

Anleitung

zum Synergy-Modulgenerator

(C) 1993 Solder of Synergy

Inhaltsverzeichnis

1. Programmbeschreibung.....	3
2. Anforderungen.....	3
3. Nun geht's los.....	4
4. Programmstart.....	6
5. Der Modulkopf.....	7
6. Einschaltmeldung.....	8
7. Bänke sperren.....	9
8. Auswahl der Programme.....	9
9. Weitere Angaben.....	10
10. Laden der Programme.....	11
11. Speichern der Brennfiles.....	11
12. Laden in gesperte Bank.....	12
13. Wie weiter mit den Brennfiles.	12
14. Begriffe.....	13
15. Copyrights.....	15
16. Service-Informationen.....	15

1. Beschreibung

Dieser Modulgenerator erstellt 32kB-, 64kB- und 96kB-Module für die interne EPROM-Erweiterung und externe Steckmodule sowie die sRAM-Karte. Dabei werden dem User sehr viele Dinge durch das Programm abgenommen, so daß die Bedienung recht einfach ist. Die folgende Anleitung wird die Bedienung anhand eines Beispiels verdeutlichen. An den entsprechenden Stellen gibt es Hinweise für weitere Möglichkeiten in kursiv.

2. Anforderungen

Die wichtigste Anforderung ist das Vorhandensein einer 256kB-Erweiterung in Plus/4 oder C16. Das muß die von Hannes (Hans Lux) ergänzte Erweiterung aus der RUN sein. (Entspricht den Erweiterungen die ich einbaue). Andere 256kB-Erweiterungen sind nicht verwendbar.

Das Programm ist für den Zwei-Floppy-Betrieb ausgelegt. Dabei sollten Sie, falls Sie eine 1541 und 1551 verwenden wollen, die 1541 als Gerät 9 einsetzen. Das Programm läuft jedoch auch mit nur einer Floppy. Im Beispiel wird dann auf die entsprechenden Unterschiede hingewiesen.

Ein wichtiger Punkt ist die Kenntnis der Verwaltung der ROM-Bänke. Je nach zu erstellendem Modul werden die Bänke 5, a und f verwendet. Die Bank 5 ist nur im Rechner vorhanden. Bei "normalem" Rechner die Bänke a und f nur am Expansionsport. Haben Sie einen Rechner mit EPROM-Erweiterung (System Hans Lux, 384k-System), so stehen die Bänke a und f auch intern zur Verfügung.

3. Nun geht's los.

Bevor Sie das Modul erstellen können, müssen Sie sich natürlich erst mal die Software aussuchen, die Sie dann brennen oder in die sRAM-Karte laden wollen. Dafür gelten folgende Richtlinien: Reine Basicprogramme und MC-Programme mit SYS-Aufruf in BASIC-Zeile (werden zukünftig nur noch als Basicprogramme bezeichnet), MC-Programme die an ihrer ersten Adresse im Speicher starten, Speicherbereiche ohne Start (z.B. F-Tasten-Belegung) und fertige Brennfiles von Originalmodulen. Alle Programme müssen einteilig, d.h. nicht nachladend, sein! Originalmodule sind solche wie Script/+ oder fertig erstellte Programmsammlungen anderer Anbieter. Diese Originalmodule müssen jedoch für die Verwendung erst aufbereitet werden, damit sie nicht automatisch starten. Meist genügt dazu die Zerstörung der Kennung "CBM". Ein kleines Maschinenprogramm (der sog. Starter) ruft diese Originalmodule dann aus dem Moduldirectory auf. Bei Script/+ wird statt eines solchen Starters der Support verwendet. Außerdem ist bei Script/+ die CBM-Kennung nicht zu löschen.

Weitere Hinweise hierzu sind in einer der ersten Ausgaben der Club-Info enthalten. Das Thema ist sehr komplex, so daß eine Erläuterung hier zu weit führt. Falls Sie ein spezielles Problem haben, stehe ich gern zur Verfügung.

Ich möchte hier auch noch auf einige Spiele eingehen. Diese werden zwar oft mit DLOAD geladen, lassen sich dann aber doch nur mit einem SYS starten. In diesem "Zustand" können sie in der Erweiterung nicht verwendet werden. Zwei Möglichkeiten gibt es in diesem Fall: Man fügt noch eine Basic-Zeile 0 mit dem SYS ein, was jedoch in machen Fällen zur Zerstörung von Daten führt. Oder man programmiert im Monitor vor das Programm

einen JMP\$xxxx, wobei für die xxxx dann die hexadezimale Startadresse zu setzen ist. Das Spiel speichert man dann vom Monitor ab. Im Modulgenerator wirkt es dann wie ein Maschinenprogramm mit Start an der ersten Adresse. In OS92 kann übrigens nach dem Laden leicht festgestellt werden, in welchen Bereich geladen wurde, wenn man die Speicherstellen >9b und folgende aufruft (low/high-Format).

Sie haben nun also die Programme ausgesucht, die in die Erweiterung sollen. Es dürfen aber nicht mehr als 22 Programme (pro Ebene) sein, da ein längeres Moduldirectory nicht auf den Bildschirm paßt. Originalmodule zählen dabei aber nicht mit, da sie keinen Eintrag erhalten. Jetzt benötigen Sie so viele Disketten, wie die Erweiterung Ebenen hat. Stellen Sie ein externes Modul zusammen oder hat die interne Erweiterung nur eine Ebene brauchen Sie nur eine Disk. Auf die Vorderseite kommen die Programme. Achten Sie darauf, daß es nicht zu viele sind. Pro Bank kann man mit 120BLK rechnen. (Bei der internen Erweiterung pro Ebene noch 294 Blocks free.) Sind Sie sich nicht sicher, so können Sie auch mehr nehmen, da später noch die Möglichkeit der Auswahl steht.

Als oben erwähntes Beispiel nehmen wir an, Sie wollen eine Ebene (96kB, Bank 5, a und f). der internen 384kB-Erweiterung erstellen. Für jede weitere der 4 Ebenen verfahren Sie so wie jetzt beschrieben. *Für externe Module gilt alles analog, nur daß Sie hier auf die Bank 5 verzichten müssen.* Als Brennfile eines Originalmoduls wollen Sie den Ungarn-Monitor verwenden. Sie kopieren also die Programme auf die Vorderseite der Disk. Die Rückseite bleibt frei. *Sollten Sie jedoch Brennfiles von Originalmodulen verwenden, die eine komplette Bank benötigen, so kopieren Sie diese gleich auf die Rückseite und benennen*

Sie sie so um: "bf0x; name". Für das x setzen Sie die Bank a oder f ein, je nach dem in welche Bank das Brennpfile kommt. Es versteht sich, daß ein Originalmodul nicht in die selbe Bank kann, wie der Modulkopf. Aus diesem Grund ist es auch nur möglich bei der internen Erweiterung maximal 2 Originalmodule pro Ebene unterzubringen (eins in a, eins in f). Extern ist dann bei einem 64kB-Modul nur ein Originalmodul möglich, beim 32kB-Modul gar keins. Beachten Sie schon bei der Zusammenstellung, daß manche Originalmodule eine ganz bestimmte Bank erfordern!

Haben sie nur eine Floppy, müssen Sie auch den entsprechenden Modulkopf (intern/extern) auf die Diskette kopieren!

Nachdem nun die Diskette (bzw. alle) erstellt wurde, können Sie den Modulgenerator laden und starten.

4. Programmstart

Laden Sie den Modulgenerator normal und starten ihn mit RUN. Arbeiten Sie mit zwei Floppys, können Sie den Modulgenerator in OS92 mit LOAD gut von der 9 laden. Vor dem Start legen Sie jedoch die Disk mit den Programmen in das Laufwerk 8. Die Disk mit dem Modulkopf kommt in die 9 und bleibt dann dort. Nach einigem Warten - in dieser Zeit wird der RAM gelöscht - erscheint die Mitteilung, daß das Directory gelesen wird. Das Programm ermittelt die Programme, die Sie vorher auf die Disk kopiert haben. Nun versucht es die Filedaten zu lesen, in der vielleicht schon die Daten der Programme (Start, Ende, Länge) abgelegt sind. Da es diese Filedaten (noch) nicht gibt, werden nun von allen Programmen diese Daten ermittelt. *Hatten Sie die Filedaten aber schon, werden erst alle Programme aus dieser Liste gelöscht, die nicht mehr im Directory stehen.*

Dann werden die Daten von den Programmen ermittelt, die neu auf die Disk kopiert wurden. Dieses Verfahren bietet den Vorteil, daß nicht immer alle Daten neu ermittelt werden, wodurch Zeit gespart wird. Allerdings kann es passieren, daß die Daten von einer neuen Programmversion nicht ermittelt werden, da diese den selben Namen wie vorher hat. Deshalb sollte man auf eindeutige Unterscheidung durch Programmnamen achten. Oder Sie löschen vorher die Filedaten.

Beobachten Sie unbedingt die Ermittlung der Filedaten! Wurde für alle Basic-Programme der Start von \$1001 festgestellt? Wenn nicht (z.B. \$4001), so wurde das BASIC-Programm bei eingeschalteter Grafik abgespeichert. Sie haben dann später noch die Möglichkeit den Start auf \$1001 umändern zu lassen. Achten Sie auch darauf, daß für die Brennpfiles der Originalmodule als Startadresse \$4000 ermittelt wird! Zum Schluß werden die neuen Filedaten auf Disk abgelegt.

5. Der Modulkopf

Nun werden Sie gefragt in welche Bank der Modulkopf soll. Erstellen Sie eine interne Erweiterung, so ist diese Frage immer mit 5 zu beantworten. *Bei externen Erweiterungen gibt es die Möglichkeiten a und f.* Bei der internen Erweiterung wird jetzt sofort der Modulkopf "mk intern" von Floppy 9 (bzw. 8) geladen. *Möchten Sie einen anderen Modulkopf für die interne Erweiterung verwenden, so wählen Sie erst eine beliebige andere Bank und dann bei der Anzeige der Modulköpfe den internen aus.* Alle internen Modulköpfe müssen in den Zeichen "mk i" übereinstimmen. *Dadurch wird dann bei Anwahl dieser Köpfe automatisch auf Bank 5 geschaltet. Haben Sie also Bank a oder f gewählt, wird das Directory mit allen Modulköpfen angezeigt. Gehen Sie einfach mit*

dem Cursor auf den gewünschten Modulkopf und drücken Sie Return. Bei externen Modulköpfen müssen folgende Zeichen immer gleich sein: "mk e"

Für das Beispiel wählen Sie natürlich Bank 5.

6. Einschaltmeldung

Geben Sie nun die Einschaltmeldung ein, die erscheinen soll, wenn der Rechner eingeschaltet und das Modul gefunden wurde. Eine Vorgabe ist bereits gemacht. Sie können Sie beliebig verändern. Für das interne Modul ist die Aufruftaste die F1 (immer). Bei externen Modulen sollten Sie die F4 nehmen. Jede andere Taste ist aber auch möglich. Beachten Sie das aber beim Eintrag in die Zeile. Sie können nach Return nicht zurück. Es hilft dann nur der Programmabbruch (gut daß die Filedaten gespeichert sind).

Als nächstes kommt die Überschrift für das Moduldirectory.

Sie können übrigens die Eingabemasken (normale Input-Masken) nicht verlassen. Drücken Sie aber nicht zweimal HOME hintereinander.

Erstellen Sie ein externes Modul, werden Sie nun gefragt, mit welchem SYS das Modul aufgerufen wird. Günstig ist dort die Adresse 3054. Den Text "SYS" nicht verändern! Der Modulkopf wird nun entsprechend geändert, so daß der Starter dann tatsächlich an dieser Adresse liegt.

Nun wählen Sie noch die F-Taste mit der das Modul aufgerufen wird. Die Vorgabe F4 sollten Sie akzeptieren.

Zum Schluß erscheint noch ein Hinweis auf den Beginn der Modultabelle, die alle Daten über die im Modul abgelegten Programme speichert.

7. Bänke sperren

Nun werden Sie noch gefragt, ob Sie Bänke sperren wollen. Antworten Sie in unserem Fall mit n. Das Sperren von Bänken ist nötig, wenn Sie ein externes 32kB-Modul erstellen wollen. Sie sperren die Bank, die Sie nicht brauchen. Der Modulkopf muß natürlich in der anderen Bank liegen. Bei der internen Erweiterung sperren Sie eine Bank oder mehrere wenn Sie ein Originalmodul verwenden wollen, das eine komplette Bank belegt. Dann wird diese vom Modulgenerator nicht berücksichtigt. Das ist z.B. bei Script/+ der Fall. Diese Originalmodule haben Sie dann auch vorher wie oben beschrieben auf die Rückseite der Disk kopiert

8. Auswahl der Programme

Nun gelangen Sie in ein Auswahlmenü, in dem Sie die Programme auswählen können, die tatsächlich in das Modul kommen sollen. Mit Return bestätigen Sie das oben angezeigte Programm. Es wird dann gleich auf das darunter liegende Programm gescrollt. Mit den Cursor-Tasten können Sie die Ursprungsliste hoch- und runterscrollen. Bereits ausgewählte Programme haben dabei ein * vor dem Namen. Diese können Sie dann mit Delete auch wieder aus der Liste entfernen. Rechts wird der noch freie Platz angezeigt, der auf den nicht gesperrten Bänken noch vorhanden ist. Haben Sie alle Programme ausgewählt, beenden Sie die Wahl mit ESC. Arbeiten Sie nur mit einer Floppy, wird hier auch der Modulkopf in der Liste angezeigt. Diesen wählen Sie natürlich nicht aus.

9. Weitere Angaben

In diesem Menü bestimmen Sie nun die Art des Programmes, die aufrufende Taste und den Titel im Moduldirectory. Zuerst wird der Programmtyp (Basicprogramm, MC-Programm, Programm ohne Start, Modul an \$4000) festgelegt. Meist stimmen die Vorgaben dazu schon. Ein Programm wird als Basicprogramm angegeben, wenn die Ladeadresse \$1001 oder \$4001 ist. Ein solches Programm wird dann mit RUN gestartet. Sollte es sich doch nicht um ein BASIC-Programm handeln, können Sie die Art ändern, indem Sie mit den Cursortasten hoch und runter in der Typenliste scrollen und mit RETURN bestätigen. Für BASIC-Programme wird dann der Start grundsätzlich auf \$1001 gesetzt! Das bedeutet auch, daß Sie so ein Programm aus dem Modul nicht bei aktiviertem Grafikbereich aufrufen können. Bisher hat es damit aber nie Probleme gegeben. Ein Modul an \$4000 sind die Brennpfiles die in der Regel immer auf dieser Adresse liegen. Lediglich bei MC-Programmen (Start mit erster Adresse) und Programmen ohne Start müssen Sie selbst die Art bestimmen. Vorgabe ist immer MC-Programm. Bei Originalmodulen wie dem Ungarn-Monitor gelangt man nun gleich nach der Typ-Bestätigung zum nächsten Programm, denn da es keine aufrufende Taste hat und keinen Eintrag im Directory bekommt, brauchen wir entsprechende Eingaben auch nicht tätigen. Dafür benennen Sie dann das Startprogramm für das Originalmodul so wie es heißt, also "Ungarn-Monitor" oder so ähnlich bei diesem Beispiel. Die Aufruf-Tasten können beliebig gewählt werden. Shift+x und Space sind jedoch reserviert für andere Zwecke. Der einzugebende Name kann auch länger als 16

Zeichen sein. Vergewissern Sie sich, daß alles richtig ist, bevor die RETURN-Taste gedrückt wird. Es gibt kein Zurück. Bei Fehlern hilft nur der Abbruch des Programms und Neustart.

10. Laden der Programme

Nachdem alle Programme bearbeitet wurden, werden Sie nun noch gefragt, in welche Bänke die Originalmodule kommen (falls Sie solche hatten). Geben Sie hier ein a an, da der Ungarn-Monitor nur in Bank a läuft. Nun beginnt das Laden der Programme, Brennpfiles der Originalmodule zuerst. Das Programm weist automatisch die Bänke den Programmen zu, so daß diese immer maximal belegt sind. Gesperrte Bänke werden nicht belegt. Sie können den Fortschritt des Ladens mit Angabe der neuen Namen verfolgen. Der Start der Ablage der Programme im Speicher wird in RAM-Bank und RAM-Adresse angezeigt.

11. Speichern des fertigen Brennpfiles

Nun können Sie entscheiden, ob Sie das Brennpfile speichern wollen. Auch wenn das immer recht lange dauert, sollten Sie das unbedingt tun. Drehen Sie die Disk um. Geben Sie einen Namen für die Brennpfiles ein. Sie bekommen dann den Namen: "bf0x; name". Das x steht wieder für die entsprechende Bank. Nun erscheint das Directory der Disk (-Rückseite). Stehen darauf noch Daten die Sie brauchen, so sollten Sie bei löschen mit Format ein "n" eingeben. Auch bei leerer Disk ist das möglich. In diesem Fall werden alle Files mit "bf0x*" gelöscht, wobei aber Files in gesperrter Bank x nicht gelöscht werden! *Das ist wichtig, wenn auf der Rückseite der Disk schon ein Originalmodul (wie Script/+)*

steht, das eine komplette Bank benutzt. Steht anderer Müll (oder die Brennfiles vom letzten Mal) auf der Disk, können Sie auch formatieren (j). Die Disk wird dann weich formatiert. Das geht natürlich schneller als das Löschen der einzelnen Files mit Scratch. Das Speichern dauert eine Weile. Sie können auf dem Bildschirm verfolgen, wie weit der Rechner bereits ist. Gesperrte Bänke werden natürlich nicht mit gespeichert.

12. Laden in gesperrte Bank

Hatten Sie eine Bank gesperrt, weil Sie den Platz für ein Originalmodul reservieren wollten, das eine komplette Bank belegt, so können Sie dieses Modul jetzt noch automatisch nachladen lassen, und Sie sparen einige Arbeit.

13. Wie weiter mit den Brennfiles?

Die Brennfiles stehen nun noch im Speicher und zwar für:

RAM	ROM
Bank1	Bank5
Bank2	Banka
Bank3	Bankf

Nach Ende des Modulgenerators können Sie gleich den EPROMer laden und die EPROMs brennen. Für den Braunrothbrenner verwenden Sie das 256kB-Brennprogramm. Beim SYNERGY-EPROMMER können Sie die RAM-Bänke bei Angabe des zu brennenden Bereiches wählen. Für die interne Erweiterung wird je ein Brennfile (eine Bank) in ein EPROM gebrannt. Für die weiteren Ebenen kommen dann jeweils die selben Bänke in das selbe EPROM, wobei im Brennprogramm immer ein 32kB-Bereich des

EPROMs weiterschaltet wird.

Was aber, wenn man das Brennfile in die sRAM-Karte laden möchte? Man kann im Monitor mit dem t-Befehl die Daten in den auf RAM-geschalteten RAM transferieren. Das ist aber nicht besonders komfortabel. Deshalb habe ich ein kleines Programm beigelegt, das dies etwas einfacher löst. Geben Sie einfach mit j oder n an, ob der Computer-RAM in den sRAM übertragen werden soll. Beachten Sie die Hinweise auf dem Bildschirm.

14. Begriffe

Modul: Ein Programm, das in einem oder mehreren EPROM(s) ab Adresse \$8000 abläuft.

Steckmodul: Die Hardware (Modul) mit EPROMs, die in den Expansionsport gesteckt wird.

sRAM-Karte: Emulator eines Steckmoduls. Das Brennen der EPROMs entfällt, da sRAMs verwendet wurden.

Bank: RAM-Bank oder ROM-Bank. RAM-Bänke gibt es im Plus/4 mit 256kB 4 Stück. Gemeint sind in der Anleitung aber die ROM-Bänke, die mit dem Adreßbereich ab \$fdd0 selektiert werden. In Bank 0 steht das Basic und Kernal. Die Bank 5 ist normalerweise beim Plus/4 mit der integrierten Soft belegt. Wir verwenden diese Bank zusammen mit den Bänken a und f für die interne Erweiterung. Die Bänke a und f sind auch am Expansionsport vorhanden. Bei erweitertem Plus/4 müssen sie dort aber erst aktiviert werden. Jede Bank ist knapp 32kB groß. Die Teilung in Low- und High-Bereich ist bei uns nicht mehr von Bedeutung und wird auch vom Modulgenerator nicht unterstützt. Folgende Bereiche stehen zur Verfügung: \$8000-\$fbff, \$ff40-\$ffff. Nur der erste Bereich wird vom Modulgenerator benutzt. Im

letzten Bereich werden nur RESET- und IRQ-Vektor initialisiert (ab \$fff6)

Ebene: Ebenen gibt es erst seit der internen EPROM-Erweiterung. Dort werden immer die 3 Bänke 5, a, f zu einer Ebene von 96kB zusammengefaßt. Diese Größe wird dann auch vom Moduldirectory verwaltet. In andere Ebenen gelangt man durch die Cursortasten, was dem Wechseln einer Disk entspricht. Prinzipiell kann man auch von Ebenen sprechen, wenn ein Modul am Expansionsport per Hand oder Soft umgeschaltet wird (eine Bank mehrmals vorhanden).

Modulgenerator: Programm zum Erstellen von Modulen bzw. fertigen Brennfiles.

Modulkopf: Das Programm an Adresse \$8000 im Modul. Es sagt dem Betriebssystem, daß ein Modul vorhanden ist und kommuniziert mit ihm. Es verwaltet das Moduldirectory, kopiert die Programme in den RAM und startet sie bei Bedarf. Bei der internen Erweiterung werden auch externe Module damit aktiviert.

Modultabelle: Enthält alle Daten über die im Modul abgelegten Programme, die der Modulkopf benötigt: Adresse im Modul, Endadresse im Modul, Adresse im RAM, Programmtyp, Starttaste u.m.

Brennfile: Das Modul ist die ablauffertige Software ab \$8000 und kann mehrere Brennfiles umfassen. Ein Brennfile ist maximal 32kB groß und liegt an \$4000, wie es der SYNERGY-ERPOMMER und der Braunroth-Brenner benötigen. Das Modul wäre an dieser Stelle nicht lauffähig.

15. Copyrights

Auf das Programm "Modulgenerator" habe ich das alleinige Copyright. Niemand ist berechtigt das Programm zu vervielfältigen und die Kopie an Dritte weiterzugeben. Das Copyright besteht nicht auf die Modulköpfe. Wer Interesse hat, kann den Quelltext (Hypra-Ass) von mir gegen Disk und Rückporto bekommen. Der Modulgenerator ist sehr universell, so daß beliebige Modulköpfe verwendet werden können. Diese müssen nur bestimmte Eigenschaften haben, können sonst aber ganz variabel sein.

Ich möchte alle User bitten, das Copyright nicht zu verletzen. 10DM für ein solches Programm (auch wenn es mehrheitlich in Basic programmiert wurde) und die Anleitung sind wirklich nicht zu viel. Sie besitzen nur die Originaldisk von mir, wenn darauf mein Adreßaufkleber ist.

16. Service

Bei Fragen, Anregungen, Kritik sollte man sich an mich wenden: Solder of Synergy, c/o Christian Schöffner, Bauernreihe 50, D-38822 Emersleben. Telefon am Wochenende: 039424/5302.

Emersleben, 5/1993