



**H**erzlich willkommen beim ersten Teil unserer Serie über Quick. Diese neue Programmiersprache für alle kleinen Ataris wurde im Auftrag des **ATARImagazins** exklusiv für die Leser entwickelt. Quick ist fast so schnell und leistungsfähig wie Assembler, aber auch annähernd so komfortabel und einfach wie Basic.

In der ersten Folge werden wir Ihnen Quick nicht nur vorstellen und seine Benutzung erklären; wir wollen Ihnen auch einen Eindruck davon vermitteln, wie eine solche Sprache dem Computer "beigebracht" wird. Außerdem präsentieren wir Ihnen die ersten beiden Programmteile.

## 8 Bit

### Von der Idee zum Konzept

Wer sich mit seinem Atari einige Zeit beschäftigt, wird bald an die Grenzen des eingebauten Basic stoßen. Leistungsfähige Programme lassen sich damit nicht verwirklichen. Es ist viel zu langsam; außerdem können die umfangreichen Hardware-Fähigkeiten (z.B. Interrupts) der Ataris überhaupt nicht voll genutzt werden. Wollte man seinen Computer so richtig ausreizen, blieb bisher nur der Schritt zu Assembler.

Maschinensprache ist zwar schnell und leistungsfähig, ihre Programmierung erfordert aber einen immensen Zeitaufwand. Bereits die Lösung einfacher Aufgaben ist mit aufwendiger Tüftelei verbunden. Assembler ist eben keine Hochsprache, und somit sind auch keine fertigen

Problemlösungen vorhanden. Schon so einfache Dinge wie PRINT, PLOT, SOUND oder LOAD bedeuten viel Arbeit und ziehen oft eine langwierige Fehlersuche nach sich.

So ist es nur zu verständlich, daß viele resignieren und die MC-Programmierung anderen überlassen. Deshalb lag auch die Idee nahe, eine Sprache zu entwickeln, welche die Vorteile von Assembler und Basic vereinigt. Sie muß schnell und universell einsetzbar sein und alle Hardware-Möglichkeiten der Ataris zugänglich machen. Sie soll sich aber auch komfortabel und strukturiert programmieren lassen.

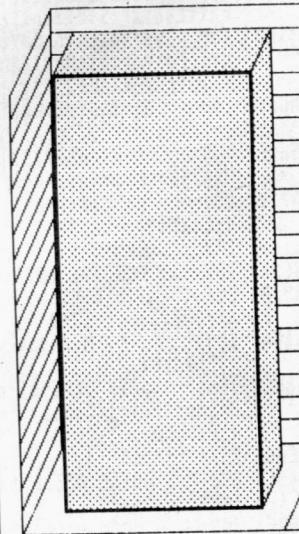
Bevor wir uns nun an die Definition der Sprache machen konnten, war zunächst ihr Haupteinsatzgebiet zu klären. Wir entschieden uns, sie auf die Programmierung von Spielen, Sounds, Grafiken und allen damit verbundenen Anwendungen (also durchaus auch Grafik- und Textverarbeitungsprogramme) zuzuschneiden. (Hier liegt auch der Grund dafür, daß es in Quick Befehle zur Verschiebung von Grafikausschnitten, zum Spielen von digitalisierten Sounds, zur Darstellung und Bewegung von Playern und sogar zur Abfrage einer Maus gibt.) Natürlich sind aber auch Strukturen wie IF...ELSE...ENDIF oder REPEAT...UNTIL notwendig.

Einen weiteren Beitrag zur Strukturiertheit liefert die Möglichkeit der Deklaration von Variablen. Man kann somit verschiedene Arten von ihnen verwenden. Dies sind Integer-Variablen, also solche, die nur ganze Zahlen darstellen können (dabei läßt sich noch zwischen Byte und Word unterscheiden), und Arrays, eine Art String-Variablen. Ein wichtiges Merkmal ist hier auch die Möglichkeit, lokale Variablen zu verwenden und an Prozeduren zu übergeben. Diese Variablen lassen sich dann nur in dem Unterprogramm verwenden, in dem sie deklariert wurden.

# QUICK

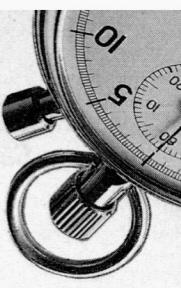
Wie Sie vielleicht bemerkten, fehlen bisher noch die Fließkommavariablen. (Ausschließlich sie können in Basic verwendet werden, woraus auch niedrige Geschwindigkeit resultiert.) Diese Art von Variablen ist jedoch für die meisten Anwendungsbereiche überhaupt nicht notwendig. Was aber, wenn doch einmal ein Programm schreiben möchte, in dem Fließkommazahlen braucht? Mit sind wir jetzt auf die wichtigste Forderung gestoßen: Quick muß leicht erweiterbar sein.

### QUICK Benchmark-Test



Definition eigener Befehle also einfach möglich sein, haben auch bereits einige Programme in Quick geschrieben! Die Fließkommazahlen werden!

Nun war noch ein weiterer scheidender Punkt zu

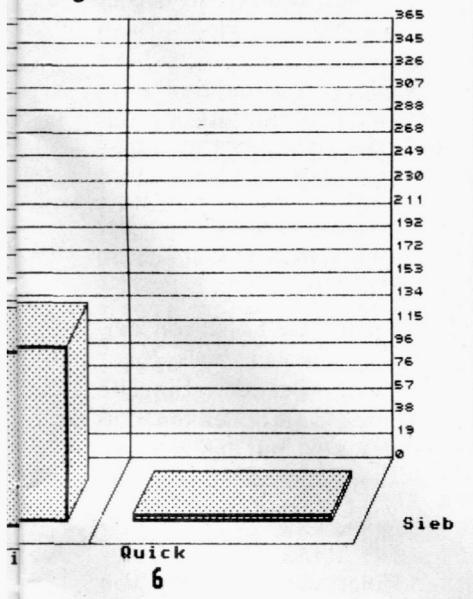


# ICK Die neue Sprache

## Das Quick-Programmiersystem

Wie ist die Sprache in der Lage, so schnelle Programme zu produzieren? Dazu mußte Quick als Compiler-Sprache konzipiert werden, d.h., man schreibt zunächst mit Hilfe eines Texteditors einen Quick-Quelltext, der dann vom Compiler in ein direkt lauffähiges Maschinenprogramm zu übersetzen ist. Dabei ist es natürlich von entscheidender Bedeutung, wie dies gelingt. Befehle wie die Addition zweier Variablen oder das Einlesen von Daten werden dabei so gut übersetzt, daß es in Assembler auch

**Nur 1/25 der Zeit die "Turbo Basic" brauchte, rechnete "Quick". Im Vergleich mit Atari-Basic ist es sogar nur 1/60!**



nicht schneller gehen kann. Andere Aufgaben (z.B. Vergleiche) müssen dagegen so flexibel sein, daß hier Assembler (wo man sich ja nur um einen bestimmten Fall kümmern muß) sicher Geschwindigkeitsvorteile bringen würde.

Wie Sie bemerkt haben, muß man immer zwischen zwei Programmen (Editor und Compiler) wechseln. Damit dieser Vorgang nicht zu lange dauert, dürfen die beiden nur einmal zu Beginn geladen werden und müssen dann jederzeit im Speicher vorliegen. Es fehlt uns also noch ein drittes Programm, das den Editor und den Compiler zu Anfang lädt und anschließend deren Verbindung darstellt. Dies ist die Shell. Damit Editor und Compiler dann aber nicht den ganzen Speicher belegen, sind sie abwechselnd in den freien Speicher hinter dem Betriebssystem zu kopieren, wenn sie nicht gebraucht werden. Auch dafür ist die Shell zuständig.

Die Erweiterbarkeit von Quick ist durch Libraries (Unterprogrammbibliotheken) gegeben. Eine solche Bibliothek ist ein Quick-Quelltext, der eine Anzahl verschiedener Unterprogramme enthält. Einige Standard-Libraries gehören von Haus aus bereits dazu. Dabei handelt es sich um die Grafik-Library (enthält Routinen zum Zeichnen verschiedener geometrischer Objekte und einen FILL-Befehl), die Mathe-Library (bindet die Fließkommaarithmetik ein) und einige andere. Natürlich können auch Sie Routinen in Quick schreiben und in einer Bibliothek ablegen. Wird eine solche Routine dann in einem Programm verwendet, kann man den Compiler veranlassen, die entsprechende Library zu laden.

Das Quick-Programmiersystem sieht dann also folgendermaßen aus:

- Editor zum Schreiben der Quelltexte

- Compiler zur Produktion der lauffähigen Programme

- Shell zur Verbindung von Editor und Compiler

- Libraries für eine einfache Erweiterung

Jetzt wollen wir Sie aber nicht länger auf die Folter spannen. Hier nun der erste Quick-Quelltext:

```
MAIN  
PRINT ("Hallo Welt")  
ENDMAIN
```

Wie Sie sehen, hat das durchaus Ähnlichkeit mit Basic (oder C). Sind die Quick-Programme nun aber wirklich auch schnell? Dazu haben wir einen kleinen Benchmark-Test geschrieben. Mit dem "Sieb des Eratosthenes" sollten die ersten 1889 Primzahlen ermittelt werden. Was dabei herauskam, entnehmen Sie bitte der Hardcopy.

Wir haben also nicht zuviel versprochen. Quick hat die Nase weit vorn. Doch nun zum praktischen Teil.

### Die Shell

Sie besteht aus einem kurzen Programm, das direkt hinter dem DOS in den Speicher geladen wird. Tippen Sie einfach Listing 1 mit der AMD ab, und speichern Sie es als AUTORUN.SYS auf einer Diskette ab, auf der sich das DOS 2.5 befindet. Nach dem Booten der Disk (dabei OPTION drücken!) befinden Sie sich in der Shell, die zunächst den

Editor und den Compiler (soweit bereits vorhanden) nachlädt. Jetzt können Sie die Systemdisk aus dem Laufwerk nehmen und eine Arbeitsdisk einlegen, auf der sich dann Programme abspeichern lassen.

Durch Eingabe von *Editor* oder *Compiler* rufen Sie das entsprechende Programm auf. Die Tastenkombination CONTROL-Q löst einen Kaltstart aus.

### **Der Quick-Editor**

Der erste Schritt zur Erstellung eigener Programme ist das Schreiben eines Quick-Quelltextes mit Hilfe des Quick-Editors. Der Quelltext wird dann vom Compiler in ein lauffähiges Programm umgewandelt.

Der Editor ist als Listing 2 abgedruckt. Speichern Sie das Programm als EDITOR.OBJ auf der Systemdiskette ab.

### **Aufbau des Editors**

Der Quick-Editor unterscheidet sich in der Handhabung stark von den üblichen zeilennummernorientierten Editoren für die XL-Computer; er kommt völlig ohne Zeilenummern aus. Sie werden feststellen, daß dies viele Vorteile hat.

Ein weiteres wichtiges Merkmal ist, daß sich der Editor ständig im Insert-Modus befindet. Alle Zeichen, die Sie eintippen, werden an der Position des Cursors eingefügt, wobei die weiter rechts stehenden verschoben werden. Der Editor arbeitet zeilenorientiert, d.h., pro Zeile lassen sich höchstens 38 Zeichen eingeben. Am Ende der Zeile bleibt der Cursor automatisch stehen.

Zu Beginn legt der Editor immer einen Textkopf an, in dem die Länge des Textes (am Anfang natürlich 0 Bytes) und der freie Speicherplatz in Hex-Zahlen angezeigt werden. Sobald Sie bei SAVE einen File-Namen eintip-

pen, wird dieser ebenfalls im Textkopf dargestellt.

In der untersten Zeile des Bildschirms erscheint nun *Edit*.

### **Der Edit-Modus**

Wenn Sie sich in diesem Modus befinden, können Sie drauflostippen. Dabei sind sowohl Groß- als auch Kleinbuchstaben, jedoch keine inversen oder Grafikzeichen möglich. Mit Hilfe der RETURN-Taste erzeugen Sie ein Zeilenende. Dann wird der Cursor an den Anfang der nächsten Zeile gesetzt. Wenn Sie inmitten einer Zeile RETURN drücken, teilen Sie diese, und der Rest wird in die folgende Zeile geschoben. In der 38. Spalte läßt sich RETURN nicht mehr betätigen!

Mit den Pfeiltasten CONTROL-(+ - = \*) können Sie den Cursor im Text bewegen. Das obere und untere Ende des Textes sind durch zwei Pfeile gekennzeichnet. Der Cursor kann diese Markierungen nicht überschreiten. Innerhalb einer Zeile läßt er sich nur bis zum RETURN-Zeichen nach rechts bewegen.

Die Taste BACK SPACE bringt den Cursor um ein Zeichen nach links und löscht dabei das dort stehende Zeichen. Dabei ist es nicht möglich, über das linke Zeilende hinauszugehen. CONTROL DELETE entfernt das Zeichen rechts vom Cursor, wobei der Zeilenrest nachgeschoben wird. Das (unsichtbare) RETURN-Zeichen am Ende einer Zeile läßt sich so jedoch nicht löschen. TAB fügt zwei Leerzeichen ein. Damit können Sie Ihren Programmen einen übersichtlicheren Aufbau verleihen.

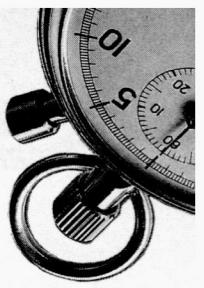
### **Die CONTROL-Sequenzen**

Durch gleichzeitige Betätigung von CONTROL und einer der folgenden Tasten lassen sich zusätzliche Funktionen aufrufen:

X: Löschen der Zeile, in der

	sich der Cursor befindet.
V:	Vereinigen einer Zeile mit der folgenden. Diese Funktion ist nur ausführbar, wenn beide Zeilen zusammen nicht mehr als 38 Zeichen umfassen.
H:	Bewegt den Cursor an den Textanfang.
N:	Bewegt den Cursor ans Textende.
U:	Bewegt den Cursor eine Seite nach oben.
D:	Bewegt den Cursor eine Seite nach unten.
:	Bewegt den Cursor ans Zeilenende.
Clr:	Löscht den Text nach Rückfrage.
B:	Setzt den Blockanfang (zeilenweise) und löscht das Blockende.
E:	Setzt das Blockende. Die Länge des Blocks darf dabei nicht mehr als 3 KByte betragen. Der Block wird dann in einen internen Buffer kopiert, so daß er unverändert bleibt, auch wenn der eigentliche Block weiter editiert wird. In der Statuszeile erscheint nun "Edit-Block defined".
C:	Kopiert den Block an der Position des Cursors in den Text.
F:	Sucht eine Folge von Zeichen, wobei ? als Joker dient. Bei der Eingabe müssen Sie bedenken, daß der Text nur bis zum ersten Leerzeichen beachtet wird. R wiederholt die Suche.
I:	Zeigt die Directory von Laufwerk 1 an. Am Ende müssen Sie eine Taste drücken.
S:	Speichert den gesamten Text ab. Dabei können Sie entweder einen beliebigen File-Namen (D:XXXX.QIK) eintippen oder, falls Sie das schon einmal getan

**AMD**  
S.65



haben, mit RETURN den gleichen Namen wieder verwenden. Mit dieser Funktion läßt sich der Text auch ausdrucken, wenn Sie beim Namen P: angeben.

L: Läßt einen Text. Dabei wird ein im Speicher stehender gelöscht. Als File-Name wird der bei SAVE angegebene oder der zuletzt im Compiler verwendete benutzt.

M: Läßt einen Text und hängt ihn an das Ende des aktuellen an. Bedenken Sie, daß Definitionen globaler Variablen dann aber an den Anfang des Textes kopiert werden müssen (Näheres dazu im nächsten Teil).

O: Zeigt einen Hilfstext, der die Steuertastenbelegung angibt. Durch Drücken einer Taste gelangen Sie wieder in den Edit-Modus.

Q: Verläßt den Editor und kehrt zur Shell zurück.

Wie Sie sehen, bietet der Editor trotz seiner Kürze eine Reihe von praktischen Funktionen, die das Schreiben von Quick-Texten sogar komfortabler gestalten als die Erstellung von Basic-Programmen.

Im nächsten Teil unserer Serie werden wir Ihnen den Compiler präsentieren. Dann können Sie bereits Ihre ersten Quick-Programme erzeugen.

Andreas Binner und Harald Schönfeld

## EDITOR.OBJ

```

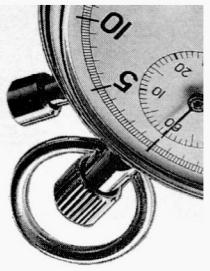
1000 MMMM RRYR TVUT IVHJ YTGR IYYM 32599
1001 YRHR IYRR UIRY RYRY RYRY 33882
1002 RYRY RYRY RYIM IGYT IYRH UFRY 32488
1003 RYRY RYRY RYRY RYRY HRIY 33524
1004 DGYR ITRU YRUT GDFJ FUF C RBYD 30794
1005 FIFJ GIFM GYRR UFTT RNTU RRRH 32178
1006 FURJ RRTT TJTH TJRR UYFT FJFN 31639
1007 FIFM GYFF GU FM FFGI RRYD FIFJ 30246
1008 GIRR RRRR RRRR RRRR RRRR 32899
1009 RRRR RRRR RRRR RRRR RRRR 32929
1010 RRRR RRRR RRRR RRRR RRRR 32930
1011 RRYT GYFD RRGJ FMGD RRGU GDGY 30775
1012 FDRR RHGJ RMFN RJTM RRRR RRRR 32195
1013 RRRR RRRR RRRR RRRR RRRR 32933
1014 RRRR RRRR RRRR RRRR TKRR 32797
1015 RRRR RRRR RRRR RRRR RRRR 32935
1016 RRRR RRRR RRRR RRRR RRRR 32936
1017 RRRR RRRR RRRR RRRR RRYJ RMYM 32822
1018 RRYD GYGY FMGY RRRH GRGY FDGU 31207
1019 GURR FTRR FCFD GJRJ RRRR RRRR 31748
1020 RRRR RRRR RRRR RRRR RRRR 32940
1021 RRYD FIFJ GIRR RBRR YYFV FMFU 31527
1022 FCRR FIFD FFFJ FNFD FIRR RRRR 30807
1023 RRRR RRRR RRRR RRRR RRRR 32943
1024 RRRR RRRR RRYI FJGY FDFU GIFM 30742
1025 GYGJ RRFM FFRR FIGY FJGF FDRR 30397
1026 RUTT RRRR RRRR RRRR RRRR 32966
1027 RRRR RRRR RRRR RRRR RRRR 32947
1028 RRRR RRRR RRRR RRRR RRRR 32948
1029 RRRR RRRR RRRR RRRR RRRR 32949
1030 RRRR RRRR RRRR RRRR RRRR 32950
1031 RRMN JCJC JCJC JCJC JCJC 28860
1032 JCJC JCJC JCJC JCJC JCJC 28770
1033 JCJC MMKJ KRHB FYTD KJYT HBFU 30849
1034 TDKJ MMHB RCRF KJRR HBRB RFYR 30916
1035 VIYT KJRR HBRD RFYR DTYY YRVK 32466

```

1036 YKKJ RTHD NKKJ UHHD NCKD NKHD 29915
1037 VCKD NCHD VVYR UMYU YRRY YIIV 32966
1038 CUYT KYRR CBNI YTJB RRHH VJMM 31580
1039 MRRI NHIV VFYT KJRR THFJ JFHB 30637
1040 RJRF KJUH FJRR HBRK RFFF JCMM 30530
1041 JCJC DTGD FJFU FCYB DUFM GDGY 30195
1042 FUF D GIFT GHGI JCYR YRYR YRYR 32432
1043 YRYR YRYR YRYR YRYR YRYR YRYC 33326
1044 YBYB YBYB YBYB YBYB YBYB YBYB 31206
1045 YBYB YBYB JCIV FDFN FGGI FHUK 30024
1046 YRYI YRYR YRYR JCJC IFGY FDFD 30442
1047 YRYR UKYR YIYR YRYR YRJC YBYB 32403
1048 YBYB YBYB YBYB YBYB YBYB YBYB 31210
1049 YBYB JCJC JCJC MMFD RRKJ 29900
1050 RRHB YMRY KJRU HBUR RYKJ YRHB 31676
1051 UTRY KJRR HDDH KJUI HDDJ KJFH 29887
1052 HBRR RYKJ YNHB RTRY KRKT KYYN 32520
1053 KJRG YRDV NIKJ VRHB RNBI KJRR 31027
1054 HBVD RYKJ NNHB VFRY KJRR HDDY 31206
1055 HDDG YRCY YYKJ RTHB MRRY KJYR 32205
1056 YRKI MTKJ RTHB MURY KJRM HBBJ 30741
1057 RYKJ RUHB BKRY KJYY HBYM RYFR 31916
1058 KYRR HKJB RRUI JBRR UDJB RRU F 31104
1059 JBRR UGNH BRMT FRKB RJRF UHNJ 30923
1060 RRHB RNR F KBRK RFN J UHHB RMRF 30701
1061 KBRN RFUH NJJD HBRN RFB RMRF 30536
1062 NJRR HBRM RFKR IYR RUYU KJRR 32393
1063 UHNB RJRF HBRN RFKJ CRNB RKR 30558
1064 HBRM RFKR IMYR RUYU FRKB RMRF 31668
1065 YRTR YUKB RNRF YRTR YUFR HBTR 32094
1066 RFIK IKIK IKKK CBYB YUJJ RRUH 31238
1067 VHKB TRRF YJRM KKCB YBYU JJRR 31500
1068 UHVH FRUR UTUY UUU UDUF UGUH 32182
1069 UJIT IYIU IIID IFFD RRYR VDYY 31843
1070 KJRR HBRY RFHB RBRF KRRR JHKK 30836
1071 HDVB KJUI HDVN YRMT YUCT VCVJ 32019
1072 FRCR RUUH NJYR HBRR RFV J CBR 30984
1073 IDNR RVBR TFKD VCHD NFKD VVHD 30479
1074 NGKD VBHD NHKD VNHD NJHV RIRF 30711
1075 NNRI RFKJ JCJT VBVH JHTH FDVC 30470
1076 HDVC KDV V FJRR HDVV KDV THFJ 30827
1077 YHHD VBK D VNFJ RRHD VNNH KRR 31392
1078 NRTJ MRYK YRMT YUIV DGYU KBRR 32131
1079 RFV J MMBR RUIV BRYU KBRR RFV J 31721
1080 MNBR RKKJ RTHB RYRF KJDB HBRR 30602
1081 RFKB RRRF JT B VHIV DGYU KJDV 31379
1082 JT B HNRT RFNR THCR TDKD VBTH 30947
1083 FJYH HDVB KDV FJRR HDVN YRMT 31687
1084 YUNH NRTH BRNC FRHV TCRF KRR 31375
1085 JHJT VBVH VRYH BRMJ KVTC RFFR 31279
1086 KJRR HBRH RFHB RVRF KBRI RFHB 30280
1087 RGRF YRMB MYB RRRF KYRR CBMY 31490
1088 YIVB RRRF MRRJ NHNV YNYD BRMY 31971
1089 IVUC YIHK RKK CBPF YDHB UJYI 30802
1090 CBRG YDHB UKYI IVRR YRKY RRCB 31747
1091 BJYI VBR RFMR RJNH NVMT YIBR 31697
1092 MYIV FUYI HKRK KKCB NTYI HBDN 30542
1093 YICB NYI HBDM YIYR RRYR IVTR 32515
1094 YIKJ RTHB RVRF KBRR RFYJ GMHB 30776
1095 RRRF YRGN YIKB RHRF MRRU IVBM 31389
1096 YCIV TRYI KBRI RFV J YHBR RFKJ 31097
1097 RRHB RHRF FRKD NHTH FJYF HDVB 30393
1098 KDNJ FJRR HDVN KDV THFJ RTHD 30669
1099 VCKD VNFJ RRHD VVKD NHTH FBRD 30510

►

1100 RFHD VMKD NJFJ RRHD BRKD VMUH 30793	1165 GMYG JHTH FDVC HDVC KDVV FJRR 30985
1101 NJRT HDVM KDBR NJRR HDBR YRNU 31456	1166 HDVV YRFR YCFR YRVJ YVKJ RRHB 31933
1102 YBKV RDRF KBRR RFVJ FRCR RUUH 31511	1167 RDRF FRYR YRYR YRYR YRYR YRYR 34087
1103 NJYR JTNA NNRD RFNN RIRF FRGN 31006	1168 YRYR YRYR YRYR YRYR YRYR YRNU 34085
1104 MNTN TMGM GCRY RDNM YCIH YVJU 31760	1169 YVIV FRYG YRNU YVYR KVUR KYRR 33368
1105 YVJB YVUG YVHJ YVNB YDRF YFRH 31753	1170 CBYM YDJB KGYR NHNR RDBR MDKJ 30744
1106 TVTB JCTH RHRN TDRI TFGB TURV 31588	1171 IRHB CNRY KJRR HBCF RYKY RRCB 31107
1107 RBRF TYRU RMRJ TTRT VJYV KBYV 32830	1172 TDUH UHNJ YRMR RHJB KBYR NHNR 31449
1108 BGYC CBYC TCYC UTYC GHYC JVYC 31327	1173 TRBR MRHK THFJ RFHB TKRF YRBI 30893
1109 NGYK JVYK NKYG URYJ NDYJ UYYG 32521	1174 YHMR RFYR VKYK IVCR YHYR VKYK 32641
1110 NIYG HJYF BTYD FCYD INYD IUYD 31158	1175 YRHG YKKY RRCB KBYR JBTD UHJB 30661
1111 TIUU YTUF YDRA YVYM YTYI RRYB 32996	1176 KIRF NHNR TRBR MYKJ IRHB RNBI 30852
1112 YDUY YGYD YFYJ YNYI RRKR GNKY 32306	1177 KJKB HDMG KJYR HDMH KJRT HDMR 30879
1113 YIYR HRUT YRVK YKFR YRNU YVKJ 32822	1178 KJRH HDMT KJRR HDMY YRIV URUR 32525
1114 GMHB YKYL KJYR HBYC YRYR MBMY 32104	1179 DGKJ RRHD MDKJ UHHD MFKJ RRHD 30402
1115 YJDM VJDJ MRRI YRVK YKFR IVGU 31988	1180 VCKJ UHHD VVCT VCVJ MMMR RVHV 31639
1116 TVYR NUUV KJTM HBYK YRKJ YTHB 31786	1181 BRMG NFVV IVFM YHJH THFD VCHD 30540
1117 YCYR YRCH URRT RFRR IIUT UKYK 32336	1182 VCKD VVFJ RRHD VVKA VCUH NJRR 31560
1118 YNYK YRTR RUIV CFYH KJFM HDVC 30875	1183 HDMU KDVV NJUH HDMI KDMU THFJ 30806
1119 KJYT HDVV YRUM YUKJ RRHD DIHD 31027	1184 RTHD MUKD MIFJ RRHD MIYR HVUR 31744
1120 DDYR MFUR RTRR VMYD KDYU VJHH 31888	1185 URRN YRKV URKJ RRHB RDRF KJVR 31578
1121 MRTN KDYU TRRU IVCF YHKB VMYD 31600	1186 HBRN BIFR KJVR HBRN BIYR KVUR 31801
1122 YRKI MTKD DIVJ TFBR NRKJ MMHB 30566	1187 KJVM HBYK YRKJ YRHB YCYR YRMB 31955
1123 MVRY YRMB MYIV HKYD YRKV URYR 32837	1188 MYYR VKYK FRKJ RFHB TKRF KNTK 31246
1124 VKYK IVBM YDRA RRYR NUUV KJBJ 32024	1189 RFKJ RRJB KGYR NHNR THBR MHKJ 31026
1125 HDVC KJYN HDVV YRUM YUKJ MMHB 31633	1190 KGBH YKYL KJYR HBYC YRKN TKRF 31768
1126 MVRY YRMB MYKJ MMHB MVRY FRKD 31349	1191 HNTT RFYR MBMY KNTT RFVJ JCMR 31642
1127 NFHB TYRF HBTI RFKD NGHB TURF 30725	1192 YMVJ TCML YNVJ GNMR TKVJ YRJR 32337
1128 HBTB RFKJ RTHB THRF YRKV YKFR 31776	1193 NGVJ GMCR NUNR TFMR BMVJ FRCR 31246
1129 KBTH RFVJ RTMR RTFR KDNF UHNB 31026	1194 RUUH NJYR JBKG YRNH IVNN YHNR 31901
1130 TYRF HBTB RFKD NGNB TURF HBTG 30561	1195 RFMR VBVK KJRR JBKG YRIV NNYH 31779
1131 RFVJ RVCR FUKB TFRF BRRD KBTG 30555	1196 KJRR FRKJ RTFR KJRT HDMR YRKV 32020
1132 RFMR DJKD NFHB TIRF KDNG HBTB 29947	1197 URKY RRCB UIYD JBKG YRNH NRRD 31253
1133 RFKJ RYHB THRF KJMG HBYK YRKJ 31173	1198 BRMD KJRF HBTB RFYR RRCB KIRF 30752
1134 YRHB YCYR KBTY RFHD VCKB TURF 31159	1199 VJII BRTK CBKI RFVJ YRMR RBUH 31437
1135 HDVV KJRR HDVB KJCY HDVN KRR 31408	1200 NJYR URRH JBKB YRNH NRTR BRNV 31854
1136 CTVC JTVA KDVC THFJ RTHD VCKD 30517	1201 HKFJ RDHB TKRF KJIR HBCN RYKJ 30825
1137 VVFB RRHD VVKA VBTH FJRT HDVB 30868	1202 RRHB CFYR YRBI YHMR RFYR VKYK 32317
1138 KDVN FJRR HDVN KDVN VBTD RFBR 31077	1203 IVCR YHYR CFYK YRHG YKKJ IRHB 31174
1139 BCKD VCVB TIRF BRBI FRFD RRKB 30250	1204 RNBI KJKB HDMG KJYR HDMH KJRT 30902
1140 THRF VJRY MRRF FRYR DYYK KDMI 31824	1205 HDMR KJRI HDMT KJRR HDMY YRIV 31911
1141 VBTG RFMR RDJR RKIV KHYF KDMU 31287	1206 URTR RUIV CFYH YRMF URDY HCRR 31800
1142 VBTF RFCR RTFR KJRR UHNB TFRF 31155	1207 UHKB YUVJ HHMR RIKD YUUR NVKD 31900
1143 HDVB KJCR NBTG RFHD VNKJ RRHD 30698	1208 YHTH FJRR HBRJ RFKD YJFJ UHHB 30510
1144 VCKJ CRHD VVKA NFUH NJRT HDVM 31207	1209 RKRF KBRJ RFTH FJUY HBRJ RFKB 30704
1145 KDNG NJRR HDBR YRNU YBKR RRKD 31537	1210 RKRF FJRR HBRK RFYR KVUR YRYU 32937
1146 NFHD VCKD NGHD VVKA RRHD VBKJ 30719	1211 YCKJ VRHB RNBI FRYR KVUR KYRR 32424
1147 CYHD VNKJ RRHD VMHD BRCT VBJT 30928	1212 CBUJ YDJB KGYR NHNR RDBR MDYR 31386
1148 VCKD VCTH FJRT HDVC KDVV FJRR 31204	1213 VMYH MRRI YRVK YKFR YRVK YKYL 33065
1149 HDVV KDVN THFJ RTHD VBKD VNFJ 30692	1214 HGKJ KJIR HBRN BIKJ KBHD MGKJ 29960
1150 RRHD VNKD VMTA FJRT HDVM KDBR 30824	1215 YRHD MHKJ RTHD MRKJ RIHD MTKJ 30900
1151 FJRR HDBR VBTG RFBR BRKD VMVB 30773	1216 RRHD MYYR IVUR TRRU IVCF YHYR 32576
1152 TFRF BRVJ KBRJ RFTH FBTB RFHB 30213	1217 MFUR FHRR RRRN YRDY YKYL FVUR 32950
1153 RJRF KBRK RFFF TGRF HBRK RFFF 30495	1218 TRRF KDYU VJHH BRNH KDYH THFD 30701
1154 YRNU YVKY RRCB UNYD JBKG YRNH 31414	1219 MDHB RJRF KDYJ FDMF HBRK RFYR 30976
1155 NRRD BRMD YRVM YHMR RIYR VKYK 32535	1220 KVUR KJVR HBRN BIFR KBRJ RFUH 30880
1156 FRYR VKYK YRHG YKKY RRCB KBYR 31699	1221 NJIF HDMK KBRK RFNJA RRHD MFKJ 30711
1157 JBBT YGNH NRTT BRMD FDRR KJRR 31280	1222 RRCT MDVJ MMMR TRKD MDTH FJRT 31320
1158 HBRV RFYR KBYV KDNK HDVC KDN 30454	1223 HDMK KDMF FJRR HDMF IVFU YKKJ 31062
1159 HDVV YRUM YUKD NFHD VCKD NGHD 30372	1224 RRHU NDMD HDMU KJCR NDMF HDMI 30172
1160 VVKA RRKY RRKB RRHB TJRF CBBT 30757	1225 FRKY RRCB KBYR VJFR CRRU THFJ 31381
1161 YGVJ YRMR YBVJ UMMR TTHB TRRF 31922	1226 YRJB KBYR NHNR TTBR NNFR YRNU 32238
1162 CTVC VJMM MRUR VBTR RFMR RUIV 32284	1227 YVKJ GMHB YKYL KJYR HBYC YRYR 32363
1163 KHYG NNTJ RFNH VHIV HIYJ JHUU 31202	1228 MBMY YJDM VJDJ MRRI YRVK YKFR 32084
1164 NBTJ RFKH KYRR VHBR VVNE VVIV 32273	1229 YRVI YTYR VKYK KJRT HDNK KJUH 31605



1230	HDNC	KJRR	HBRD	RFFR	KBTH	RFVJ	30887	1295	JCMR	RIHH	IVVC	YBVH	JHTH	FDVC	30847
1231	RYMR	RCKJ	DGHB	YKYR	KJYR	HBYC	31303	1296	HDNK	KDVV	FJRR	HDNC	FRYR	TIYN	31881
1232	YRFR	KJMG	HBYK	YRKJ	YRHB	YCYR	31917	1297	KRRR	CTVB	JTVC	KDVC	UHNJ	RTHD	31164
1233	FRKR	RRKY	RYCT	NFVJ	MMMR	THVJ	32028	1298	VCKD	VVNJ	RRHD	VVKD	VBUH	NJRT	31776
1234	JCBR	RUVK	MRRJ	VHIV	NCYK	JHUH	31580	1299	HDVB	KDVN	NJRR	HDVN	VDBR	BRBN	30871
1235	NBRG	RFTH	FBRI	RFVJ	YHJR	RTFR	31644	1300	KDVB	VDVM	BRBH	YRYY	YNFR	KRRR	32494
1236	VNRI	RFKJ	RTHB	RVRF	YRNU	YVKJ	32403	1301	CJVC	RRJJ	URYN	VHVR	RFBR	MDFR	31564
1237	RRHB	RDRF	FRKJ	RTHB	RVRF	YRNU	32029	1302	KRRR	CJUR	YNJJ	VCRR	VHVR	RFBR	31899
1238	YVKJ	RTHD	NKKJ	UHHD	NCKJ	RRHB	30518	1303	MDFR	RRRR	RRRR	RRRR	RRYR	TIYN	33204
1239	RDRF	FRKJ	RTHB	RVRF	YRNU	YVKB	32150	1304	KRRR	CTVB	JTVC	KDVC	THFJ	RTHD	30966
1240	RJRF	UHNJ	IFHD	VCKB	RKRF	NJRR	30988	1305	VCKD	VVFJ	RRHD	VVKD	VBTB	FJRT	31488
1241	HDVV	KRRR	CTVC	VJMM	MTR	KDVC	31611	1306	HDVB	KDVN	FJRR	HDVN	VDBR	BRBN	30790
1242	THFJ	RTHD	VCKD	VVFJ	RRHD	VVIV	31735	1307	KDVB	VDVM	BRBH	YRYY	YNFR	IHHB	31435
1243	IKYC	KDVC	HDNK	KDVV	HDNC	KYRR	31265	1308	RKBI	KJRM	HBTG	BRKJ	KYHB	THBR	30750
1244	YRCM	YBNH	NRRB	BRMH	KJRR	HBRD	30689	1309	KJHY	HBR	RYKJ	YNHB	RTRY	FHIR	31851
1245	RFFR	KYRR	KBRY	RFBR	TVHK	IHYR	31873	1310	IHKJ	RTHB	RBRF	HBRK	BIKJ	RRHB	30382
1246	VJYV	KJRR	HBRV	RFKD	NKHD	VCKD	30544	1311	TGBR	KJHK	HBTH	BRKJ	FHFB	RRRY	31089
1247	NCHD	VVYR	UMYU	FHKK	NHNR	THBR	31578	1312	KJYN	HBRT	RYFH	IRKJ	FHFB	RRRY	31543
1248	BMFR	KYRR	HKIH	YRKB	YVKJ	RRHB	31482	1313	KJYN	HBRT	RYKY	RRKJ	RRJB	IGYT	32029
1249	RVRF	KDNK	HDVC	KDNC	HDVV	YRUM	31542	1314	NHNR	YHBR	MHKN	RDRF	KBT	RYBR	31638
1250	YUFH	KKNH	NRTH	BRNI	FRFD	RRKB	30713	1315	RBKB	RCRF	IJKK	HBRC	RFKJ	TIHB	30252
1251	RTRF	VJRB	MRTY	KJRT	HBRV	RFKJ	31592	1316	TVRY	KBRC	RFJB	IGYT	KJRR	HDIB	30669
1252	RRHB	RIRF	YRNU	YVKJ	RRHB	RDRF	31536	1317	IVFY	NIMN	JCYR	YRYR	YRYR	YRYR	33812
1253	FRKJ	RTHB	RHRF	IVFU	YIKJ	JCJT	31192	1318	IDII	IJDY	IMDY	YBIH	IDIV	DRJC	30449
1254	NHYR	NUYV	YRKD	YBKJ	RRHB	RDRF	31272	1319	JCDY	IDDI	DDDY	INUK	INFD	GGYR	30869
1255	FRKJ	RTHB	RVRF	KBRD	RFBR	KTFR	31216	1320	IVFJ	FNFD	JCDI	ITIY	UKYR	YRYR	32582
1256	KDNH	THFB	RDRF	HDVB	KDNJ	FJRR	30558	1321	DIFT	FYGD	FVFT	GIFM	GYJC	IYIT	31232
1257	HDVN	KDVB	UHNJ	RTHD	VCKD	VNNJ	31129	1322	IUIC	DUUK	YRIY	FTFU	FCYR	DUGR	31779
1258	RRHD	VVYR	YIYV	KJRR	HBTR	RTVN	32602	1323	YNJC	IUIM	INDI	DYIM	IVYN	YNYN	32309
1259	RDRF	VNRI	RFFR	KDNH	THFJ	YHHD	30686	1324	JCYB	IIID	IVUK	YRII	FDFV	FDGI	30518
1260	VMKD	NJFJ	RRHD	BRYR	UGYN	FRFD	31225	1325	FDYR	IUFH	FTGY	JCYB	IUIV	DYUK	31765
1261	RRKJ	RTHB	RVRF	KJYR	HBRR	RFYR	31986	1326	YRIU	FVFD	FTGY	YRDI	FDGH	GIJC	30267
1262	GNYI	YRGN	YIFR	KJRT	HBRV	RFKB	31219	1327	YBUC	UKYR	YRYR	IGFM	GIFM	YRID	31421
1263	RIRF	VJRY	CRRT	FRKB	RIRF	UHNB	31135	1328	IMIV	JCYB	IHK	YRYR	YRIH	FMFB	31390
1264	RDRF	URMF	VJRY	JRMY	KDNH	THFB	31134	1329	FDJC	YBIN	UKYR	YRYR	IGFM	GIFM	31388
1265	RDRF	HDVC	KDNJ	FJRR	HDVV	KDVC	30898	1330	YRDI	FDGH	GIFD	FNFI	JCYB	DDUK	30021
1266	THFJ	RTHD	VBKD	VVFJ	RRHD	VNYR	31863	1331	YRYR	YRDR	FTFG	FDYR	GDGR	JCYB	30814
1267	YIYV	VNRI	RFKJ	RRHB	TRRT	FRKB	31415	1332	IIUK	YRYR	YRDR	FTFG	FDYR	FIFM	31181
1268	RIRF	UHNJ	RTHB	RDRF	FRKB	RDRF	30647	1333	GGFN	JCYB	DHUK	YRYR	YRII	FDFV	31646
1269	BRRT	FRVN	RDRF	FRKB	RIRF	UHNB	30893	1334	FDGI	FDYR	FVFJ	FNFD	JCYB	DFUK	30409
1270	RDRF	VJRY	CRRT	FRNN	RDRF	FRKB	31100	1335	YRYR	YRIK	FMFJ	FNYR	FVFB	FNFD	30802
1271	RTRF	VJRN	CRRJ	YRNU	YVKJ	RRHB	32037	1336	GUJC	YBIY	UKYR	YRYR	IYFV	FMFU	32455
1272	RDRF	FRYR	NUYV	YRKD	YBKJ	RRHB	31657	1337	FCYR	GUGI	FTGY	GIJC	YBID	UKYR	31448
1273	RDRF	FRKB	RYRF	MRRJ	YRNU	YVKJ	32518	1338	YRYR	IYFV	FMFU	FCYR	FDFN	FIJC	30448
1274	RRHB	RDRF	FRYR	NUYV	YRCM	YBKJ	32046	1339	YBIU	UKYR	YRYR	IUFM	GRGJ	YRIY	32465
1275	RRHB	RDRF	FRKB	RVRF	BRRT	FRKB	31028	1340	FVFM	FUF	JCYB	IFUK	YRYR	YRIF	31980
1276	RIRF	VBRG	RFBR	RUIV	UDYB	CRDJ	31074	1341	FJFN	FIYR	YRYR	YRYR	YBDY	33377	
1277	KBRG	RFUH	NBRI	RFHB	RFRF	KDNF	30342	1342	UKYR	YRDY	FDGR	FDFT	GIYR	FFFJ	30742
1278	HDVC	KDNG	HDVV	KDVC	THFB	RFRF	30598	1343	FNFI	JCYB	IIVK	YRYR	YRIV	FMFT	32369
1279	HDVB	KDVV	FJRR	HDVN	KBRJ	RFHD	30706	1344	FIJC	YBIB	UKYR	YRYR	IBFD	GYFG	31237
1280	VMKB	RKRF	HDBR	YRUG	YNKB	RJRF	31373	1345	FDJC	YBDU	UKYR	YRYR	DUFT	GFFD	31456
1281	UHNB	RFRF	HBRJ	RFKB	RKRF	NJRR	31106	1346	JCYB	IJUK	YRYR	YRII	FJGY	FDFU	31531
1282	HBRK	RFKB	RIRF	BRRT	FRKR	RRCT	31579	1347	GIFM	GYGJ	JCM	KDMR	RKRK	RKRK	31448
1283	NHVJ	FRCR	RUTH	FJYR	JTNF	VHVV	32245	1348	KKFR	YRII	URKD	MTJB	IKRU	KDMY	31584
1284	RIRF	BRNM	FRKB	RKRF	VJCR	BRRH	31028	1349	JBIC	RUKJ	RUJB	IYRU	KDMG	JBII	30788
1285	KBRJ	RFVJ	HHJR	RTFR	KBRI	RFUH	31249	1350	RUKD	MHJB	IDRU	YRDF	NIFR	YRII	31409
1286	NBRG	RFHB	RFRF	KBRJ	RFHD	VCKB	30179	1351	URKJ	RGJB	IYRU	KDMU	JBIH	RUKD	31092
1287	RKRF	HDVV	KDVC	UHNB	RFRF	HDVB	30612	1352	MIJB	IJRU	KDMD	JBII	RUKD	MFJB	30266
1288	KDVV	NJRR	HDVN	KDNF	UHNJ	RTHD	31116	1353	IDRU	YRDF	NIFR	YRII	URKJ	RCJB	31158
1289	VMKD	GNJ	RRHD	BRYR	NUYB	KBRJ	31378	1354	IYRU	KDMU	JBIH	RUKD	MIJB	IJRU	31026
1290	RFTH	FBRF	RFHB	RJRF	KBRK	RFFJ	30439	1355	KDMD	JBII	RUKD	MFJB	IDRU	YRDF	30890
1291	RRHB	RKRF	IVUD	YBKR	RRCT	NKVJ	31756	1356	NIFR	YRII	URKJ	RVJB	IYRU	YRDF	32036
1292	JCMR	RIVH	IVKG	YBVH	JHTH	FDNK	30877	1357	NIFR	FHHD	MNFH	HDMM	KRRT	KYRT	31855
1293	HDNK	KDNC	FJRR	HDNC	FRKD	NKHD	30117	1358	CTMN	JDNM	NHVV	VRRI	BRMF	KDMN	31099
1294	VCKD	NCHD	VVVF	VVKR	MNCT	VCVJ	31847	1359	THFJ	RIHD	MGKD	MMFJ	RRHD	MHKR	30755

1360 RRCT MNVJ YRMR RIVH BRMG RRJH 31657  
 1361 THFD MNHD MNKD MMFJ RRIH KDMN 30721  
 1362 IHIV IVUR FHHF MNFH HDMM KRRT 31231  
 1363 KYRT CTMN JDHY NHVH VRRD BRMF 31415  
 1364 KDMN THFJ RIHD MNKD MMFJ RRIH 30793  
 1365 KDMN IHIV FVUR RRMM MMNR RYNT 32657  
 1366 RYRR YR 3468 \*

**AUTORUN.SYS**

**AMD**  
S.65

1000 MMMM RRTV RRTM RRRR RRRR RRRR 32878  
 1001 RRRR RRRR RRRR RRRR RRRR RRRR 32921  
 1002 RRRR RRRR RRRR RRRR RRRR RRRR 32922  
 1003 RRRR RRRR RRRR RRRR RRRR RRRR 32923  
 1004 RRRR RRRR RRRR RRRR RRRR RRRR 32924  
 1005 RRRR RRRR RRRR RRRR RRRR RRRR 32925  
 1006 RRRR RRRR RRRR RRRR RRRR RRRR 32926  
 1007 RRRR RRRR RRRR RRRR RRRR RRRR 32927  
 1008 RRRR RRRR RRRR RRRR RRRR RRRR 32928  
 1009 RRRR RRRR RRRR RRRR RRRR IVGF 32147  
 1010 TVIV MYTV YRJI TNRT RIRR IIUK 32055  
 1011 IUII IBDR IJIV IDDY YNIM IYIK 31078  
 1012 YRYR BRTN RFYY MKTN YRHJ TNYR 32515  
 1013 GVTB YRJI TNRT RIRR IIUK IDII 31007  
 1014 IJDI IMDY YNIM IYIK YRYR BRTN 32047  
 1015 RFRR RRYR YRBR TNMM TGRR YRYR 32793  
 1016 HJTN KJFR HBHR UTYR JITN RTRI 31859  
 1017 RRII UKIT DUDU YNIM IYIK YRUR 32174  
 1018 RGYR BRTN RRRF GKUT YRHJ TNKJ 31691  
 1019 MMHD RHKJ RTHB MMRF HDRJ KJRR 30809  
 1020 HBII RYKJ IVHB FTBD KJGU HBFY 30634  
 1021 TDKJ TVHB FUTD KRRR HVYM RYHV 32244  
 1022 VFYR KJNG HBUR RYKJ TBHB UTRY 31710  
 1023 KJRM HBVD RYJH JJRR RFVH VRKU 31793  
 1024 BRMH KRHR KJRR JJRR RRVH BRMK 31407  
 1025 KJIR HBRN BIKJ RGKY NIKR FYYR 31650  
 1026 DVNI KJYY HBYM RYYR MBMY VJIU 32196  
 1027 MRTB VJID MRRG VJTT BRMT IVGG 31291  
 1028 NIKB MMRF VJRT MRRU YRGV TBKJ 31803  
 1029 RTHB MMRF IVRR YRKJ FKHB FYTD 30976  
 1030 KJTB HBFU TDIV GINI KBMM RFVJ 30671  
 1031 RYMR RUYR GVTB KJRY HBMM RFIV 31384  
 1032 RRTM KJRR KHKK HDVC HDVB HDVM 30362  
 1033 HDBT KJYR HDVV KJBH HDVN KJUH 30811  
 1034 HDBR KJVR HDBY KJRR HBRB BIHB 30201  
 1035 RNBI KJMN HBRT BUGH CTVC HBMN 30335  
 1036 RFCT VBFT VCKB MNRF JTVA NRTR 31470  
 1037 CRRN CTVM HBMN RFCT BTJT VMKB 30859  
 1038 MNRF JTBT VHBR BBNF VVNF VNNF 30969  
 1039 BRNF BYNH NRTH BRBR KJMM HBRT 30820  
 1040 BUKJ YYHB YMRY KJIR HBRN BIDH 30470  
 1041 FRGR GRGR IYMY TBGR RYIT NGTB 31528  
 1042 DYDY DYDY DYDY DYDY DYDY DYDY 31828  
 1043 UTGD FJFU FCRB UUFH FDFV FVRR 30897  
 1044 UFTT RNTT DYDY DYDY DYDY DYDY DYDY 31927  
 1045 DYDY DYDY RHYU RJFM FBGR FJFV 30828  
 1046 FDGY RRRR RHYD RJFI FJGI FMGY 30724  
 1047 RRRR RRRR RRRR RRRR RRRR RRRR 32967  
 1048 RRRR RRRR RRRR RRRR KDMR RKRK 32006  
 1049 RKRK KKFR YRIY TNKD MTJB IKRU 31405  
 1050 KDMY JBIC RUKJ RUJB IYRU KDMG 31035  
 1051 JBII RUKD MHJB IDRJ YRDF NIFR 30733

1052 YRIY TNKJ RGJB IYRU KDMU JBIH 30728  
 1053 RUKD MIJB IJRU KDMU JBIH RUKD 30278  
 1054 MFJB IDRU IVFF TNYR IYTN KJRV 32080  
 1055 JBIY RUIV FFTN FHHD MNFH HDMM 30175  
 1056 KRRT CTMN JJNM RRVH VRRI BRMF 31365  
 1057 KDMN THFJ RIHD MGKD MMFJ RRHD 30223  
 1058 MHKR RRCT MNVJ YRMR RIVH BRMG 31468  
 1059 RRJH THFD MNHD MNKD MMFJ RRIH 30441  
 1060 KDMN IHIV IKTN FHHF MNFH HDMM 30101  
 1061 KRRT CTMN JJMY RRVH VRRD BRMF 31405  
 1062 KDMN THFJ RIHD MNKD MMFJ RRIH 30491  
 1063 KDMN IHIV FKTN RRRR RRRR RRRR 32511  
 1064 RRRR RRRR RRRR RRMM MMNR RYNT 32627  
 1065 RYRR TV 3165 \*