohl die Eingabe wie auch die Ausgabe arbeiten auf diese Weise:

Immer wenn eine Eingabe oder Ausgabe erforderlich ist, ruft ein besonderes Unterprogramm in NAS - SYS namens ATE eine Anzahl von Routinen zur Ein/Ausgabe - Behandlung auf. Wie alle anderen NAS - SYS - Unterprogramme haben auch diese Nummern, mit denen sie aufgerufen werden können. Es gibt eine Tabelle für die Eingabe- Programme und eine für die Ausgabe- Programme. Jede dieser Tabellen wird mit einer Ø abgeschlossen. Die Adresse der Eingabe- Tabelle ist in \$IN (ØC75 - ØC76) gespeichert, die der Ausgabe - Tabelle in \$OUT (ØC73 - ØC74).

Wie in NAS - SY üblich werden die Unterprogramm- Nummern in aktuelle Adressen umgerechnet, indem eine entsprechende Tabelle abgesucht wird. Die Startadresse dieser Tabelle ist in \$STAB (ØC71 - ØC72) gespeichert.

Die ATE - Routine, die jede dieser Ein/Ausgabe- Unterprogramme nacheinander aufruft, rettet automatisch die Register HL,DE und BC. Die Ausgabe- Programme müssen das AF - Register retten. Eingabe- Programme müssen mit dem gesetzten Carry- Flag und dem eingegebenen Zeichen in A zurückkehren, wenn ein Zeichen eingegeben wurde, andernfalls muß das Carry - Flag rückgesetzt sein.

Die NAS - SYS - Befehle N, U und X arbeiten so, daß sie die Adressen der Ein/Ausgabe - Tabellen verändern, also \$IN und \$OUT. NAS - SYS enthält unterschiedliche Ein/Ausgabe- Tabellen für diesen Zweck.

Nun da Sie wissen, wie die Ein/Ausgabe funktioniert, werden Sie sehen, daß es möglich ist, eigene Tabellen mit Adressen von Unterprogrammen zu erstellen und IN und OUT so zu verändern daß sie auf Ihre Tabelle zeigen. Vielleicht wollen Sie auch Ihre eigenen Ein/Ausgabe- Programme verwenden, und das ist sehr einfach mit dem U - Befehl zu machen. NAS - SYS enthält einige Unterprogramme, die es Ihnen erleichtern, die Adressen der Tabellen zu verändern.

<u>Code</u>	<u>Name</u>	<u>Funktion</u>
DF 71	NOM	Legen Sie in HL die Adresse der neuen Ausgabe-Tabelle ab und rufen Sie diese Routine auf. Die Adresse wird für Sie verändert, das Programm kehrt mit der alten Adresse in HL zurück.
DF 72	NIM	Legen Sie die Adresse der neuen Eingabe- Tabelle in HL ab. Dann rufen Sie diese Routine auf. Die Adresse wird für Sie verändert, das Programm kehrt mit der alten Adresse in HL zurück.
DF 77	NNOM	Setze den normalen Wert für die Ausgabe- Tabelle ein. Das Programm kehrt mit der alten Adresse in HL zurück.
DF 78	NNIM	Setze den normalen Wert für die Eingabe- Tabelle ein. Das Programm kehrt mit der alten Adresse in HL zurück.

Jetzt müssen Sie die Nummern der Unterprogramme kennen, die Sie in Ihre Tabellen einsetzen müssen.

Eingabe - Routinen

<u>Code</u>	<u>Name</u>	<u>Funktion</u>
61	KBD	Überprüfe die NASCOM - Tastatur
70	SRLIN	Überprüfe den seriellen Eingabekanal
74	XKBD	Überprüfe eine externe ASCII - Tastatur (s.a.X-Befehl)
76	UIN	Vom Benutzer angegebene Eingabe- Routine

Ausgabe - Routinen

<u>Code</u>	<u>Name</u>	<u>Funktion</u>
65	CRT	Anzeige auf dem NASCOM - Bildschirm (s.a. 'Editiermöglichkeiten')
6F	SRLX	Ausgabe zum seriellen Ausgabekanal
6E	XOUT	Ausgabe zu einem externen ASCII - Gerät (s.a.X-Befehl)
75	UOUT	Ausgabe über eine vom Benutzer gelieferte Ausgabe-Routine (s.a.U- Befehl)

NAS - SYS - Arbeitsspeicher

NAS - SYS benötigt einen bestimmten Speicherbereich als Arbeitsspeicher. Dies sind die Speicherstellen ØCØØ bis ØC7F, die erste freie Speicherstelle für Benutzer- Programme ist ØC8Ø. Der erste Teil des Arbeitsspeichers von ØCØØ bis ØC6A wird bei Reset auf Ø gesetzt. Der zweite Teil, ØC6B bis ØC7D, wird mit einer Tabelle aus NAS - SYS geladen. Auch ØC7E bis ØC7F werden von NAS - SYS initialisiert. Mit Benutzer- Programmen kann falls erforderlich diesen Arbeitsspeicher für bestimmte Zwecke benutzen und einzelne Werte verändern. Deshalb ist die folgende Tabelle beigefügt, aus der die einzelnen Funktionen der Adressen im Arbeitsspeicher hervorgehen.

Adresse	Länge	Name	Funktion
ØCØØ	1	PORTØ	Kopie des Zustandes von Port Ø
ØCØ1	9	KMAP	Zustand der Tastatur
ØCØA	1	ARGC	Buchstabe des zuletzt ausgeführten Befehls oder Nummer der letzten Ein/Ausgabe- Routine
ØCØB	1	ARGN	Anzahl der Variablen in einer Eingabezeile
ØCØC	2	ARG1	Erster eingegebener Wert
ØCØE	2	ARG2	Zweiter eingegebener Wert
ØC1Ø	2	ARG3	Dritter eingegebener Wert
ØC12	ØC	ARG49	Vierter bis neunter eingegebener Wert
ØC1E	2	ARG1Ø	Zehnter eingegebener Wert
ØC2Ø	1	NUMN	Anzahl der Zeichen in einem Wert, der mit der NUM- Routine geprüft wurde
ØC21	2	NUMV	Wert als Ergebnis der NUM- Routine
ØC23	2	BRKADR	Adresse des Breakpoints
ØC25	1	BRKVAL	Gespeicherter Inhalt der Breakpoint - Adresse
ØC26	1	CONFLG	Normalerweise Ø, aber auf -1 gesetzt für E- Befehl
ØC27	1	\$KOPT	Tastatur - Option, s. K- Befehl
ØC28	1	\$XOPT	X - Option, s. X- Befehl
ØC29	2	CURSOR	Position des Cursors
ØC2B	1	ARGX	Letztes eingegebenes Befehls- Zeichen
ØC2C	35	MONSTK	NAS - SYS - Stack (Stapelspeicher)
ØC61	2	RBC	Bereich für) Reg. BC
ØC63	2	RDE	das) Reg. DE
ØC65	2	RHL	Retten) Reg. HL
ØC67	2	RAF	der) Reg. AF
ØC69	2	RPC	Register) Programm - Zähler
----Von hier ab von einer Tabelle in NAS - SYS geladen----			
ØC6B	2	RSP	Ber.f. Retten der Register, Stapel- Zeiger
ØC6D	2	\$KTABL	Länge der Tastatur - Tabelle
ØC6F	2	\$KTAB	Start der Tastatur- Tabelle
ØC71	2	\$STAB	Start der Tabelle der Routinen- Adressen, für Routine ØØ. Da die erste Routine tatsächlich 41H ist, liegt die Tabelle 82H über dieser Adresse
ØC73	2	\$OUT	Start der Tabelle für die Ausgabe- Routinen
ØC75	2	\$IN	Start der Tabelle für die Eingabe- Routinen
ØC77	3	\$UOUT	Sprung zu den Benutzer- Ausgabe- Programmen
ØC7A	3	\$UIN	Sprung zu den Benutzer- Eingabe- Programmen
ØC7D	3	\$NMI	Sprung zum NMI - Unterprogramm. NAS - SYS setzt dies für die Anzeige der Register

Adressierung des Bildschirm-Speichers

Der Bildschirm-Speicher (Video-RAM) wird wie in untenstehender Tabelle adressiert. Die Adresse der obersten Zeile folgt auf die der anderen 15. Sie wird nicht von NAS-SYS gescrollt.

Links von jeder Zeile ist ein 10 Byte breiter Rand, dann folgt die 48 Byte breite Zeile, dann noch einmal ein rechter Rand von 6 Byte. Wenn der Bildschirm von NAS-SYS gelöscht wird, werden alle sichtbaren Speicherzellen mit Leerzeichen gefüllt, in die Ränder werden Nullen geschrieben, mit Ausnahme der ersten 10 Byte links neben der ersten Zeile und den letzten 6 Byte rechts neben der letzten Zeile, also der obersten Zeile, die nicht gescrollt wird.

Zeile Nr.	Linker Rand	Linke Seite		Rechte Seite	Rechtes Zeilenende
16	OBCO	OBCA	OBCB ...	OBF9	OBFF
1	0800	080A		0839	083F
2	0840	084A		0879	087F
3	0880	088A		08B9	08BF
4	08C0	08CA		08F9	08FF
5	0900	090A		0939	093F
6	0940	094A		0979	097F
7	0980	098A		09B9	09BF
8	09C0	09CA		09F9	09FF
9	0A00	0AOA		0A39	0A3F
10	0A40	0A4A		0A79	0A7F
11	0A80	0A8A		0AB9	0ABF
12	0AC0	0ACA		0AF9	0AFF
13	0B00	0BOA		0B39	0B3F
14	0B40	0B4A		0B79	0B7F
15	0B80	0B8A	0B8B ...	0BB9	0BBF

Mit "Linke Seite" und "Rechte Seite" sind jeweils die äußersten noch sichtbaren Stellen bezeichnet.