

Programmierbeispiel einer Tastenbelegung für Spectran NF

Dieses Programm muss mittels des Dateimanagers in der LCS-Software als Datei 100 auf dem Spectran NF abgespeichert werden.

Durch Druck auf die Taste 0 werden nun nacheinander 7 verschiedene Einstellungen geladen.

Diese Einstellungen und die Programmierung werden nun anhand des Programmcodes für Datei 100 erläutert.

(die Einstellungen können natürlich auch an eigene Bedürfnisse angepasst werden)

Zur Handhabung muss noch folgendes erwähnt werden:

Der Prozessor des Spectran kann 3 Werte in seinem Stack speichern. So können Rechenoperationen oder Vergleiche vorgenommen werden (siehe Programmers Guide).

Die Erläuterungen (beginnend mit der Raute #) gehören nicht zum Programm, sie dienen nur zur Erklärung.

Das gleiche Programm steht als eigene Datei zur Verfügung (Taste0.fsi). Es muss daher nicht mehr abgetippt werden.

Laden des Programms in den Spectran:

Voraussetzung: installierte Spectran-Treiber und LCS-Software auf einem Windows-PC

Schließen Sie den Spectran an das USB-Kabel an, schalten Sie ihn ein und starten Sie die LCS-Software.

Klicken Sie im Menü auf „Extras“ und dann auf „Dateimanager“. Es erscheint ein Fenster mit dem LCS-Dateisystem auf dem linken Seite, rechts finden Sie die ID, die Beschreibung, den Inhalt der Dateien und einige Steuerknöpfe (Buttons) zur Bedienung.

Links müsste die „Taste 0“ markiert sein, rechts dürfte noch kein Inhalt zu sehen sein.

Klicken Sie nun auf „Verbinden“. Sogleich erscheint der Inhalt der Datei 100, die aus dem Spectran geladen wurde. Die Datei 100 ist für die Taste 0 „zuständig“.

Um nun das neue Programm in den Spectran zu speichern, klicken Sie auf „Laden“ und wählen den Ordner aus, in dem die Datei „Taste0.fsi“ abgelegt ist. Dann wählen Sie die Datei aus und klicken auf „Öffnen“. Sollte sich die Datei nicht öffnen lassen (es erscheint kein Inhalt), dann müssen Sie noch einen kleinen Umweg gehen. Öffnen Sie stattdessen die Datei „Taste0.txt“ mit einem Editor (z.B. Windows Notepad), markieren und kopieren den gesamten Text und fügen diesen im Inhaltfenster des Dateimanagers ein (per rechtem Mausklick).

Der Programmcode sollte nun sichtbar sein. Jetzt klicken Sie auf „Übersetzen“. Wenn ein Fehler im Programmcode ist, wird eingeblendet, wo dieser Fehler liegt (Zeile & Spalte). Ansonsten wird der „Schreiben“-Button aktiviert, den Sie als nächstes anklicken müssen. Wurde die Datei ordnungsgemäß im Spectran gespeichert, erscheint als Meldung „Erfolg“ bei „Beschreibung“.

Die Tastenbelegung ist nun geändert und sofort verfügbar.

Erläuterung zum Programm (Datei 100):

Das Programm beginnt zunächst mit einer Abfrage der Variablen 05. Diese wird hier benutzt, um den momentan eingestellten Bereich abzufragen. (Variable 05 wird ansonsten nicht genutzt)

LOADVAR 05h	# => <u>Beginn Abfragebereich</u>
LDSHCO 00h	# Der Wert der Variable 05 in den Stack wird in den Stack geladen.
	# Dann wird der Wert 0 in den Stack geladen.

JUMPEQ LABEL-1	# Die beiden Werte werden verglichen. Sind sie gleich, springt das # Programm zur Sprungmarke LABEL-1. Bei ungleichen Werten läuft # das Programm einfach weiter.
LOADVAR 05h	# Wieder Wert der Variable 05 in den Stack laden.
LDSHCO 01h	# Jetzt Wert 1 in Stack laden.
JUMPEQ LABEL-2	# Vergleich, evtl. Sprung nach LABEL-2
LOADVAR 05h	# Weitere Vergleiche mit steigenden Werten...
LDSHCO 02h	# (Das Programm kann nach Bedarf erweitert bzw. gekürzt werden.)
JUMPEQ LABEL-3	
LOADVAR 05h	
LDSHCO 03h	
JUMPEQ LABEL-4	
LOADVAR 05h	
LDSHCO 04h	
JUMPEQ LABEL-5	
LOADVAR 05h	
LDSHCO 05h	
JUMPEQ LABEL-6	
LOADVAR 05h	
LDSHCO 06h	
JUMPEQ LABEL-7	
LOADVAR 05h	
LDSHCO 07h	
JUMPEQ LABEL-8	# Beim Wert 7 in Variable 05 Sprung nach LABEL-8
JUMP LABEL-9	# Ist der Wert in Variable 05 aus irgendeinem Grund anders (kein # Sprung bisher), springe nach LABEL-9. # => <u>Ende des Abfragebereiches</u>
	# => <u>Beginn des Sprungbereichs</u>
LABEL-1:	# Sprungmarke LABEL-1 (Namen der Sprungmarken mit # Doppelpunkt abschließen
AJUMP LABEL-10	# Hier wird der erweiterte Sprungbefehl AJUMP benutzt, um zum # LABEL-10 zu gelangen. Dies ist notwendig, wenn zwischen dem # dem Sprungbefehl JUMP und der Sprungmarke mehr als 127 Bytes # Programmcode liegen.
LABEL-2:	# Entsprechende Sprünge für die weiteren Werte...
AJUMP LABEL-11	
LABEL-3:	
AJUMP LABEL-12	
LABEL-4:	
AJUMP LABEL-13	
LABEL-5:	
AJUMP LABEL-14	
LABEL-6:	
AJUMP LABEL-15	
LABEL-7:	
AJUMP LABEL-16	
LABEL-8:	
AJUMP LABEL-17	# => <u>Endes des Sprungbereichs</u>
LABEL-9:	# Bereich, wenn bei der Abfrage am Anfang kein gültiger Wert # vorhanden war. # Diese Marke wird auch nach der Einstellung jedes Bereiches # angesprungen.
LOADVAR 05h	# Wert von Variable 05 laden.
LDSHCO 01h	# Wert 1 laden.
ADD	# Werte addieren (um 1 erhöhen).

```

STOREVAR 05h          # neuen Wert in Variable 05 speichern.
LOADVAR 05h           # Wert von Variable 05 wieder laden.
LDSHCO 08h            # Wert 8 laden.
JUMPGE LABEL-18        # Wenn Werte gleich oder größer sind, springe nach LABEL-18.
STOP                  # Ansonsten Programm hier beenden.
LABEL-18:
AJUMP LABEL-19         # Erweiterter Sprung nach LABEL-19.

# => Beginn des Einstellungsbereichs
LABEL-10:              # Sprungmarke für 1. Bereich.
PRINTSTR 03h, "NETZ"   # Ausgabe von „NETZ“ in der Info-Anzeige des Displays.
SETUP                  # Einleitung, um mehrere Parameter mit ihren Namen zu ändern.
    RESBANDW:13 VIDBANDW:10      # Einstellungen und Namen
    SWEPTIME:1000 ATTENFAC:-10   # dem ProgrammersGuide entnehmen.
    REFLEVEL:-1 DISPRANGE:1
    DISPUNIT:4 PULSEMODE:0      # Besonderheit: PULSEMODE = SENSMODE !
    DEMODMODE:0 SPECPROC:0
    13:0 14:0                  # Direkte Programmierung der Register 13 und 14.
LDCONST 0.045          # Um kleinere Werte (unter 1 kHz) für z.B. die Start- oder Stop-
STORESET STARTFREQ     # frequenz einzustellen, muss mit einem Trick gearbeitet werden.
LDCONST 0.065          # Man lädt mit LDCONST den Wert in den Stack und speichert
STORESET STOPFREQ      # ihn dann in die entsprechende Umgebungsvariable.
CLRFLD 00h             # Die direkte Eingabe STARTFREQ:0.045 funktioniert nicht !
AJUMP LABEL-9          # Springe nun nach LABEL-9.
LABEL-11:              # Sprungmarke für 2. Bereich.
PRINTSTR 03h, "BAHN"   # Text „BAHN“ wird angezeigt.
SETUP                  # Einstellungen vornehmen mittels Umgebungsvariablen...
    RESBANDW:14 VIDBANDW:10
    SWEPTIME:3000 ATTENFAC:-10
    REFLEVEL:-1 DISPRANGE:1
    DISPUNIT:4 PULSEMODE:0
    DEMODMODE:0 SPECPROC:0
    13:0 14:0              # Direktes Ansprechen von Registern.
LDSHCO 01h
STORESET SPECDISP
LDCONST 0.015          # Manuelles Setzen der Start- und Stopfrequenz
STORESET STARTFREQ
LDCONST 0.030
STORESET STOPFREQ
CLRFLD 00h             # Grafik-Display (Pixel-Feld) „leeren“.
AJUMP LABEL-9
LABEL-12:              # 3. Bereich ... (ähnlich wie 1. und 2. Bereich)
PRINTSTR 03h, "TFT+SN"
CLRFLD 00h
SETUP
    STARTFREQ:20 STOPFREQ:70    # Direkte Einstellung möglich, da Frequenz im kHz-Bereich
    RESBANDW:7 VIDBANDW:7
    SWEPTIME:200 ATTENFAC:-10
    REFLEVEL:-1 DISPRANGE:1
    PULSEMODE:0
LDSHCO 01h
STORESET SPECDISP
AJUMP LABEL-9
LABEL-13:

```

```

PRINTSTR 03h, "ELAMPE"
CLRFLD 00h
SETUP
  STARTFREQ:20 STOPFREQ:80
  RESBANDW:7 VIDBANDW:7
  SWEEPTIME:200 ATTENFAC:-10
  REFLEVEL:-1 DISPRANGE:1
  PULSEMODE:0
LDSHCO 01h
STORESET SPECDISP
AJUMP LABEL-9
LABEL-14:
PRINTSTR 03h, "AnlgTV"
CLRFLD 00h
SETUP
  RESBANDW:9 VIDBANDW:9
  SWEEPTIME:100 ATTENFAC:-10
  REFLEVEL:-1 DISPRANGE:1
  PULSEMODE:0
LDSHCO 01h
STORESET SPECDISP
LDCONST 30.5
STORESET STARTFREQ
LDCONST 32.5
STORESET STOPFREQ
AJUMP LABEL-9
LABEL-15:
PRINTSTR 03h, "ADSL"
CLRFLD 00h
SETUP
  STARTFREQ:138 STOPFREQ:1104
  RESBANDW:7 VIDBANDW:7
  SWEEPTIME:1000 ATTENFAC:-10
  REFLEVEL:-1 DISPRANGE:1
  PULSEMODE:0 13:2          # XY-Mag-Sensor, 3D-Messung
  14:0
LDSHCO 01h
STORESET SPECDISP
AJUMP LABEL-9
LABEL-16:
PRINTSTR 03h, "ADSL2+"
CLRFLD 00h
SETUP
  STARTFREQ:138 STOPFREQ:2208
  RESBANDW:7 VIDBANDW:7
  SWEEPTIME:1000 ATTENFAC:-10
  REFLEVEL:-1 DISPRANGE:1
  PULSEMODE:0 13:2
  14:0
LDSHCO 01h
STORESET SPECDISP
AJUMP LABEL-9
LABEL-17:
PRINTSTR 03h, "ISDN"

```

```

CLRFLD 00h
SETUP
  RESBANDW:8 VIDBANDW:8
  SWEPTIME:1000 ATTENFAC:-10
  REFLEVEL:-1 DISPRANGE:1
  PULSEMODE:0 13:2
  14:0
LDSHCO 01h
STORESET SPECDISP
LDCONST 0.3
STORESET STARTFREQ
LDCONST 130
STORESET STOPFREQ
AJUMP LABEL-9      # => Ende des Einstellungsbereichs
LABEL-19:          # Sprungmarke, um wieder zum Anfang der Bereiche zu schalten.
LDSHCO 00h         # Wert 0 in Stack laden.
STOREVAR 05h       # Stackwert in Variable 05 speichern.
STOP               # Programm beenden.

```

Das Programm ist in 3 Bereiche aufgeteilt: Abfragebereich, Sprungbereich und Einstellungsbereich.

Um das Programm zu erweitern, müssen im Abfragebereich weitere Abfragen eingefügt werden, die die Variable 05 mit höheren Werten vergleichen und entsprechende Sprünge in den Sprungbereich ausführen.

Der Sprungbereich muss in diesem Programm vorhanden sein, da die letzten Einstellungsbereiche mehr als 127 Byte vom JUMP-Befehl „entfernt“ sind. Durch Verwendung der Zwischensprünge mithilfe des Sprungbereiches wird dieses Problem umgangen.

Wird das Programm gekürzt (auf z.B. 3 Einstellungsbereiche), kann man direkte Sprünge verwenden. Dann kann der Sprungbereich entfallen, die Sprungbefehle JUMPEQ müssen dann die entsprechenden Sprungmarken des Einstellungsbereichs direkt anspringen.

Noch einige Hinweise:

Sollte ein selbst erstelltes Programm nicht funktionieren, schalten Sie den Spectran aus und schließen Sie den Dateimanager. Schalten Sie den Spectran wieder ein, drücken sofort auf die CLR-Taste und „verbinden“ die LCS-Software wieder mit dem Spectran. Danach starten Sie den Dateimanager und lesen das Programm aus. Evtl. sehen Sie nun Fehler im Programm, die vorher nicht erschienen sind. Die Anzeige der Programme im Dateimanager wird nicht immer sofort aktualisiert, daher ist die hier beschriebene Methode die sicherste, um Fehler zu finden.

Falls einmal gar nichts mehr funktioniert, wie es einmal war, führen Sie einen Factory Reset durch. Hierbei werden alle Einstellungen wieder in den Lieferzustand zurückgesetzt. Evtl. geänderte Tastenbelegungen werden hierbei auch gelöscht bzw. von den Standardbelegungen überschrieben. Sie sollten daher am besten Ihre selbst erstellten Programme als Datei auf Ihrem PC speichern, um Zeit zu sparen und größere Programme nicht mehrfach eintippen zu müssen.

In den Beschreibungen im ProgrammersGuide finden Sie die Variablen-Bezeichnungen für Spectran HF V1-3, HF V4 und NF.

Diese lauten zum Teil unterschiedlich, belegen aber die gleiche Adresse.

Beispiel: HF Adresse 0x0A PULSEMODE
 NF Adresse 0x0A SENSMODE

Der Dateimanager kann SENSEMODE nicht übersetzen. Verwenden Sie stattdessen (wie oben im Programm) den Begriff PULSEMODE oder direkt die Adresse 0Ah bzw. 10. Beim erneuten Einlesen des Programms in den Dateimanager kann Ihnen aber wieder PULSEMODE angezeigt werden. Die Vergabe der Label-Namen wird durch den Dateimanager vorgenommen. Sie können eigene Labelnamen vergeben, die nach dem nächsten Einlesen aber evtl. anders lauten. Das Sprungziel selbst ist natürlich richtig.

Für weitere Programmierungen und Variablennamen ziehen Sie bitte den ProgrammersGuide zu Rate. Dort sind entsprechende Erläuterungen zu finden.

Viel Erfolg beim Messen mit dem Spectran NF

wünscht Ihnen

Ihr Aaronia AG - Team