

Sprachsynthese:

Stimmbänder für Heimcomputer

Wenngleich nicht neu, so doch verblüffend einfach ist die Sprachsynthese mittels digitalisierter NF-Signale. Fast jeder Computer mit Kassetten-Interface ist dazu geeignet. Hier aber macht der ZX Spectrum die Probe aufs Exempel.

Über die verschiedenen Methoden der Sprachsynthese hat die FUNKSCHAU zuletzt in Heft 2/1984 berichtet. Und in Heft 7/1984 erschien eine Bauanleitung für die Sprachsynthese mit einem Phonet-Prozessor. Der Vorteil dieser Methode: der Speicherplatzbedarf ist äußerst gering.

Im Vergleich dazu geht das Verfahren, das wir jetzt aufgreifen, geradezu verschwenderisch mit Speicherplätzen um. Aber dafür kommt es – kaum zu glauben – prinzipiell ohne jegliche Zusatz-Hardware aus! Der Computer und ein Kassettenrecorder genügen.

der ein Kassetten-Interface hat. Wird hier das Sprachsignal eingespeist, geschieht folgendes: Überschreitet der NF-Pegel eine am Komparator eingestellte Schaltschwelle, dann „kippt“ das Signal am Komparator-Ausgang von L- auf H-Pegel und kippt erst dann wieder auf L-Pegel zurück, wenn der NF-Pegel die Schaltschwelle unterschreitet. Das Analogsignal wird also in ein Digitalsignal umgeformt.

Von der filigranen Amplitudenstruktur des NF-Signals bleibt freilich nicht mehr viel übrig, nur noch die Information: Amplitude ist größer oder kleiner als die Schaltschwelle. Doch diese 1-Bit-A/D-Wandlung (da nur eine Schaltschwelle) genügt.

Über einen Kamm geschoren

Die Sprachsynthese mit digitalisierten NF-Signalen baut darauf auf, daß Sprachsignale stark verfremdet werden dürfen, ohne daß die Verständlichkeit gleich verlorengeht. So genügt z. B. beim Telefonieren ein Übertragungsbereich von 300 Hz bis 3,4 kHz völlig, um eine ausreichende Verständigung sicherzustellen. Für die Verständlichkeit einer Stimme sind alle übrigen Frequenzanteile ohne große Bedeutung. Weniger bekannt ist, daß auch die feinen Amplitudenschwankungen innerhalb eines Sprachsignals fehlen dürfen, ohne größere Einbußen an Verständlichkeit hinnehmen zu müssen.

Eine Schaltung (Komparator), die diese feinen Amplitudenschwankungen tatsächlich standhaft ignoriert, steckt schon in beinahe jedem Heimcomputer,

Statt Programme Sprache laden

Die weitere Signalverarbeitung hängt in Einzelheiten stark vom jeweiligen Computer-Modell ab. Schließlich muß das digitalisierte NF-Signal, das jetzt genauso wie die Bits eines Programms als serieller Strom von Nullen (L-Pegel) und Einsen (H-Pegel) vorliegt, irgendwie im Speicher des Computers abgelegt werden.

Dazu muß der Datenstrom bei jedem Computer eine vom Betriebssystem festgelegte Speicherzelle „durchfließen“. Wird ein Programm eingelesen, dann fragt das Betriebssystem den augenblicklichen Inhalt dieser Speicherzelle in genau definierten Zeitabständen ab (Baudrate) und kann so aus dem seriellen Datenstrom Bytes bilden, die der Reihe nach im Programmspeicher abgelegt werden.

Kennt man die Adresse der Speicherzelle, dann kann man die Abfrage des Inhalts auch mit einem selbst geschriebenen Programm vornehmen. Im Falle der Sprachsignal-Einspeisung zappelt in dieser Speicherzelle ein Bit mit der Frequenz des Sprachsignals zwischen L- und H-Pegel hin und her. Um möglichst jeden Pegel-Wechsel mitzubekommen, muß die Speicherzelle sehr oft (z. B. mit einer Abtastfrequenz von 10 kHz) abgefragt werden.

Basic ist dafür zu langsam. Das nötige Tempo bei der Abfrage wird am besten durch ein Maschinenprogramm sichergestellt. Dann läßt sich sogar noch eine Warteschleife einbauen, um die Abtastfrequenz experimentell zu verändern. Immerhin ist das Ergebnis jeder Abfrage ein Bit (0 oder 1), das im Speicher abgelegt werden muß; bei unnötig hoher Abtastfrequenz wäre der Speicher sehr bald voll.

Verschwenderisch wäre es auch, wenn man jedes abgefragte Bit in einer 8-Bit-Speicherzelle unterbringen würde, ohne die übrigen sieben Bits zu nutzen. Besser, da programmiertechnisch einfach und kaum zeitraubende Schiebeoperationen benötigt werden, ist folgendes Verfahren: Im gesamten freien Speicherbereich wird zuerst das erste Bit einer jeden Adresse belegt, dann mit dem zweiten Bit wieder von vorne begonnen usw. bis schließlich auch das achte Bit jeder Adresse mit einer „Abtastprobe“

```

1000 REM **SPRACHSYNTHESE**
1020 RESTORE 2000: CLEAR 40000:
LET S=0
1030 PRINT "Bitte warten"
1040 FOR I=40000 TO 40145
1050 READ X: LET S=S+X: POKE I,X
1060 NEXT I: IF S=18404 THEN PR
INT "Pruefsumme stimmt": STOP
1070 PRINT "Pruefsumme stimmt ni
cht": STOP
2000 DATA 80,195,95,234,8,10,1,2
54,0,237,120,230,84,95,253,42,69
,155,253,85,237,120,230,84,157,4
0,249,95,21,32,245,221,42,66
2010 DATA 156,43,64,156,84,0,35,
125,221,189,32,248,124,221,188,3
2,243,30,1,42,64,156,237,120,230
,64,40,1,123,182,119,205,200,156
2020 DATA 35,125,221,189,32,238,
124,221,188,32,233,203,3,48,226,
201,1,254,0,58,22,203,203,203,
53,203,63,230,7,87,30,1,221,42,1
2030 DATA 66,156,42,64,156,125,1
53,40,3,62,24,175,237,32,121,15,20
57,158,35,125,221,189,32,203,124,
221,188,32,233,3,48,226,201,
2040 DATA 253,42,68,156,253,125,
51,32,253,201
    
```

① **Synthese-Programm:** Der Maschinencode findet zwischen Adresse 40000 und 40145 seinen Platz im Speicher

versorgt wurde. Damit ist die digitalisierte Sprachinformation gespeichert, und die größte Hürde gemeistert.

Filter bessern die Sprachwiedergabe

Zur Sprachsynthese läuft der gesamte Vorgang umgekehrt ab. Man muß dazu die gespeicherten Bits in der richtigen Reihenfolge in eine ebenfalls vom Betriebssystem des Computers festgelegte Speicherzelle laden (Kassetten-Interface-Ausgang), so daß wieder der ursprüngliche Datenfluß entsteht. Diesen Datenfluß kann man verstärkt einem Lautsprecher zuführen, aus dem dann die gespeicherte Stimme ertönt.

Wer mit der Verständlichkeit der synthetischen Stimme nicht zufrieden ist, kann – wenn er zum Lötcolben greift – eine Verbesserung herbeiführen. Einmal mit einem Bandpaß (300 Hz...3,4 kHz), der das in den Computer eingespeiste NF-Signal filtert und weiterführend mit einem Tiefpaß ($f_0 = 3,4$ kHz), der den ausgegebenen Datenstrom von Oberwellen befreit.

Kritisch ist der richtige NF-Pegel am Eingang des Kassetten-Interfaces. Er hängt von der Schaltschwelle des Komparators ab und wird am besten experimentell ermittelt. Dazu spricht man einige Testworte auf Band, spielt es ab und speist das NF-Signal am Eingang des Kassetten-Interfaces ein. Klingt die synthetische Stimme dann sehr stark verzerrt, ist der Vorgang mit anderen Lautstärkepegeln zu wiederholen.

Der ZX Spectrum bekommt Stimmbänder

Für erste Experimente mit dem ZX Spectrum ist außer dem Computer nur noch ein Kassettenrecorder mit Mikrofon nötig. Die Adressen für das Kassetten-Interface haben beim ZX Spectrum ein und denselben Wert bei der Eingabe und Ausgabe von Daten, nämlich 254. Nur wird bei der Eingabe der Datenstrom durch Bit 6 dieser „Portadresse“ geschleust und bei der Ausgabe durch Bit 3.

Da das Experimentierprogramm (Bild 1) die digitalisierte Sprache nicht nur am MIC-Ausgang bereitstellt, son-

dern über Bit 4 der Portadresse auch den eingebauten Piezo-Lautsprecher ansteuert, tönt die Sprache in bescheidener Qualität auch aus dem Computer.

Das Experimentierprogramm, das im Speicher zwischen Adresse 40000 und 40145 abgelegt ist, benötigt einige Variablen. So bestimmt die unter Adresse 40004 abgelegte Variable die Abtastfrequenz. Je kleiner der Wert ist, desto besser ist die Wiedergabequalität, aber um so mehr Speicherplatz wird benötigt. Ein Kompromißwert wäre 20; damit werden bei 1 s Sprachaufzeichnung 1000 Byte Speicherplatz belegt.

Damit die Vergabe von Speicherzellen erst dann beginnt, wenn ein Sprachsignal am EAR-Eingang eingespeist wird, hat das Programm am Anfang eine Warteschleife. Sie wird nur dann verlassen, wenn am EAR-Eingang eine definierte Zahl von Pegeländerungen stattgefunden hat. Definiert ist diese Zahl als Variable unter der Adresse 40005, wobei der Wert 8 zweckmäßig ist.

Zwei weitere Variablen (Adresse 40000 und 40002, je 2 Byte) legen den Speicherbereich fest, in dem die digitalisierte Sprache abgelegt wird. Weist man diesen Variablen die Adressen 40146 und 65293 zu (48-KByte-Spectrum), reicht der Speicherplatz, um 25 s Sprache in durchschnittlich guter Qualität aufzuzeichnen. Alle vier Variablen müssen vor dem Programmstart mit Werten versorgt werden. Danach darf der Einlesevorgang mit RANDOMIZE USR 40006

gestartet werden. Im gleichen Moment ist an der EAR-Buchse ein NF-Signal einzuspeisen, am besten das von zuvor auf Band gesprochenen Testworten. Ist dabei der Pegel zu hoch, artet später die Wiedergabe in ein Prasseln aus; ist der Pegel zu niedrig, hört man nur Bruchstücke der Testworte. Eingeleitet wird die Wiedergabe durch den Befehl

RANDOMIZE USR 40084

War der Pegel beim Einlesen richtig, müssen jetzt die Testworte vom Computer „gesprochen“ werden. Deutlicher und lauter ist die Wiedergabe jedoch, wenn ein NF-Verstärker an der MIC-Buchse des Computers angeschlossen wird. Mit

SAVE „Name“ CODE a, b

läßt sich die künstliche Stimme auch auf Band speichern, wobei a die Anfangsadresse des Sprachspeicherbereichs ist und b die Zahl der dafür reservierten Bytes.

Ein Hilfsprogramm als Experimentierhilfe

Ein Hilfsprogramm, das die Eingabe der Übergabe-Variablen für das Maschinenprogramm erleichtert, zeigt Bild 2. Dieses Programm übernimmt im Dialogbetrieb auch den Aufruf der Eingabe- und Ausgaberroutinen für die Sprache.

Zuerst ist jedoch das Eingabeprogramm für den Maschinencode (Bild 1) einzutippen und zu starten. Danach hat der Maschinencode seinen Platz im Adreßbereich 40000 bis 40145. Anschließend ist das Eingabeprogramm zu löschen und das Hilfsprogramm einzutippen. Startet man es dann mit RUN 9999, so wird das Hilfsprogramm und daran anschließend der Maschinencode auf Band gespeichert. Beim Wiedereinlesen bewirkt Zeile 5 das automatische Laden des Maschinencodes.

Das Maschinenprogramm nutzt Befehle, die in der Z-80-Befehlstabelle nicht aufgeführt sind. Diese Befehle beziehen sich auf die Registerhälften von ix und iy. Erst durch sie ist es gelungen, das komplette Programm in 150 Byte zu packen. Ein knapp kommentiertes Assemblerlisting des Maschinenprogramms ist gegen Einsenden von 2,40 DM in Briefmarken beim Franzis-Software-Service erhältlich (Stichwort: FUNKSCHAU 21/1984, Sprachsynthese). Jörg Roth/-ll

```

5 CLS : PRINT "Bitte warten":
LOAD ""CODE": CLS
10 INPUT "Ab welcher Speichera
dresse soll die Sprache gespeich
ert werden? (z.B. 40146) von: I
NPUT "Bis zu welcher Adresse? ",
(z.B. 50000) bis
20 INPUT "Sprachqualitaet? (z.
B. 20) qual
30 POKE 40002, bis-256*INT (bis
/256): POKE 40003, INT (bis/256)
40 POKE 40000, von-256*INT (von
/256): POKE 40001, INT (von/256)
50 POKE 40004, qual
60 CLS : PRINT AT 10,7: "NF-Sig
nal wird eingelesen": RANDOMIZE
USR 40006
70 CLS
80 RANDOMIZE USR 40084: INPUT
"Sprachausgabe wiederholen?": JS:
IF JS="J" THEN GO TO 80
90 INPUT "Neu einlesen in den
gleichen Speicherbereich?": JS: I
F JS="J" THEN GO TO 60
100 INPUT "Einlesen in einen ne
uen Speicherbereich?": JS: IF JS=
"J" THEN GO TO 10
110 STOP
9999 SAVE "Speaker" LINE 0: SAVE
"M-Code"CODE 40000,150

```

② **Hilfsprogramm:** Im Dialogbetrieb werden die Übergabevariablen für das Maschinenprogramm mit Werten versorgt