

The easiest way to send  
a professional looking invoice

# INVOICE09

quantity	description	unit price	amount
1		35.00	35.00
1		5.00	5.00

by Chris Dekker

SUBTOTAL	40.00
shipping	5.00
TOTAL	45.00
paid	45.00
balance	.00

includes mailing list software

requires OS9 level 2, 512K CoCo 3  
printer that supports IBM character set

## CONTENTS

INSTALLATION.....	2
general notes.....	2
Floppy disk users.....	2
Harddisk users.....	3
PACKAGE OUTLINES.....	4
USER INTERFACE.....	4
PROGRAM DESCRIPTIONS.....	6
invoice.....	6
sales.....	6
saleslist.....	7
invsetup.....	8
setnumber.....	9
ml.....	9
envwriter.....	10
labelwriter.....	10
TECHNICAL REFERENCE.....	12

## INSTALLATION

### General notes.

To deal with the variety in hardware two installation routines have been included in the package. The user MUST use only the routine relevant to his or her setup.

For all practical purposes systems have been divided into two groups: a group that boots up from floppy disk and a group that boots up from their hard disk. To keep confusion levels low you should only follow the instructions for your particular setup and not worry about other avenues.

Remember that the floppy disk installation routine is trying to build an environment that is suitable for the programs. Spending some extra time to get it just right may pay off in less trouble later on.

If any problems show up with the installation routines, you can always try to put the system together yourself as there are many ways to achieve a desired result. Precise requirements for the system are given in the technical reference section of the manual. Once you are finished with the installation, you should run the program called **invsetup**. Secondly you should read the manual carefully. Some programs are intertwined to the degree that one program may use data files set up by another. Running the former first will be very frustrating.

Before you get into installing the package you should make a backup copy of your programs disk and work with that. It is also advisable to put a write protect tab on this disk. This will prevent damage in case of a mistaken disk swap. NOTE: to distinguish between text and commands; all commands that must be typed will be preceded by a double asterisk (\*\*) in the installation sections of this manual.

### Floppy disk users.

The installation routines expect you to boot up your computer from the OS9 "system master" disk, since every user should have one available. If you start your computer from another setup the routines may work but will probably generate (false) error messages. Once you have booted up and the OS9: prompt appears:

put your Basic09/Config disk in drive d0 and type:

```
** chx /d0/cmds
```

```
** load runb gfx2 syscall inkey
```

Now put your programs disk (supplied with the package) in drive d0. Type:

```
** load /d0/install
```

```
** install #16k
```

Install should take care of the entire process for you. Just

answer the prompts and swap disks when necessary. If install aborts before copying the programs to your new system disk, you can force the program back on track by typing: **\*\* copier #16k**

If you can not get the routines to run successfully; you can look at the technical reference section of this manual to see how far you got and what's left to do. NOTE: install will format your new system disk for you, so using an unformatted disk is just fine.

### Harddisk users.

The program provided for you will only copy the programs from your programs disk (supplied with the package) to your hard drive. It also creates a data directory on the disk, but does not deal with the bootfile. In order for the package to work properly, the only requirements for your bootfile are that it will boot into the windows environment, since the programs can not run from the green 32 column screen of the VDGint module.

The second requirement concerns drive access. Since the programs are coded to access the disk through the dd descriptor, it MUST point to your harddrive.

If your system does not meet these requirements and you do not know how to fix them, you can still install the package on a floppy disk and run it from there. This will have some effects on the programs' performance, but since their disk access is fairly limited this should not be a big concern.

Now to install your package: boot up your computer, put your programs disk in drive d0 and type:

```
** load /d0/installHD
```

```
** installHD #16k
```

If there are no error messages you should find the programs copied to your cmds directory. In your harddrive's root directory there should be a new entry called INVOICEDATA, a data directory and a file called startINVOICE. This is the startup file that makes sure the programs have their utility modules available to them. You should always run this file before running any of the package's programs.

The software also assumes that your harddisk's startup file takes care of things like setting up monitor type, system time, etc. If this is not the case you can add the necessary commands to the startINVOICE file.

If you can not finish the installation for some reason, you can look in the technical reference section of the manual to see how far you got and what remains to be done.

## PACKAGE OUTLINES

The invoice09 package consists of the following programs: INVOICE which you use to enter the data to be printed on an invoice. This program also prints the required form. You use the program INVSETUP to tell *invoice* which subroutines it must use and other relevant data that it is printed on the invoice, like your name, address, etc.

Two database managers ML and SALES are also included. The program ML is used to maintain a database holding the names and addresses of your clients. New records are automatically added to this database by *invoice*, but if you want to make any changes in it you must use this program.

SALES has the same function as *ml* but it allows you to make changes to the file that holds a record of all sales. SALESLIST allows you take a look at the contents of this file. You can have the program's output send to a printer if necessary. Various options allow you to print a complete list, or have data grouped per client or product.

ENVWRITE and LABELWRITER are two utilities included in the package. ENVWRITE allows you to print an address on an envelope. The program can take it's data from a database or as input from the keyboard. It can also print a return address. LABELWRITER allows you to print labels. The program is geared toward the use of standard 1"x3.5" mailing labels and can print one to four lines of text on the label.

It is ESSENTIAL that you keep all these programs together in a directory called CMDS. If you move one or more of the programs, chances are that they can not locate the data modules they need and will refuse to run.

## USER INTERFACE

The programs *invoice* and *saleslist* are prompt driven. Every time you must input data or make a choice you will be prompted on the screen for what you must enter, which key to press, etc.

*Envwrite* and *labelwriter* are also driven from the keyboard. However before these programs do any printing you must first select the correct record. You must use the arrow keys to step through the database record by record. Other active keys like F1 or F2 will be displayed on the screen along with their functions. All you have to do is press a key to execute that particular function.

In order to select a database you will be presented with a small menu as the program starts. To access a certain file you must press a number key corresponding to the number

shown in front of the file name.

The programs *ml* and *sales* allow you to select a datafile in the same manner. However these programs are entirely mouse driven. Before you can use these two programs you must run a program called TERM2. Term2 sets up 3 device windows on a 40 column text screen. The lower half of the screen becomes your working area, while the other two windows display records and provide other information.

To make a selection you must put the cursor in front of an item and click the left mouse button. When you click the right button the program displays the next batch of records (if possible). Any time you click on a position that is considered illegal, your choice will be ignored.

Any keyboard input and messages are handled through dialog boxes and the upper windows. At certain points you can use shortcuts by clicking both mouse buttons simultaneously. The programs will alert you to that with a message in the upper left window. The buttons are usually referred to as F1/F2 analogous to the keys that are part of the keyboard mouse.

If you do not have a mouse you can either use a joystick for that purpose or use OS9's keyboard mouse. The keyboard mouse consists of the four arrow keys plus the F1 and F2 keys. It can be toggled ON and OFF by pressing <CTRL><CLEAR>. It's main drawback is that it disables the backspace function of the left arrow key. This means you have a lot of toggling to do if there are many typing errors.

One more thing that should be mentioned here is the automatic pause feature. This puts all mouse driven programs that do not run in the interactive window to sleep. This prevents the system from becoming sluggish when 2 or more programs run simultaneously. When you return to the window you will notice that the cursor no longer blinks. This is a sign the program is sleeping. Pressing one of the mouse buttons will put you back in business.

The name of the described terminal is *term2*. It is located in the utilities file and will be loaded on startup. *Term2* sets up a terminal using window descriptors w4, w5 and w6.

## PROGRAM DESCRIPTIONS

### INVOICE

*Invoice* is the program that allows you to print invoices. Apart from that it saves the data you enter into several files. One file holds specifics of every invoice: item sold, quantity, price, etc. A second file holds names, addresses and phone numbers of clients.

You can also, using the *invsetup* program, have invoice update the journal of the accounting level 2 package if you have that available. *Invoice* has a subroutine that can print the address on an envelope for you. However to be able to use this effectively your printer should have a paper parking feature.

*Invoice* also has features built-in like calculating taxes, shipping and handling and it can print a message of up to 250 characters long on the invoice. However, to streamline day to day use, a lot of these features are only available if you tell the program to use them. You must do so by answering the various prompts of the *invsetup* program. So it is essential that you run this program before you try to use *invoice* itself.

The form that the program prints should look the same as the invoice included in your order. If it doesn't make sure that your printer is using the IBM character set while printing the form. This is necessary to get the lines and shading printed. It is also possible that your printer needs different commands to achieve the desired result. The default values included in *invoice* are used with a Star NX2420 printer. This printer emulates the IBM proprinter X24E. If your printer emulates a different machine you will probably have to change some commands. This is also done with the *invsetup* program.

### SALES

*Sales* is the database manager for the sales record. Before you can run this program you must run a program called **term2**. This program will be loaded as part of the utilities file. To run it simply type: **term2** and press the <ENTER> key. You will now see 3 windows appear on the screen. On the lower half of the screen you should also see the OS9: prompt. Now you type the name of program (*sales*) and press <ENTER>.

The program tries to find a file called `/dd/invoicedata/sales` and reads it's contents. If the file holds more than 100 entries

it will load only the last 100 records. Using this program you can add, delete or modify a record. Since the program is mouse driven and self prompting this needs no further explanation. Upon exiting the program the data will automatically be saved back to the disk file.

### SALESLIST <filename> <device>

*Saleslist* does as it's name implies: it can print a listing of all sales on record. The two parameters you can pass to the program are optional. If you do not type a filename (or pathlist) the program will prompt you for one. Device is necessary if you want the program to send it's output to a place other than the screen you are looking at.

For instance if you want some hardcopy you would type: `saleslist("/p")`. The brackets and quotation marks are necessary because *saleslist* is written in Basic09. If you use Shell+ you don't have to type these special characters. When you type only a filename (at command line or prompt) the program looks for that file in the `/dd/invoicedata` directory. Alternately if you type an entire pathlist, the program will follow that to find the file.

The program has two menus each containing three options. You select an option by pressing the appropriate number key. If you send long listings to the screen, the output will pause every time the screen is full. You must then press a key before the program moves on.

This program can produce three different listings:

- a list of all entries in the selected datafile
- a list of all sales per customer
- a list of all sales per product

NOTE: if it is more convenient for you to have more than one datafile, for instance one for every year or month, etc. you can simply rename the file `/dd/invoicedata/sales` to give it another name. You can still access these files with this program, but no longer with the *sales* program. *Invoice* will simply create a new file named `/dd/invoicedata/sales` if it can no longer find that file.

## INVSETUP

This program allows you to customize your version of *invoice* and also let's you adapt that program to your system. Although it is not very difficult to use this program it may be harder to figure out what numbers you have to enter.

When the program starts it displays a menu showing 10 printer commands as well as the codes necessary to execute those commands on a Star NX2420 printer in IBM emulation mode. This printer emulates the IBM proprinter X24E. So chances are that if your printer uses the same protocol you won't have to worry about the printer codes at all. Otherwise your printer's manual should have a table in it where you can find the codes to make *invoice* work with your printer.

If a command is not supported by your printer, for instance you don't have a color printer, you can set the codes for that command to zero. This will cause the program to skip that command. The printout may look a little different but you can still use the program.

To move the cursor around you must use the arrow keys. To change a code you must set the cursor in front of that code and type the new code number. These characters will not be echoed on the screen but appear as soon as you press the <ENTER> key, which you must do after every code. E.g. to enter the code 27 12 you must type 27<ENTER>12<ENTER>.

If you have to change the codes for line distance (line 10) make sure you keep the codes 27 50 as the 4th and 5th bytes otherwise your printer will most likely ignore the command. The 3rd byte will be filled in by the software as necessary.

When you are finished here you must put the cursor in front of EXIT and press the right arrow key. The program will now ask you a number of questions about what parts of *invoice* you want to use, etc. Whenever relevant the current setting of a variable will be displayed. For some variables you can keep the current setting by simply pressing the <ENTER> key. This message will be displayed as part of the screen prompt.

The program provides you with 5 lines to provide personal information that will be printed in the top portion of the *invoice*. One for your name, one for your telephone number and three lines for your address. Since *invoice* sees this data simply as 5 strings that have to be printed you can enter whatever you want to appear on your *invoice*. The maximum length of each string is 45 characters.

When you have answered all prompts you should see a sign-off message that the data has been verified and saved. Please note that this program alters a data module that is part of the file which holds the *invoice* program. Although

this technique has certain advantages for *invoice* it also means that *invsetup* won't run if it can not find *invoice*. It looks for the file in the /dd/cmds directory.

## SETNUMBER

This program gives you an easy way to manually update the *invoice* number held in *invoice*'s data module should this be necessary. Under normal circumstances *invoice* will increase this number automatically every time it has finished printing an *invoice*.

However if something went wrong or you want to skip to a different starting number, this program will do the job for you without having to go through *invsetup*. The program will prompt you for the number of the next *invoice* you want to print.

NOTE: this program must also have access to the file /dd/cmds/*invoice* to do it's job.

## ML

*ML* is the database manager that takes care of your mailing list. To run this program you must set up a terminal using the program *term2*. This program uses the same interface as *sales*, which is also described in the user interface section of this manual. Over here I will only discuss a few points that are unique to this program.

Concerning the user interface you will see that the main menu is larger since this program also includes printer drivers for printing mailing labels as well as a list of all names, addresses and phone numbers in a datafile. The routine that prints the labels allows you to print 1 to 9 labels for every record in the file or you can select only certain records using the point and click interface. If you accidentally tag a record you can deselect it by setting the number of labels to be printed for that record to zero.

*ML* will first try to read the /dd/invoicedata directory so it can present you with a menu of the (data)files in that directory. If it can not find the directory it will look for a directory called /dd/mailldata. This allows you to use *ml* as a standalone program where you can use it for keeping track of family, friends, etc.

Another thing that I want to mention is that you can also use *ml* for other purposes. For instance I have a database file that doesn't hold names and addresses labels but the text of various disk labels, reminders, warning labels, etc. Since

the program can't adapt it's prompts they look a little out of place but everything else works fine.

For this purpose I use the name, street, town and country fields of a record. The first three are defined as 30 character strings, while the country field is 20 characters long. This gives you 110 characters divided over four lines that you can print on label. This will hold more then enough information for most purposes.

To create a new datafile you must press the 0 key at the file selection menu and then enter a filename. *MI* will then create that file for you. You can have a maximum of 9 files in any given directory. Each file holds 100 records.

### ENVWRITE

*Envwrite* is a utility that lets you print an address straight on to an envelope. If you have only one or two addresses to print it is usually quicker and easier to simply push an envelope into your printer then to print out a couple of mailing labels and stick them on the envelope.

This program allows you to get both the address and the return address in the right spot with a minimum of typing. If you want the return address printed you must make sure this program has access to the file /dd/cmds/invoice (the *invoice* program) because it tries to find the return address in a datamodule in that file.

The program will prompt you for the size of the envelope and will adjust it's output accordingly. There are two ways you can enter the address to be printed. You will be prompted for the one you want to use. If you decide NOT to use the database you can enter the name and address from the keyboard.

If you want to use a record from the database, you will have to select a datafile using the same menu as in *invoice* or *ml*. After loading that file the computer will display the first record in the file. Using the left and right arrow keys you can get to the record you want to print. To actually print it you must press the F1 key. At this point you can print another address if you want or you can exit the program. You can exit a datafile at any time by pressing the F2 key.

Like *ml*, *envwrite* can find datafiles in either the /dd/invoicedata or the /dd/maildata directory.

### LABELWRITER

You can use this program to print labels for many applications but it is not very suitable for printing address labels. If

you want to print address labels use the print routines built into *ml*.

There are two reasons for not using this program for address labels. One is that the routine centers each line on the label. The second, most important, reason is that the routine prints only four fields from a record. Those fields are: name, address, town and country. Since each of these fields is defined as a string you can use them to make all kinds of labels. E.g identifying labels for diskettes, warning labels or just about anything else you can think of. For more information on the size of the strings see the description of *ml*.

At the start of the program you will be presented a menu to choose a datafile. The program then proceeds with displaying record one in that file. The display is structured in such a way that it gives you a good idea of what your label looks like before it gets printed.

Use the left and right arrow keys to get to the record you want to print, then press the F1 key. You will now be asked how many labels you want to have printed. If you enter 0 your selection is canceled and you can make a new selection.

To exit a file press the F2 key. You can exit the program in two ways. You can press the 0 key while in the file selection menu or you can simply press the BREAK key.

*Labelwriter* will search the /dd/invoicedata and /dd/maildata directories for files containing a database.



## TECHNICAL REFERENCE.

### "system disk" requirements.

The bootfile on the disk must include the following modules:

OS9p2	init	ioman	rbf	cc3disk
d0	dd	scf	cc3io	grfint
term_win	w	w1	w2	w3
w4	w5	w6	w7	printer
p	clock	cc3go		

If you want to use more floppy drives you must add their descriptors. If you want to add a ramdisk and/or harddrive you must add the driver/descriptor pairs for these devices. Of course adding even more devices gives no problems, but they are irrelevant to this package. The term\_win descriptor should be defined as a text window (either 40 or 80 columns).

In the root directory two directories must be available. They are: CMDS and INVOICEDATA. CMDS must have a file in it named **shell**. This file should contain the following modules: *shell*, *runb*, *syscall*, *gfx2*, *iniz*, *load*, *display* and *link*. This combination should keep the file's length just within a 16K block; showing as \$3FE4.

If you are using Shell+ or run the package from your harddisk, a slightly different arrangement will also work. In this case you must make sure that *Gfx2* and *syscall* are merged into one file with *Runb*. Then force the computer to load this file with a *load runb* command early in your startup routine. If the computer loads one of the three modules BEFORE loading the Runb file, you will most likely experience problems with programs aborting on an error 43 message.

Also in the CMDS directory must be a copy of the file Grfdrv. This file contains subroutines and has a length of \$1FFC. The following files must be present when the installation routines have been run. In the CMDS directory:

invoice	invsetup	setnumber
sales	saleslist	utilities
ml	envwrite	labelwriter

In the disk's root directory you should find a file called startup (startINVOICE if you used installHD) and a directory called /dd/INVOICEDATA.

The floppy disk installation routine tries to merge the following modules into the **utilities** file: *del*, *dir*, *list*, *echo*, *merge*, *copy*, *date*, *setime*, *unlink*, *free*, *mfree*, *rename*, *attr*, *iniz*, *deiniz*, *montype*. This will give you easy access to those commands.