



Apex II™  
Program Generator

Planning & Generating  
A Program

Part # 02MC564

**COMPSEE, INC.**  
*A Subsidiary of McRae Industries, Inc*

## Apex II Program Generator

---

### Topics Covered in this Booklet

Welcome to the Apex II Program Generator! .....	4
Prefer On-line Help? .....	4
Want More Information About the Apex II?.....	4
Design Your Custom Program .....	4
Planning a Project .....	5
Step 1 Decide what you want the project to do.....	5
Step 2 Decide how you want the application to do it. ....	5
Step 3 Decide how many applications you need to accomplish these tasks. ....	5
Step 4 Decide what you want the screens to look like.....	6
Step 5 Decide how many input fields are needed and plan the record layout.....	7
Step 6 Decide how to collect the data.....	8
Step 7 Design the input fields. ....	9
Step 8 Define the bar code scanning abilities of each input field. ....	10
Step 9 Design the input masks.....	11
Step 10 Design the lookup files. ....	12
Step 11 Design the look and function of the application sub menu. ....	13
Step 12 Design the application looping pattern. ....	14
Step 13 Design the application branching pattern.....	15
Step 14 Design the math registers.....	15
Step 15 Decide the global operating options.....	16
Step 16 Decide whether you want to make a distribution diskette.....	17
Generating a Program .....	18
Step 1 Install and launch the Program Generator. ....	18
Step 2 Explore and edit an Apex II program.....	19
Step 3 Create, name and save a new program. ....	20
Step 4 Name the first application. ....	21
Step 5 Define input prompts and fields. ....	22
Step 6 Set the options. ....	31
Step 7 Set up the data file and sub menu.....	34
Step 8 Check the order of data collection in your program. ....	35
Step 9 Generate and download the application code. ....	36
Step 10 Run the program on the Apex II. ....	37
Step 11 Retrieve data from the Apex II. ....	40
Step 12 Set up lookup files, definitions and displays. ....	41
Step 13 Set up loop, math registers, and branching .....	48
Step 14 Define record layout.....	51
Step 15 Download, run and retrieve data from the completed program.....	52
Step 16 Document and check your program. ....	52
Step 17 Create a diskette.....	53
Empty Forms for Planning a Project.....	57
Step 1 Decide what you want the project to do.....	57
Step 2 Decide how you want the application to do it. ....	57
Step 3 Decide how many applications you need to accomplish these tasks. ....	57
Step 4 Decide what you want the screens to look like.....	58

Step 5 Decide how many input fields are needed and the program layout. .... 59

Step 6 Decide how to collect the data. .... 59

Step 7 Design the input fields. .... 61

Step 8 Define the bar code scanning abilities of each input field. .... 62

Step 9 Design the input masks. .... 64

Step 10 Design the lookup files. .... 64

Step 11 Design the look and function of the application sub menu. .... 65

Step 12 Design the application looping pattern. .... 65

Step 13 Design the application branching pattern. .... 66

Step 14 Design the math registers. .... 66

Step 15 Decide the global operating options. .... 67

Step 16 Decide whether you want to make a distribution diskette. .... 68

## **Apex II Program Generator**

---

Welcome to the Apex II Program Generator!

This booklet is written for Version 3.0, copyright 1995-97 of the Apex II Program Generator by Compsee, Inc. If you have an earlier version of the software, please call us at (1-800-628-3888) to order updated software.

### **Prefer On-line Help?**

---

---

Once you are in the Program Generator, press [F1] from any window. Help screens give detailed reference information and facilitate rapid learning. They may be printed individually, or the entire help file (Apexhelp.txt) can be printed.

### **Want More Information About the Apex II?**

---

---

The Apex Manual describes the hardware and software procedures for the Apex II unit, including how to use the installed software that is shipped with the Apex II unit. The software delivered with the Apex II can be modified using the Program Generator.

### **Design Your Custom Program**

---

---

We will show you how to produce a simple program for the Apex using the Program Generator. When it's downloaded, the program can be used to track employee ID, aisle, bin, Stock Keeping Unit (SKU) number, and the quantity of items counted.

## **Planning a Project**

---

---

Once you know what you want your program to look like and do, the rest is easy.

### **Step 1 Decide what you want the project to do.**

Analyze which of your inventory goals can be accomplished through well designed Apex software.

We needed a project that would:

- a) Expedite inventory through scanning and uploading to a PC.
- b) Significantly reduce data entry errors and operator errors.
- c) Keep track of which employees inventory which items.

### **Step 2 Decide how you want the application to do it.**

Although this decision may need to be changed later in the planning process, make a tentative decision now.

- a) First, before anything else, we want people to enter a unique employee ID number so that we can track who did what.
- b) Next, we want the handheld unit to display the employee's name, title, and ID number so that people can check to make sure they enter the correct number.
- c) Then we want the person to enter the inventory location, the Stock Keeping Unit (SKU), and the quantity for each inventoried item.
- d) We want the application to reject incorrect data.
- e) We want the operator to be able to enter data quickly and not have to perform rework on inventory tasks.
- f) We want to scan Code 3 of 9 bar codes for the employee number and Code 128 bar codes for the Aisle Bin, and SKU number. We do not want to allow people to scan quantities.

### **Step 3 Decide how many applications you need to accomplish these tasks.**

Each Apex program can contain up to three applications.

We decided to accomplish the above tasks with one application. We named it "Inventory."

**Step 4 Decide what you want the screens to look like.**

We've used specific words so you can tell when we're talking about the Program Generator and when we're talking about the Apex II. This booklet and the on-line help windows use the term "Window" for displays within the Program Generator on your PC and "Screen" for the display screen on the Apex II.

With others on the team, we designed the input screens to expedite the tasks listed in Step 2.

**Application 1 (Inventory) Input Screens**

Screen/Line	Handheld Unit Display	
Screen 1	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;">                     Employee No: _ _ _ _                      Aisle No.: _ _ _ _ _                      Bin No.: _ _ _                      &lt;ALT&gt;&lt;Q&gt; to Quit                 </div>	
Line 1		
Line 2		
Line 3		
Line 4		
Screen 2		
Line 1		<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;">                     SKU No.: _ _ _ _ _                      Quantity: _ _ _ _                      &lt;ALT&gt;&lt;Q&gt; to Quit                 </div>
Line 2		
Line 3		
Line 4		

We want to display a screen with employee information after the person enters an ID number.

**Employee Display Screen**

Line	Handheld Unit Display
Line 1	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;">                     Employee No:0001                      is President                        John Doe                 </div>
Line 2	
Line 3	
Line 4	

Once the SKU number is scanned (or keyed), we want the application to look up the SKU number and display the description of the scanned item. This way, people can check the SKU number against what they are seeing.

**SKU Description Display Screen**

Line	Handheld Unit Display
Line 1	<div style="border: 1px solid black; padding: 5px;">                     SKU No. B-010                      Description:                      Apex, Standard                 </div>
Line 2	
Line 3	
Line 4	

For efficient data control, we want the unit to keep track of the number of records and notify the operator when 1000 or more records are stored in the program.

**1000 Records Notification**

Line	Handheld Unit Display
Line 1	<div style="border: 1px solid black; padding: 5px;">                     Handheld Unit                      Now Contains                      1000 Records                      Retrieve Data!                 </div>
Line 2	
Line 3	
Line 4	

**Step 5 Decide how many input fields are needed and plan the record layout.**

The goal is to collect the required data easily and accurately. We studied the screens and concluded that we want to collect the following data for each record.

Employee Number⇒      Aisle Number⇒      Bin Number⇒      SKU Number⇒      Quantity

We know all the Bin Numbers begin with the letter B and a hyphen, and we do not want the person to enter this each time a new Bin Number is entered. We will plan how to enter “B-” automatically later (page 15).

For now, we need to plan for the modified data, (B-123), rather than the entered data (123) to be stored and uploaded to the PC. This is called the Record Layout. We want the records stored in the order they are collected. We want all the records stored as they are entered, except the bin number. Therefore our planned Record Layout is:

Employee Number⇒      Aisle Number⇒      (B-) plus entered Bin Number⇒      SKU Number⇒      Quantity

Also, we do not want people to have to enter their employee number every time they enter the other information. We will show you how to take care of that with a “loop” later (page 48).

**Step 6**

**Decide how to collect the data.**

The following table lists the fields and sample data similar to what we want to collect and transfer to the PC file named "Invent.txt."

**Data Collected in the Invent.txt File**

EMPL	AISLE	BIN	SKU	QUAN	Date
0001	AA001	B-123	A-005	001	960808
0001	AA001	B-124	B-050	100	960808
0001	AA001	B-125	B-015	001	960808
0001	AA001	B-126	A-060	2	960808
0002	BB001	B-57	A-055	03	960808

We used the following table to plan how to collect our data.

**Data Collection Planning**

Operating Parameter	Value Set
Collection File Name	Invent.txt
Do you want to overwrite the contents of your data collection file, or do you want to append new records to it?	Append
What type of PC File do you want for data collection? Fixed Length ASCII file. Fixed Length ASCII File with Carriage Returns Term Records. Comma Quote Delimited File (separate each record with a comma). Custom Quote Delimited ASCII File. (separate each record with a character that you define). Dbase File. Access Database File.	Fixed Length ASCII File with Carriage Returns Terminating Records.  (Each record ends with a carriage return.)
Data Collection - Handheld unit: Do you want to allow people to create duplicate records; that is, with identical data in one of the fields? If you do not want duplicate records, which field do you want to be unique?	Yes  Not applicable for us.
Do you want the date for each record, without having to key it? Do you want to include the century in the date?	Yes No
Do you want the time for each record, without having to key it?	No
Do you want to delete the data from the handheld unit after it is transferred to the PC text File?	Yes No
Do you want power to the Laser turned off between scans?	Yes



**Step 7****Design the input fields.**

We used the following table to plan our five input fields, focusing on fast and accurate data entry.

The Program Generator classifies input characters as letters and spaces (Alpha), numbers (Numeric), and everything else (Special Characters).

**Field Design Planning**

<b>Design Consideration</b>	<b>Our Field Decisions</b>				
	EMPL	AISLE	BIN	SKU	QUAN
What's a short, easy to remember field name?					
Which screen is it on?	1	1	1	2	2
How long is this field including spaces and special characters?	4	5	3	5	3
Do you want this field to accept letters?	No	Yes	No	Yes	No
Do you want this field to accept numbers?	Yes	Yes	Yes	Yes	Yes
Do you want this field to accept Special Characters?	No	No	No	Yes (-)	No
Do you want this field to be required?	Yes	Yes	Yes	Yes	Yes
Do you want this field to display a default value when the screen is first displayed? If so, what?	No	No	No	No	1
Do you want the program to only allow certain first characters? Which ones?	Yes 00	No	No	No	No
Do you want the field to accept only values in a certain range?	No	No	Yes	No	Yes
Minimum Range Value:	N/A	N/A	1	N/A	1
Maximum Range Value:	N/A	N/A	999	N/A	999
What are the smallest and largest number of characters you want people to enter for this field?					
Minimum Field Length:	4	5	1	5	1
Maximum Field Length:	4	5	3	5	3
After people scan or key data, should the cursor automatically advance to the next input field?	Yes	Yes	Yes	Yes	Yes
Do you want people to be able to scan Input?	Yes	Yes	Yes	Yes	No
Do you want people to be able to key Input?	Yes	Yes	Yes	Yes	Yes

**Step 8 Define the bar code scanning abilities of each input field.**

Use the following table to define which fields you want to accept scanning with the various bar code types. We selected Code 3 of 9 for the EMPL field and Code 128 for the AISLE, BIN, and SKU fields. We want the application to prohibit people from scanning the quantity.

If necessary, you can identify multiple bar code types for one field. For example, the SKUs could be marked by multiple vendors with multiple bar code types.

**Bar Code Design Planning**

Bar Code/Options	EMPL	AISLE	BIN	SKU	QUAN
Codabar Minimum length? Codabar pairs?	No	No	No	No	N/A
Code 3 of 9 Full ASCII?	Yes No	No	No	No	N/A
Code 11 Number of check digits? (1 or 2)	No	No	No	No	N/A
Code 93	No	No	No	No	N/A
Code 128	No	Yes	Yes	Yes	N/A
EAN - 8	No	No	No	No	N/A
EAN - 13 Require 2 or 5 Character Supplemental Code?	No	No	No	No	N/A
Identicon 2 of 5 Checksum? Fixed Length? Length in characters?	No	No	No	No	N/A
Industrial 2 of 5 Checksum? Fixed Length? Length in characters?	No	No	No	No	N/A

**Bar Code Design Planning, Continued**

Bar Code/Options	EMPL	AISLE	BIN	SKU	QUAN
Interleaved 2 of 5 Checksum? Fixed Length? Length in characters?	No	No	No	No	N/A
Matrix 2 of 5 Checksum? Fixed Length? Length in characters?	No	No	No	No	N/A
MSI Plessy	No	No	No	No	N/A
UPC - A Require 2 or 5 Character Supplemental Code?	No	No	No	No	N/A
UPC - E	No	No	No	No	N/A
How many seconds do you want the bar code to be displayed on the handheld unit after a person scans it?	We do not want the bar code to be displayed. Zero seconds.				

**Step 9**

**Design the input masks.**

With an input mask, you can define two levels of validity checking. We decided to use an input mask on the Employee field only. Since all of our Employee ID numbers start with two zeros, we decided to reject anything except zeros for the first two digits. If people enter anything other than zeros, the handheld unit beeps and the input is rejected.

In addition, you can define other masks to check data entry before it is accepted. These masks allow only a specified range of values, or only values equal to, greater than, or less than a value. For example, we thought about creating a mask to require that the quantity be equal to or less than 600. You could also require that the second through the fourth digits be “392” if you wanted to. Input masks save time during inventory because mistakes are caught when they occur, rather than later.

**Input Mask Planning Summary**  
**one input mask to require zeros for the first two digits in the employee field**

Step 10

Design the lookup files.

Lookup files are flat ASCII files. The application can use them to take the data keyed or scanned into the Apex II unit, and look up that data in the lookup file. Designing the lookup files means identifying the data that needs to be in them.

The Program Generator uses four lookup types: none (no lookup file); the record must exist in the lookup file to be accepted; lookup and display; and the record must not exist in the lookup file to be accepted. All this minimizes errors.

We decided to have the application display a person’s name and title after the Employee ID number is entered. In addition, we do not want the program to continue unless a real employee number is entered.

Therefore the EMPL field needs a lookup file (EMPL.fle).

Since we want to display information from the file, we will use the “Lookup and Display” type. For now, we just need a list of the information for the lookup file.

<b>List of Employees</b>	
0001	- John Doe, President
0002	- Mary Smith, Engineer
0003	- Frank Sanchez, Engineer
0004	- Carry Mahoney, Purchasing
0005	- Frances Coyle, Production
0006	- Marilyn Hill, Tech. Sup.
0007	- Ron Jones, Sales Mgr.
0008	- Claudia Anderson, Engineer
0009	- Lou Goldman, Office Mgr.
0010	- Lee French, Software M.

<b>List of Aisles</b>	
AA001	CC001
AA002	CC002
AA003	CC003
BB001	DD001
BB002	DD002
BB003	DD003

We decided we wanted the application to reject everything except actual Aisle numbers for the AISLE field.

Therefore the AISLE field also needs a lookup file (AISLE.fle).

We will use the “Must Exist” type.

Since we add new bin numbers quite often as inventory grows, we decided to not use a lookup file for the BIN field.

**(No Data Needed)**

<b>List of SKU Numbers</b>	
A-005	Apex Standard, 128K
B-010	Apex Standard, 512K
B-015	Apex Standard, 1 Meg
A-020	Apex Integ., 128K
B-025	Apex Integ., 518K
B-030	Apex Integ., 1 Meg
A-050	Cab. Apx-PC 9 pin, 6ft
B-050	Cab. Apx-PC 25 pin, 6ft
A-055	Cab. Apx-QR, 6ft
B-055	Cab. Apx-QL, 6ft
A-060	Apex Battery Pack
B-060	AA Alkaline 100 / Case

Although we do add Stock Keeping Unit (SKU) numbers, we want them to be in our system before they are inventoried. The application should reject everything except SKU numbers in our configuration tracking system. Therefore the SKU field needs a lookup file (SKU.fle).

We will use the “Lookup and Display” type.

We want people to be able to key any number from 1 to 999 into the handheld unit. Therefore the QUAN field does not need a lookup file.

The following table summarizes our decisions about the three fields we want to use lookup files with.

**Lookup File Planning**

Parameter	EMPL	AISLE	SKU
Look up type	Lookup and Display	Must Exist	Lookup and Display
Look up file name	EMPL.fle	AISLE. fle	SKU_DESC. fle
Each record contains	Employee ID number, Name and Title	Aisle number	SKU and description
Key lookup field	Employee number	Aisle number	SKU number

**Step 11 Design the look and function of the application sub menu.**

The Compsee Program Generator includes several pre-programmed (“canned”) sub-routines people can use, in addition to the project they create using the main part of the program generator.

The following table lists the six operations people can do with data. A person selects one of these operations from a sub menu displayed on the Apex II. In the Program Generator, we can enable or disable these operations. For each data operation, we decided to enable it and use the default title.

**Sub Menu Operations Planning**

Sub Menu Operations	Default Title for Program	Enable	Title We Will Use
Enter data	Enter Data	ON	Enter Data
Edit data	Edit Data	ON	Edit Data
Verify data	Verify Data	ON	Verify Data
Transmit data via program	Xmit Data	ON	Xmit Data
Delete data	Delete Data	ON	Delete Data
Transmit data via wedge	Xmit Wedge	ON	Xmit Wedge

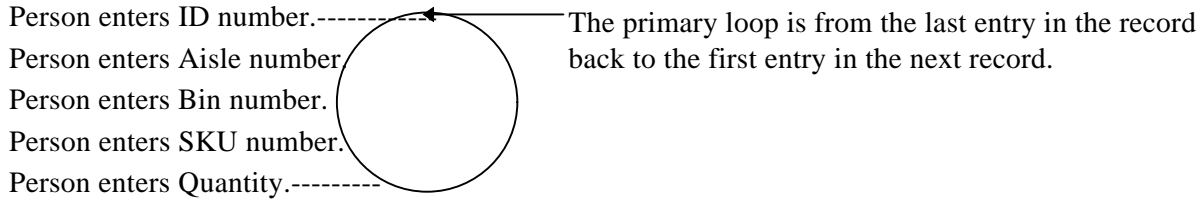
The Enter Data option must be enabled in order for the generated program to work. The Edit and Verify Data options, if enabled, work with our generated program so people can view, check, and edit the data collected with the Enter Data option.

The transmit data via program sub-routine, if enabled, gives people an alternative to the procedure on page 40. The transmit data via wedge sub-routine, if enabled, is for those PCs with a wedge attached. The Delete data program sub-routine, if enabled, deletes the data file in the Apex II.

**Step 12**

**Design the application looping pattern.**

In computer jargon, a Loop is a set of repetitive data collection steps. Each Apex application has one primary loop. That is, you perform a series of steps to create a record.



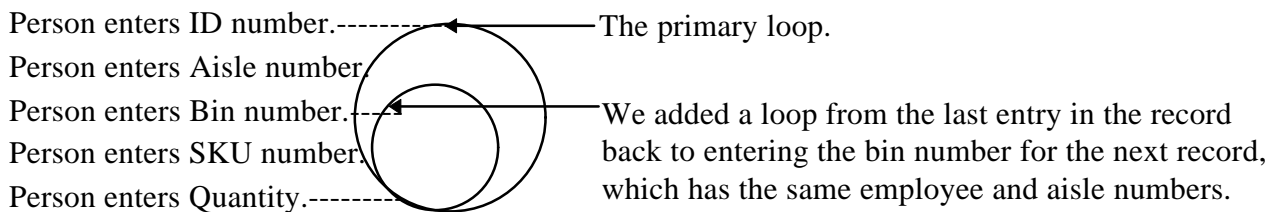
To create another record, you repeat the same steps, and so on. The term “loop” comes from the circular, or looping pattern of steps.

You might want to create additional loops so that people do not have to repeat all the steps each time they create a record.

We decided to add an additional loop, so one employee doing inventory on, say 100 SKUs, does not have to enter the employee number 100 times. To be sure exactly where we wanted to put the loop, we had to list and analyze the steps to create a data record.

Where we work, we may have such a large quantity of an SKU that several bins are required. However, whether we currently have many or few of the SKU, each bin contains only one SKU. When people do inventory, they work down an aisle from bin to bin.

Therefore, we want to place the loop between entering the aisle number and entering the bin number. This is because there are many times when a new record is entered with the same Employee number and Aisle number. We do not want the person to perform unnecessary data entry if we can design the program to do it automatically.



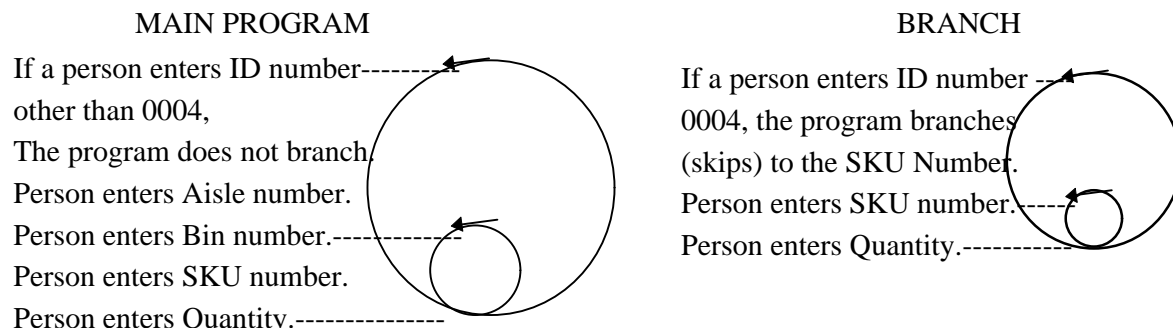
**Loop Planning Summary**  
**one loop from the last field back to the Bin Number**

**Step 13**

**Design the application branching pattern.**

In computer jargon, a branch is a decision about which input field and prompt to display next. Unless a branch is defined, the Apex “decides” to display the next input field and prompt. The Program Generator numbers the fields as you define them. After you define all the fields, you can view and change the order in which they will be displayed for data entry.

We decided not to define a branch since our tasks are relatively straightforward. But suppose that we have a number of vehicles outside the building that need to be inventoried. The aisle and bin numbers would not be used. Further, suppose that the only kind of inventory one employee ever does is vehicle inventory. No other employees perform vehicle inventory. In this situation, we could tell the program to branch to the SKU Number when one particular employee number (for example, 0004) is entered.



**Application 1 (Inventory) Branching Planning Summary  
no branching**

**Step 14**

**Design the math registers.**

You can define registers in the Program Generator. A register is a storage location within an application. Registers are used for accumulating totals of the results of mathematical calculations, and for performing other data manipulations. Registers can contain data from external sources such as files.

We decided to have the program keep track of the number of records collected in the handheld unit and display a warning/direction message when the handheld unit contains one thousand records. We want people to upload data to the computer consistently. This assures that the information on our PC is up to date.

We also decided to have the system automatically add “-“ at the beginning of each BIN record.

Our planning for registers:

**Application 1 (Inventory) Registers Planning Summary**

**one register; named “count”; insert after all record information is gathered and written to the file (“DONE”); start with the value of the register; then add one each time a record entry is completed**

**one register; named “Bin\_Num;”; insert after BIN; start with the value B-; then add this value to the beginning of each BIN record.**

**Step 15 Decide the global operating options.**

Global operating options control Apex II operations. These options are set for the whole project (up to three applications) rather than individually set for each of the applications.

<b>Operating Parameter</b>	<b>Our Decision</b>	<b>Enable</b>
Key Combinations For Basic Operations Keys to quit the operation or program Keys to clear data from the Apex II Keys to increase contrast of the display Keys to decrease contrast of the display	Keep the defaults ALT-Q ALT -. ALT -I ALT -J	N/A N/A N/A N/A
Remap Key Options	Remap the E to a hyphen	N/A
Hardware Options	Keep the defaults	
Disable Scanner Between Scans	0	N/A
Idle time in seconds before the Apex II automatically powers off	90 Seconds	N/A
Data Time Out (Communications Abort Time)	5 Seconds	N/A
Banner Information Banner Line 1 Banner Line 2 Banner Line 3 Banner Line 4	Enable and change the default. Introducing the Apex II Handheld Unit - Welcome! --Compsee Inc.	On



<b>Operating Parameter</b>	<b>Our Decision</b>	<b>Enable</b>
Parameters Options		
Set the date and time on the handheld unit	Yes	On
View the available memory in the Handheld Unit	Yes	On
Parameters Menu Title	Parameters	On
Date/Time Menu Title	Edit Date	
Memory Menu Title	View Memory	
General Options - Perform Operations Automatically Without Being Prompted		
On exiting a program, save changes in project file	Yes	
When generating a program, save changes in the project file	Yes	
Always overwrite the previous version when generating a program	Yes	
When downloading, save the program in the handheld unit's FLASH	Yes	
When downloading, set the handheld in Run Mode	Yes	

**Step 16** Decide whether you want to make a distribution diskette.

This doesn't really take any planning. Since we may want to use the inventory applications in other facilities, we plan to make a diskette.

### Generating a Program

---

---

Generating a program takes much less time than planning it. We spent several days planning, and about an hour generating. As we go through the steps to generate a program, remember how we planned it. This understanding of cause and effect will help you design your first custom program.

#### **Step 1** **Install and launch the Program Generator.**

Your Apex II generator software is delivered on five 3 1/2 by 5 inch diskettes. To install the software on your computer:

1. Insert installation disk 1 into the diskette drive of your PC.
2. Assure that only the necessary windows applications are running.
3. From the Start Button (Windows '95) or from the Program Manager (Windows 3.1), run A:\Setup.

The Setup program prompts you to accept or edit the default installation directory.

When the Setup program finishes, the Apex window is displayed.

4. Double click on the Apex.exe (bar code) icon to launch the Program Generator.



009

#### **NOTE**

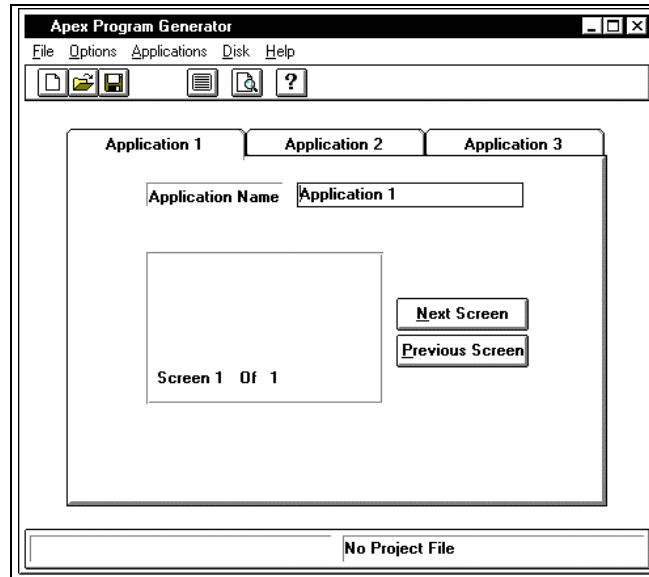
In addition to the five installation disks, you may also receive an update disk. If you receive an update disk, you must use it to assure that you have the latest version of the software.

Copy the ApexUpdt.exe file from the update disk to your installation directory.

Then run the ApexUpdt.exe program.

**Step 2****Explore and edit an Apex II program.**

When the Program Generator is launched, the main menu and application builder are displayed.



080

Before generating your own program, you may want to explore and edit the completed program. The MyInvent.map program is identical to what you will create by following the procedures in this document. MyInvent.map and your finished program need three lookup files to work properly when you download them to the handheld unit. Before exploring, you may want to make copies of the sample map and its three lookup files:

AISLE.file  
 EMPL. fle  
 SKU\_DES. fle

Now launch the Program Generator. Open the MyInvent.map file by selecting File, and then Open. Explore the file. The program displays menus and options on the screen to guide you through the procedures.

To access help information for a window, press <F1> on your keyboard. The help screens give detailed reference information and facilitate rapid learning. To access the complete list of help topics, select the help option from the main menu.

To read or print the entire help file (Apexhelp.txt), open it from any word processor.

After exploring the Program Generator, continue through these steps to generate the Inventory program.

**Step 3** **Create, name and save a new program.**

If you haven't already, we recommend that you make a copy of the following files.

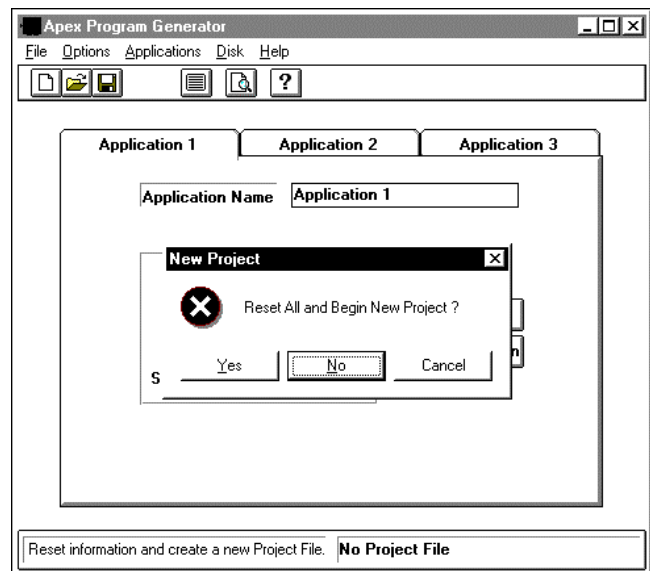
- AISLE.flc
- EMPL. flc
- SKU\_DES. flc

During your learning activities, these files may be changed. If you save copies now, you can refer to them later and compare your work to the finished program.

Select the File main menu option.  
Then select New Project.

You are asked "Reset all and begin new project?"

Answer "Yes."



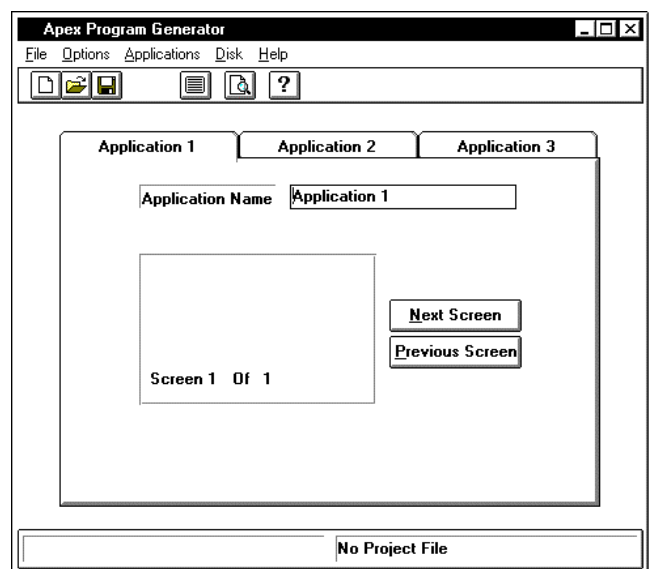
126

The dialog window is removed.

The application Name is "Application 1."

The main menu and application builder is displayed with three application tabs:

- Application 1
- Application 2
- Application 3



080

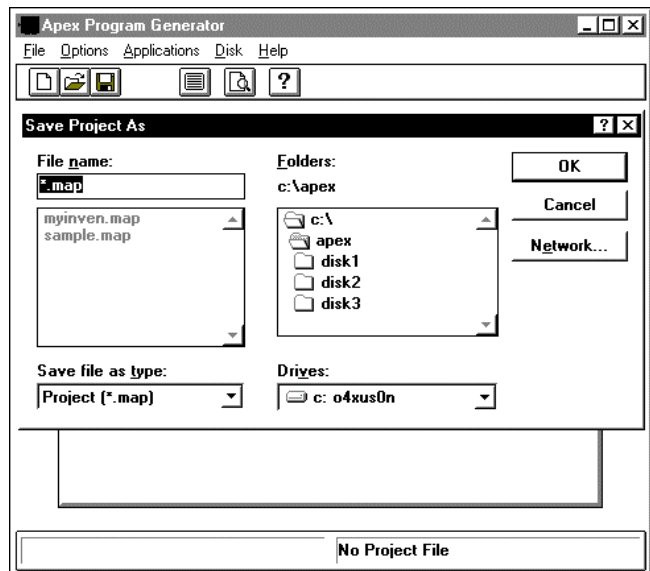
Select the File main menu option.  
Then select Save Project As.

A dialog prompts you to name and save the project. If necessary, navigate to the Apex directory on your PC.

Name the file Inventor.map.

Select OK. The window is removed and the application builder is again displayed.

The Inventor.map is now in the Apex directory.



050

You have now created, named, and saved a new Project.

## Step 4

## Name the first application.

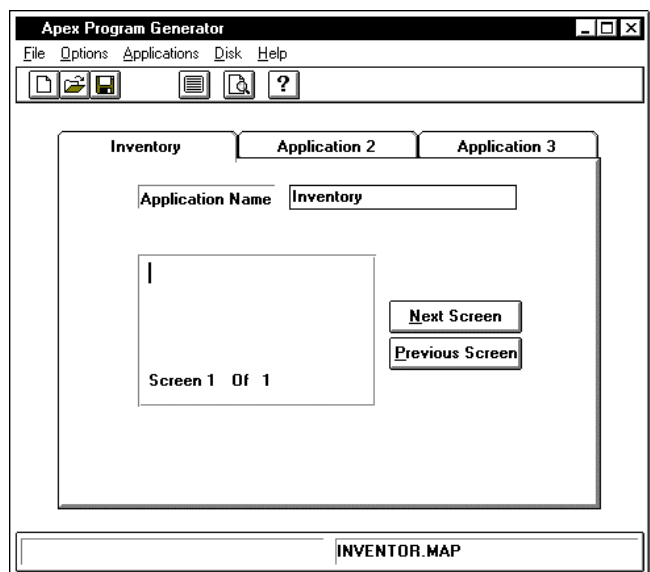
Select the Application Name input field and type "Inventory." Do not type the quotation marks.

The Application 1 tab at the top of the application builder now reads Inventory.

Select the File main menu option. Then select Save Project.

The Program Generator saves the application within the project.

You have now named and saved the first application. We recommend saving the project often.



081

**Step 5**

**Define input prompts and fields.**

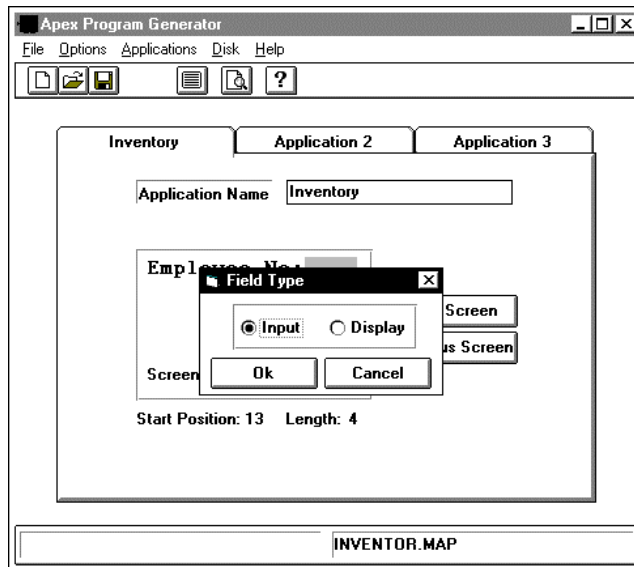
We are now going to implement the planning we did on page 9. The box in the center of the application builder represents what the Apex II will look like when the program is downloaded.

Using the input screen planning on page 6, create the input prompt for the first input field.

Click once in the upper left hand corner of the display box and type "Employee No:." Don't type the quotation marks.

Then click and drag four spaces to the right. As you drag the mouse, the Start Position should display 13 and the Length should be 4.

Release the mouse, and the Field Type window is displayed. Select Input. Then select OK. The Data Definition Window is displayed.

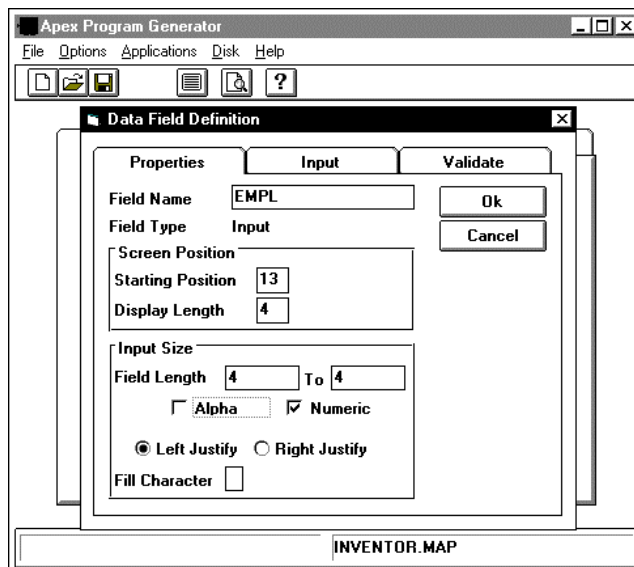


036

This is a little tricky at first. If you do not create a field starting at position 13 with a length of 4, don't worry. Just cancel the data definition window, select the input field, and press the Delete button on your keyboard. You may also type the start position and field length in the Data Field Definition window.

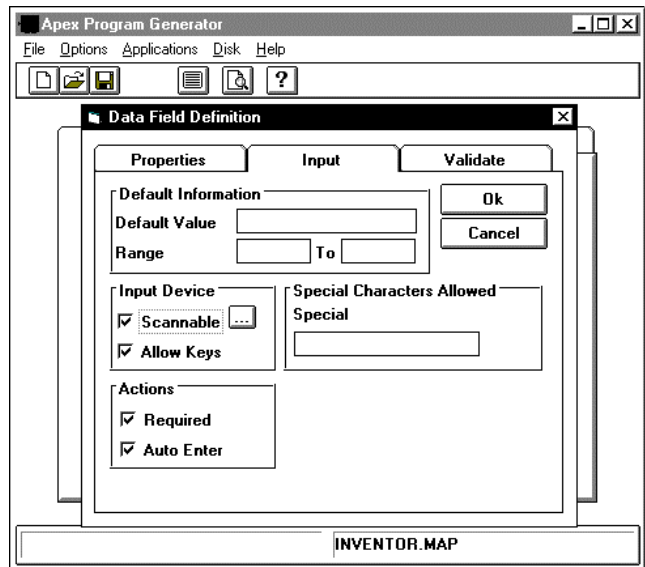
Use the table on page 9 to remind you how you want to define the EMPL field.

- Type EMPL for the Field Name.
- Assure that Starting Position is 13 and Display Length is 4.
- Assure that Field Length is 4 to 4.
- Deselect Alpha and leave Numeric selected.
- Leave the Left Justify button selected.
- Leave the Fill Character box empty.



053

Select the Input tab. Other options are displayed.  
 Leave the Special (characters) fields blank.  
 For Input device, select Scannable and Allow Keys.  
 Leave the Default value and Range fields blank.  
 For Actions, select Required and Auto Enter.  
 When you select Scannable, the “...” button to the right of it becomes active (changes appearance).



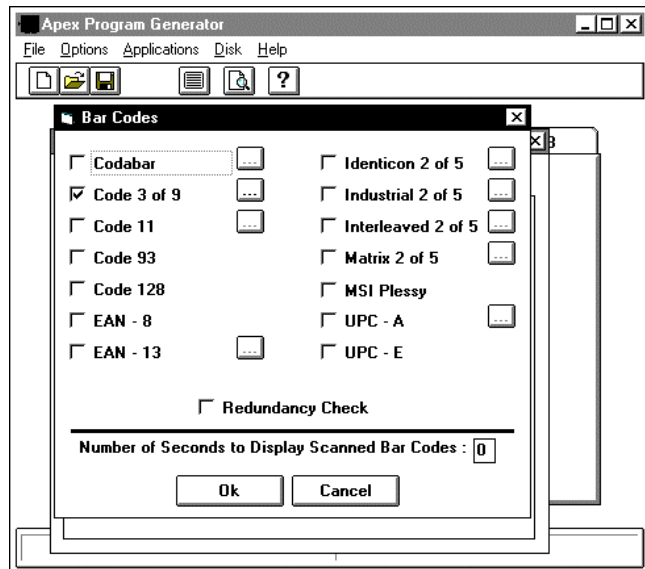
012

Select the “...” button to the right of the Scannable option box.

The Bar Codes window is displayed.

On page 10, we decided to use Code 3 of 9 for the EMPL field, so select Code 3 of 9.

Then select the “...” button to the right of the Code 3 of 9 option.



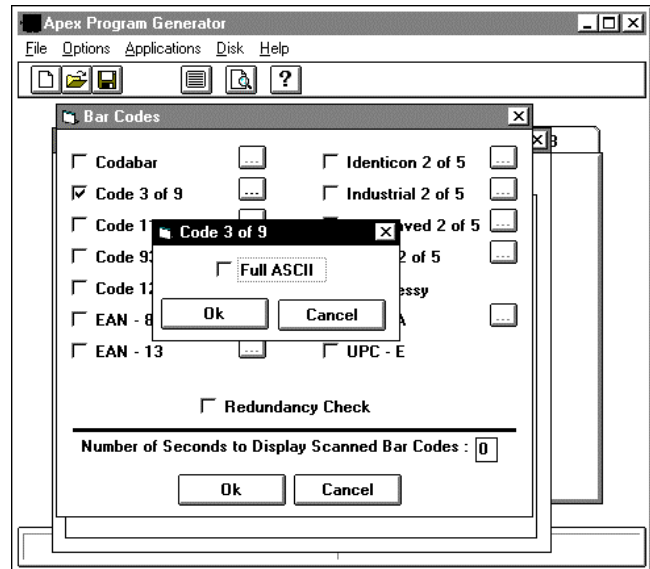
014

The Code 3 of 9 window displays the Full ASCII option box.

Since we do not have any Full ASCII characters in this field, we do not need to select Full ASCII.

Select OK. The Bar Codes window is displayed.

Select OK from the Bar Codes window. The Data Field Definition window is displayed again.



015

Select the Validate tab on the Data Field Definition window.

Deselect Lookup field for now. We will define the lookup files later (page 41).

Select the Mask option box. The Mask Information area below it becomes active (changes appearance).

On page 9, we decided to have the program only allow zeros for the first two characters in the Employee ID number field.

Type 1 and 2 in the From and To columns.

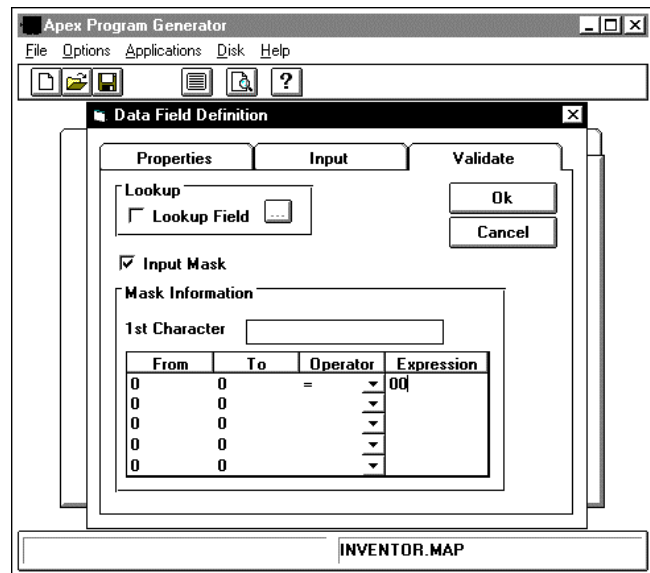
Select the equal sign from the drop down list.

Type two zeros in the Expression column.

Then select OK.

The Field Definition window is displayed.

Select OK from the Data Field Definition window to return to the main menu.



089



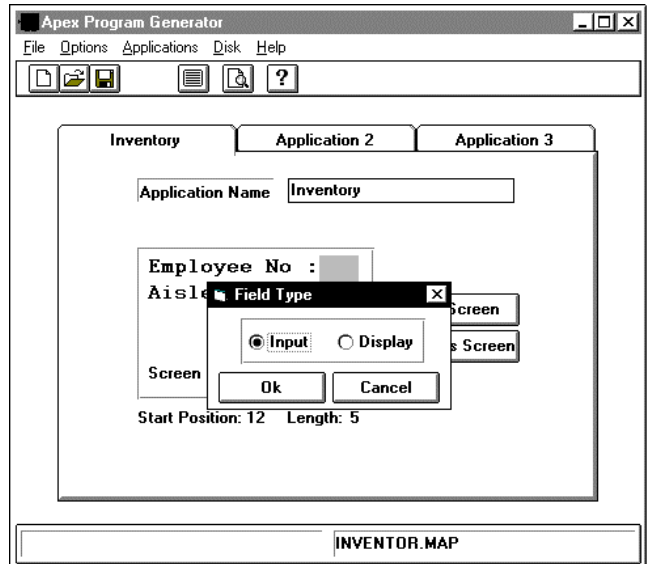
We have defined the Employee input field except for the lookup data, which we will define later (page 41).

Next, we are going to create and define the AISLE input field.

Using the input screen planning on page 6, type the input prompt for the AISLE input field.

Then highlight the input field.

Release the mouse, and the Field Type window is displayed. Select Input. Then select OK. The Data Definition Window is displayed.



082

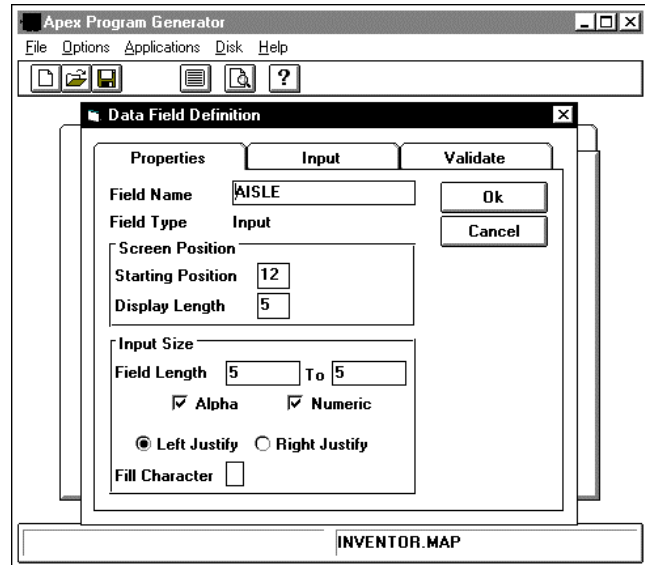
During inventory, the operator scans (or keys) a barcode label at the beginning of the stockroom aisle being inventoried.

Use the table on page 9 and this figure as you define the AISLE field.

Select options on the Properties and Input tabs.

We will define the aisle lookup file from the Validate tab later (page 41).

Select OK to return to the main menu.



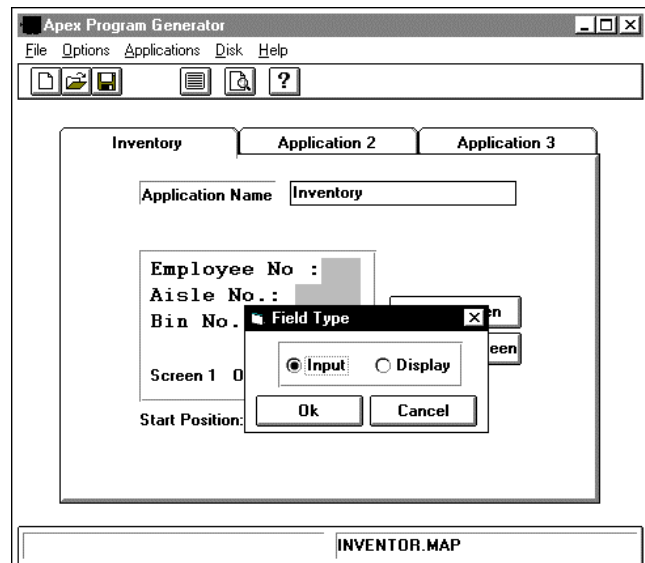
083

Next, we are going to create and define the BIN input field.

Using the input screen planning on page 6, create the input prompt for the BIN input field.

Then highlight the input field.

Release the mouse, and the Field Type window is displayed. Select Input. Then select OK. The Data Definition Window is displayed.



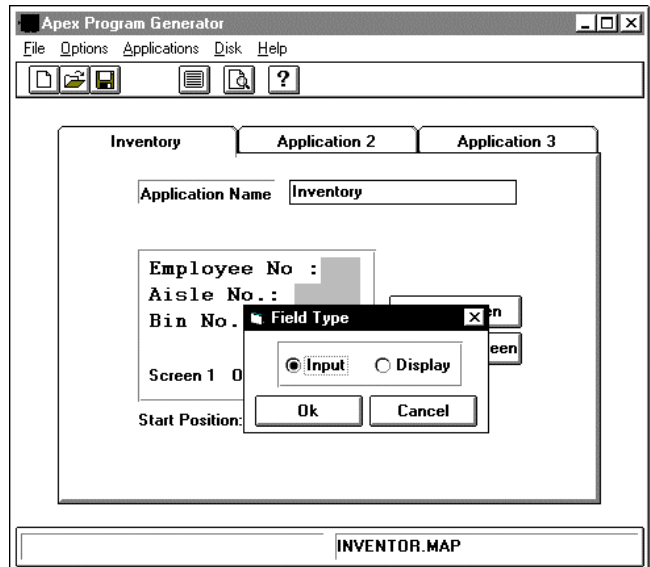
084

During inventory, the operator scans or keys the number of the bin being inventoried.

Use the table on page 9 and this figure as you define the BIN field.

Select options on the Properties and Input tabs.

Select OK to return to the main menu.

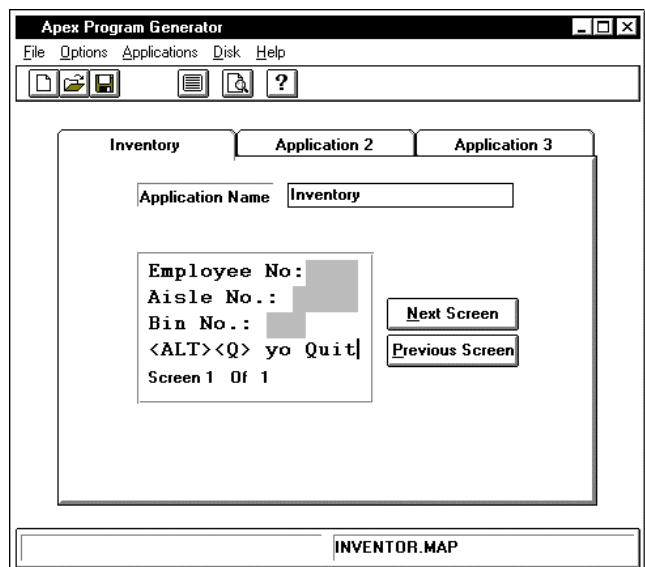


085

The application builder now contains the Employee, Aisle, and Bin prompts and input fields.

We decided to remind people how to quit the application.

Type “<ALT><Q> to Quit” on the bottom line of the display box.

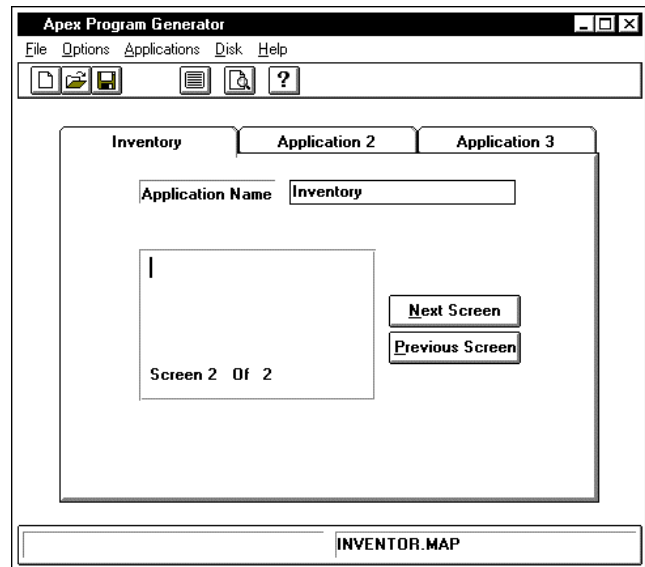


064

Select the Next Screen option button.

The application builder displays the second screen in the first application.

The Next button can also be activated by pressing and holding the <ALT> key and then pressing the <N> key.



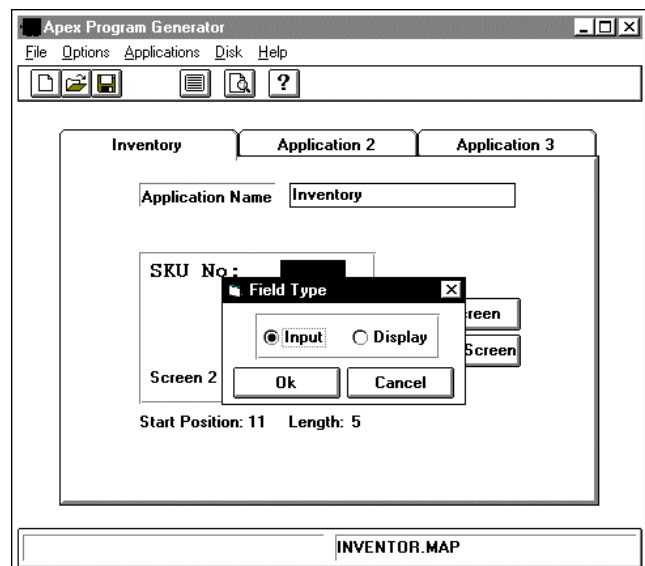
086

Next, we are going to create and define the SKU input field.

Using the input screen planning on page 6, create the input prompt for the SKU input field.

Then highlight the input field.

Release the mouse, and the Field Type window is displayed. Select Input. Then select OK. The Data Definition Window is displayed.



087

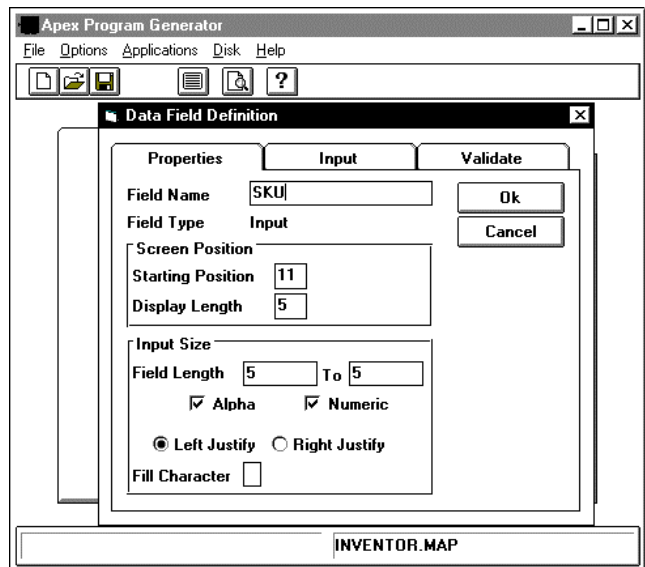
During inventory, the operator scans (or keys) the bar-coded Stock Keeping Unit (SKU) label on the item being inventoried. The operator scans the SKU number on any one of the items in the bin being inventoried.

Use the table on page 9 and this figure as you define the SKU field.

Select options on the Properties and Input tabs.

We will define the SKU lookup file from the Validate tab later (page 41).

Select OK to return to the main menu.



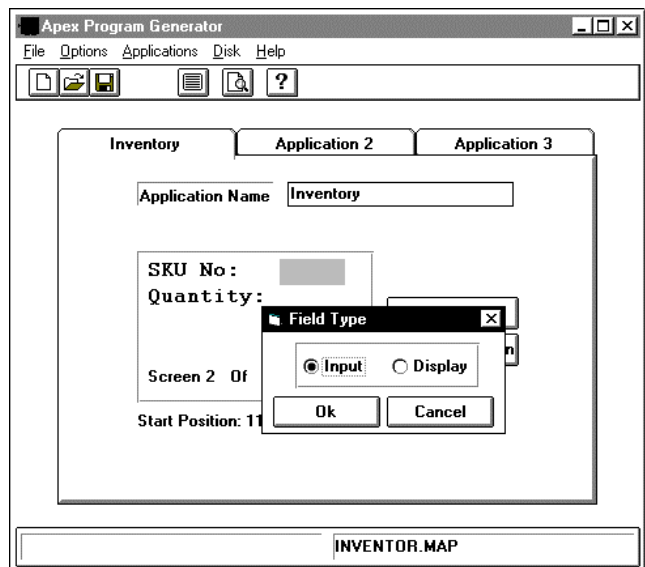
088

Next, we are going to create and define the QUAN input field.

Using the input screen planning on page 6, create the input prompt for the QUAN input field.

Then highlight the input field.

Release the mouse, and the Field Type window is displayed. Select Input. Then select OK. The Data Definition Window is displayed.



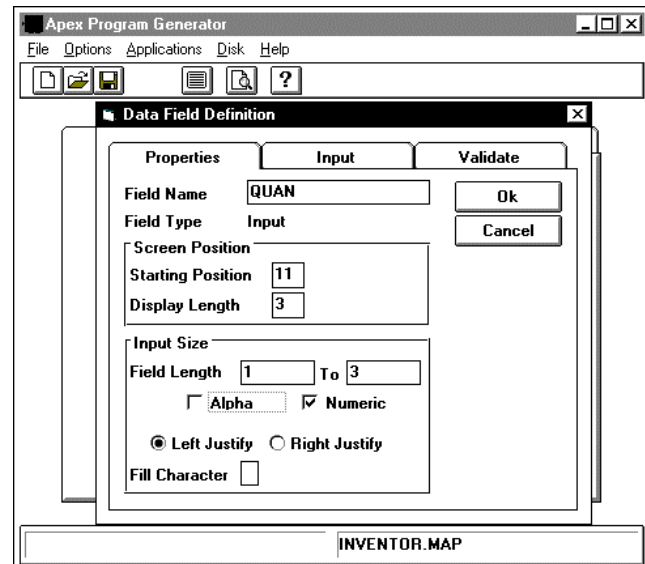
090

During inventory, the operator counts the number of items having the identified SKU number in the identified bin. Then the operator keys the quantity into the Apex II unit.

Use the table on page 9 and this figure as you define the QUAN field.

Select options on the Properties and Input tabs.

Select OK to return to the main menu.



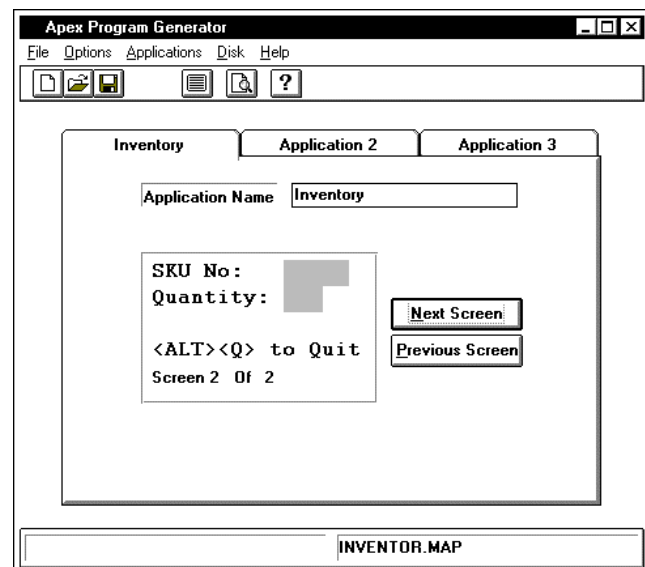
091

The application builder now contains the SKU and Quantity prompts and input fields.

We want to remind people how to quit the program on this screen too.

Type “<ALT><Q> to Quit” on the bottom line of the display box.

Then save the project again (select File, and then Save Project).



092

We have now defined all the input fields. We will define lookup files for the Employee, Aisle, and SKU fields later (page 41).

**Step 6**

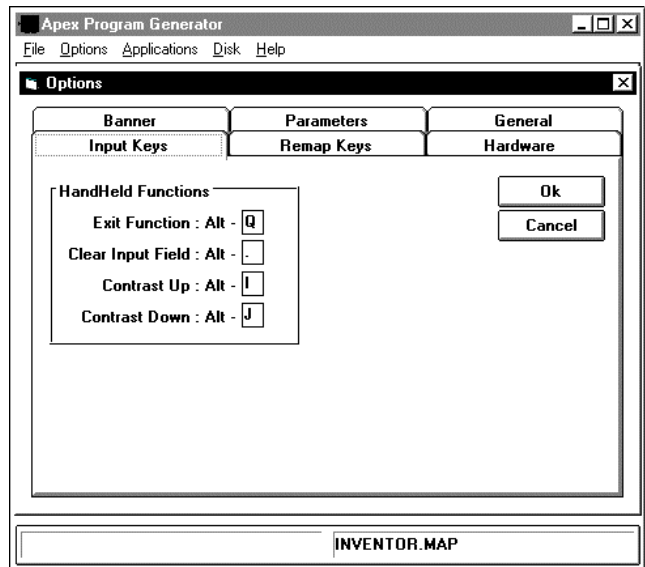
**Set the options.**

Global operating options control Apex II operations. These options are set for the whole project (up to three applications) rather than individually set for each of the applications.

Select the Options main menu option.  
Then select the Input Keys tab.

During operations, the defined keys are used to perform the functions listed in the Input Keys tab.

Since we kept the default settings (page 16), just view the tab.



043

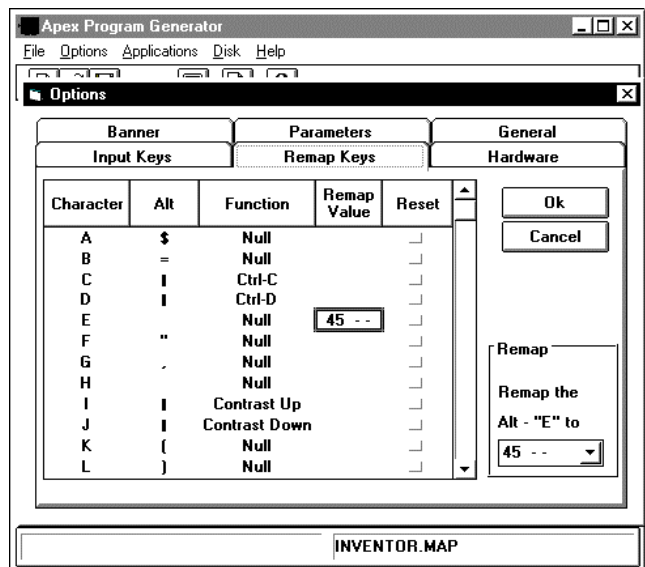
When special characters are collected as part of the data, you need to assure that they are mapped (assigned to a key and used in combination with the ALT key for special characters). We used a hyphen (-) during the planning on page 16.

Select the Remap Keys tab from the Options window.

Scroll through the Key Remap grid, paying special attention to the Alt column. Since the hyphen does not appear in this column, it is not mapped.

Scroll to the letter E. Since it is not currently mapped with a special character, we will map the hyphen to it.

Select the letter E. Select the Remap drop down list in the lower right of the window. The list displays a menu with ASCII codes and characters. Use the down arrow key to scroll through the list. Select number 45 - hyphen.



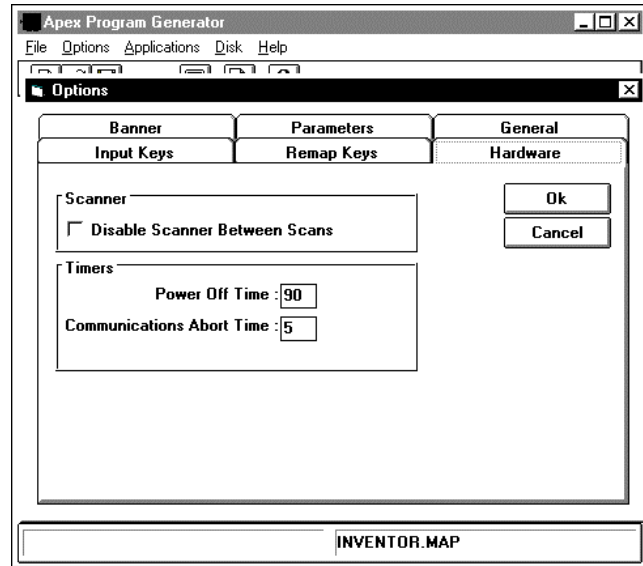
018

Select the Hardware tab from the Options window.

During operations, the handheld unit powers off after a specified period of inactivity (Power off time).

Communications are canceled after a specified period of unsuccessful communication effort (Communications Abort Time).

Since we kept the default settings (page 16), just view the tab.

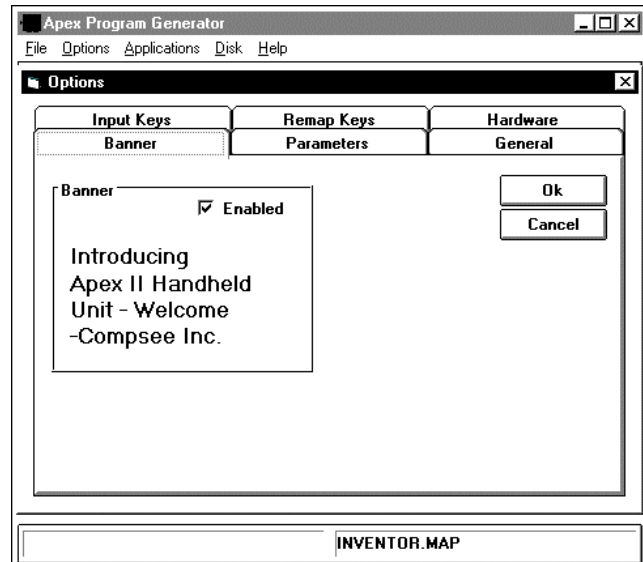


039

Then select Banner from the Options window.

During Apex II operations, the banner is displayed immediately after the program is downloaded.

Using the screen planning on page 16, type the message for the Banner.



093



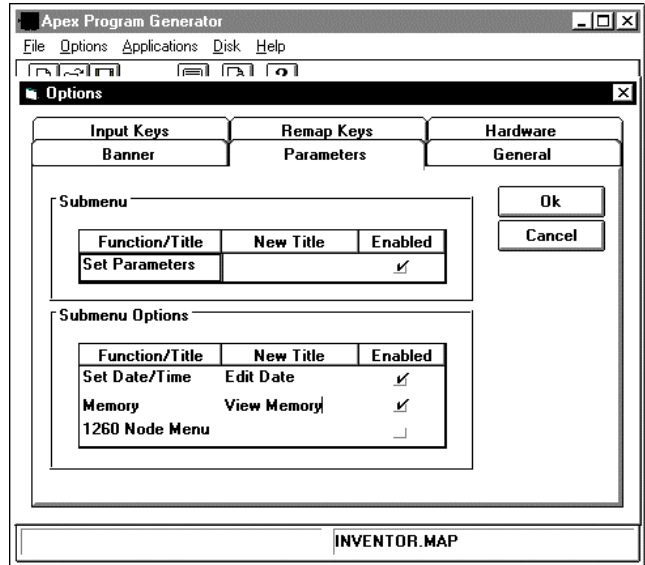
Select the Parameters tab from the Options window.

During operations, this menu is displayed when the F4 function key is pressed.

Set options as defined on page 16 and this figure.

Assure that the appropriate Enabled option boxes are selected.

Then press OK to return to the main menu.

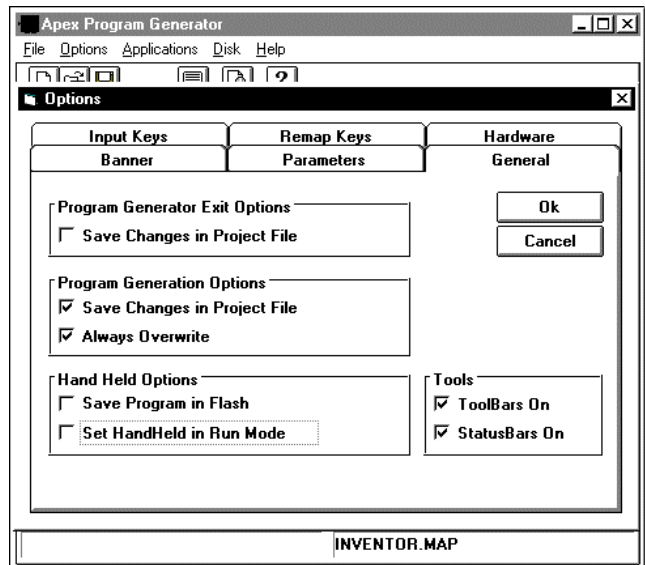


140

Select the General tab from the Options window.

Select any or all of the option boxes to indicate your preferences.

Then press OK to return to the main menu.



019

Step 7

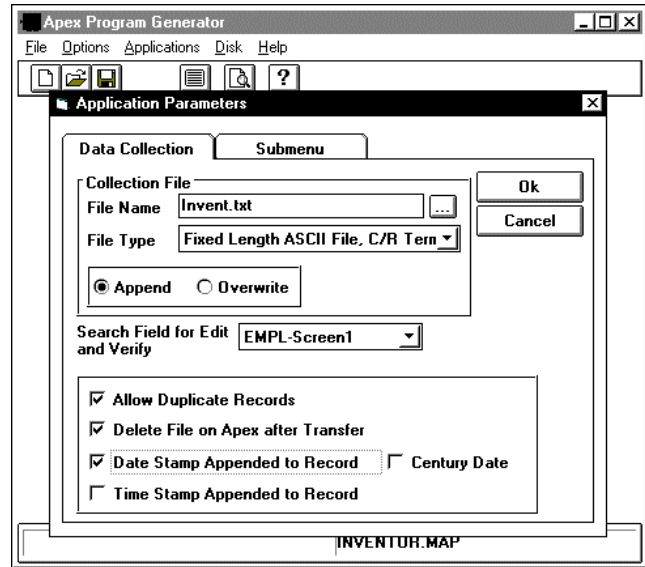
Set up the data file and sub menu.

Select the Applications main menu option. Then select Parameters. The Data Collection tab is active.

During operations, the scanned and keyed input data is stored in the file defined on this window.

Using the planning on page 8 and this figure, define the data collection file.

Then press OK to return to the main menu.



004

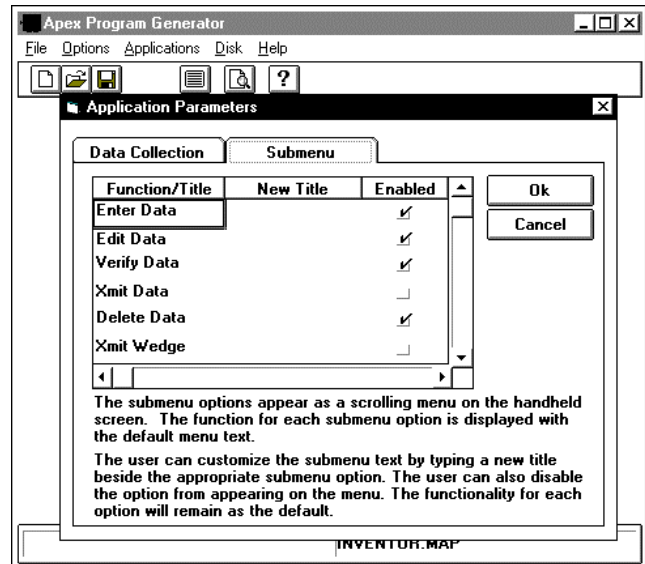
Select the Submenu tab from the Application Parameters window.

During operations, the “Enter Data” option from this submenu is displayed on the handheld unit’s main menu. When you press the arrow keys from the main menu, the rest of these menu options (if enabled here) are displayed.

We decided to use the default titles listed in this figure. Page 13 describes the options.

Assure that the Enabled option boxes are selected.

Then press OK to return to the main menu.



005

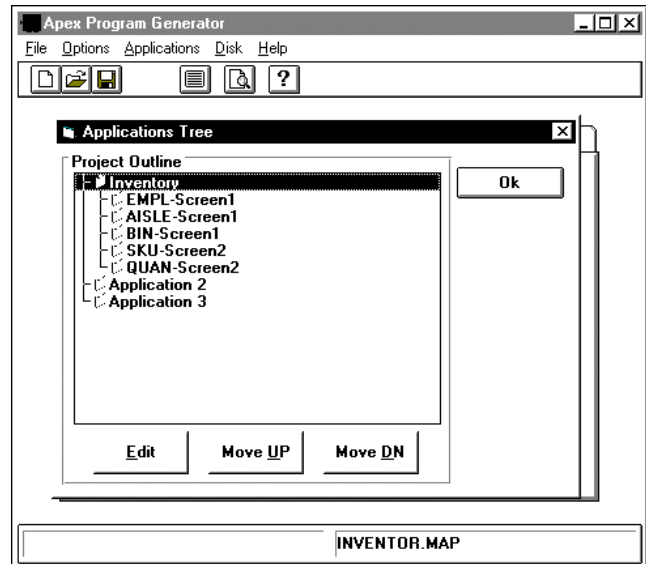
**Step 8** **Check the order of data collection in your program.**

Select the Applications main menu option.  
Then select Project Outline.

Since we defined the input fields in order,  
just view this screen.

What if you do define the input fields in one  
order and then want to rearrange the order in  
which they are activated, so that a person can  
input data in a different order? Use the buttons  
at the bottom of the window to edit and move the  
input fields.

Then press OK to return to the main menu.



097

**Step 9** **Generate and download the application code.**

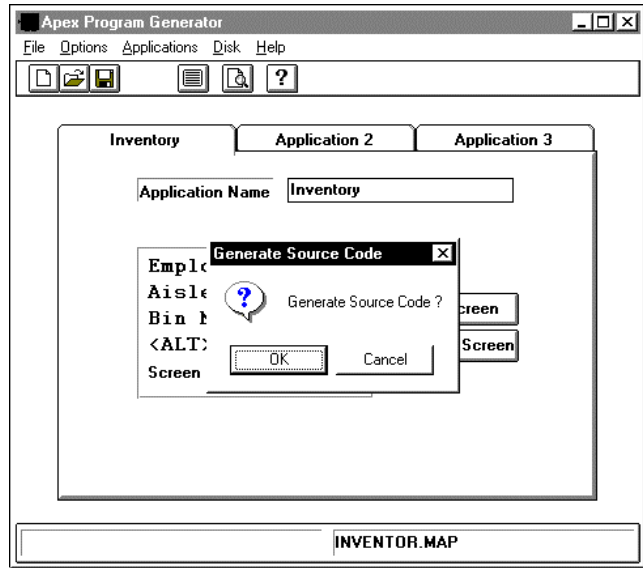
Before we complete our project, we want to download it and see how we have done so far.

Select the File main menu option.  
Then select Generate Code.

A message prompts you to confirm that you want to generate code.

Select Yes.

A status bar at the bottom of the Program Generator window indicates that the code generation file is being written.



150

Select the File main menu option.

Then select Download Files.

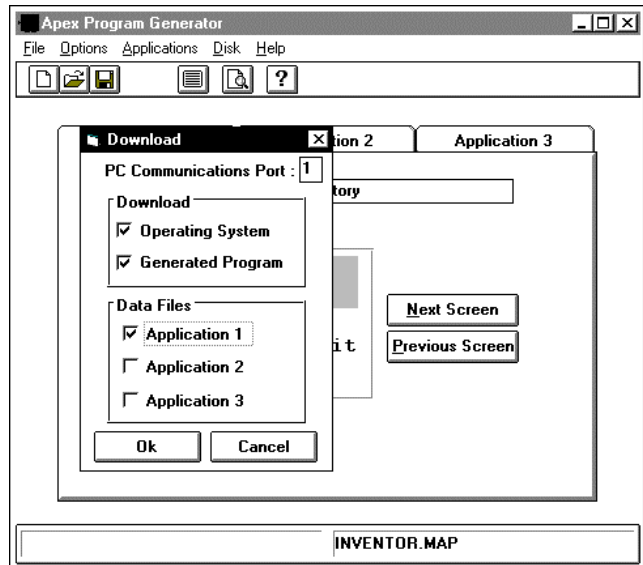
The Download Files Window is displayed.

Be sure that the PC Communications Port selected on this screen identifies the physical connection from your PC to the handheld unit.

The Communications Port may be changed by selecting another port from this window.

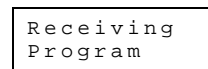
A status bar and message are displayed at the bottom of the PC window:

Downloading Program.



102

A message is displayed on the handheld unit:



When the download is complete, the PC screen displays a prompt:

Send Save and Set Apex in Run Mode?

If you are using an MEQ rather than an Apex II, the PC screen displays the following prompt:

Set Handheld in Run Mode?

Select OK. The banner you designed is displayed on the Apex II display.

**Step 10****Run the program on the Apex II.**

After you download the program, use it to see how the generated program works.

When the program is finished downloading, the Banner is displayed.

```
Introducing the
Apex II Handheld
Unit - Welcome!
Compsee Inc.
```

Then the program automatically advances to the main menu. Application 1 (Inventory) is accessed by pressing the F1 function button.

```
F1 Inventory
F4 Parameters
```

When you press F1, the sub menu for Inventory is displayed. Press the arrow keys to scroll through the sub menu.

```
Sub Menu For
Inventory
Enter Data
Press Enter
```

Each time you press the arrow key, another sub menu option is displayed.

```
Sub Menu For
Inventory
Edit Data
Press Enter
```

We planned these sub menu titles on page 13.

```
Sub Menu For
Inventory
Verify Data
Press Enter
```

```
Sub Menu For
Inventory
Xmit Data
Press Enter
```

```
Sub Menu For
Inventory
Xmit Wedge
Press Enter
```

```
Sub Menu For
Inventory
Delete Data
Press Enter
```

```
Sub Menu For
Inventory
Return to Main
Press Enter
```

When you scroll back to the Enter Data option, Press Enter.

```
Sub Menu For
Inventory
Enter Data
Press Enter
```

The first screen is displayed. Enter the appropriate data.

```
Employee No: _ _ _ _
Aisle No.: _ _ _ _
Bin No.: _ _ _
<ALT><Q> to Quit
```

**NOTE**

If you have defined lookup files, use valid data (page 12).

After you enter the Bin number, the second screen is displayed. Again, enter the appropriate data. The first screen is displayed again.

```
SKU No.: _ _ _ _
Quantity: 1 _ _
<ALT><Q> to Quit
```

If you have defined looping (page 48), the bin number is active.

```
Employee No: 0 0 0 1
Aisle No.: A A 0 1
Bin No.: _ _ _
<ALT><Q> to Quit
```

Otherwise, the Employee field is active.

```
Employee No: _ _ _ _
Aisle No.: _ _ _ _
Bin No.: _ _ _
<ALT><Q> to Quit
```

Enter 2 or 3 more records.

Then press <ALT><Q>.

The main menu is displayed.

To edit the date and time or view the available memory, press the F4 function button.

```
F1 Inventory
F4 Parameters
```

The sub menu for Date/Memory (Parameters) is displayed.

```
Sub Menu For
Parameters
Edit Date
Press Enter
```

We defined this on page 33. Press Enter.

If date and time are not correct, enter the current date in MM/DD/YY format.

```
Enter Date/Time
07/18/96      14:57
```

Then enter the current time in HH:MM format.

The sub menu for Date/Memory is again displayed.

```
Sub Menu For
Parameters
View Memory
Press Enter
```

Press Enter.

The display indicates the available amount of memory in bytes.

```
83301 bytes of
Memory Remaining
```

Then the display indicates the number of records currently stored in the Apex II.

```
5 Records
in this terminal
```

Finally, the display indicates the number of additional records the Apex II can store.

```
Room for Another
3407 Records
```

The sub menu for Date/Memory is again displayed.

Press Enter.

```
Sub Menu For
Date/Memory
Return to Main
Press Enter
```

The program returns to the main menu. Continue to the next step to transfer the files to your PC.

```
F1 Inventory
F4 Parameters
```

The complexity of the generated program determines how many records the Apex II can store. Our program has five fields in each record. A program in which each record contains many long fields stores fewer records than ours. In addition, a complex program uses more Apex II memory than a simple one.

In any case, the Apex II can comfortably store a day's worth of inventory. We recommend that you upload data at the end of the workday or inventory task, whichever comes first. This way, your PC contains the latest inventory data.

Step 11

Retrieve data from the Apex II.

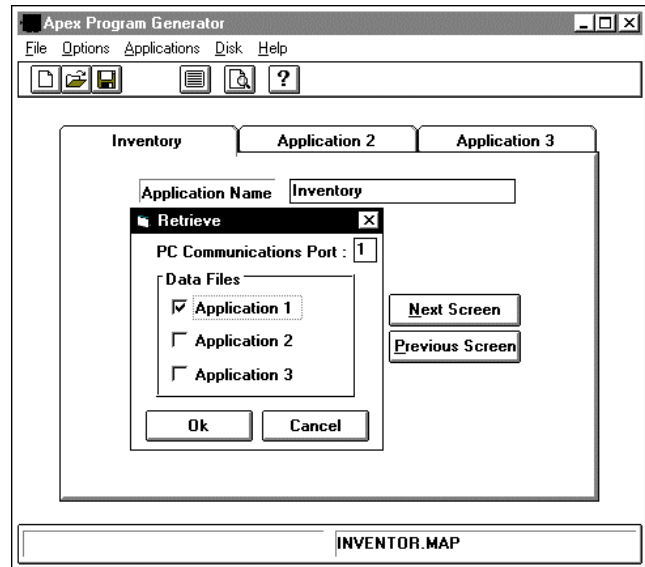
Select the File main menu option.  
Then select Retrieve Files.

The Retrieve Files Window is displayed.

Since we defined communications Port 1 on the hardware options, (page 31), be sure that PC Communications Port 1 is selected on this screen.

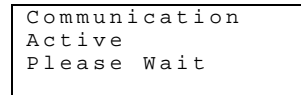
This field may be changed if Port 1 is not available.

Select OK.



040

The Apex II display indicates that communication is occurring.



The Program Generator window (PC screen) displays a status bar, indicating that the data collection file is being received.

When the data collection file is received, the following prompt is displayed on the PC screen:

Delete Application 1's Data Collection File?

Select OK. Another prompt is displayed:

Set Apex in Run Mode?

Select OK. The Receive window (PC) is removed. The banner is displayed on the Apex II.

To view the data collection file, open it in a word processor, DOS, Dbase, or Access, depending on how you defined the data collection file on page 34.

Since our data collection file (Invent.txt) is ASCII with a carriage return separating records, we can open the file in DOS or any word processor.

The following figure illustrates how the file looks:

**Invent.txt File**

```

0001AA001123A-005001960818¶
0001AA001123B-050100960818¶
0001AA001123B-015001960818¶
0001AA001123A-0602 960818¶
0002BB001001A-05503 960818¶

```

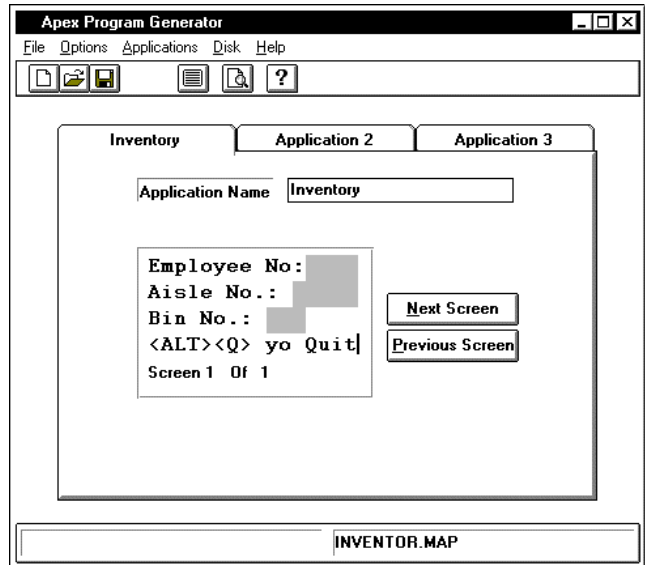


**Step 12** Set up lookup files, definitions and displays.

After completing Steps 10 and 11 successfully, we have a program that works on the Apex II. We are now going to make the program be more selective about what data it accepts.

Return to the first screen we defined. Double click in the Employee No Input field (the highlighted area to the right of the words "Employee No:")

The Data Field Definition window is displayed.



064

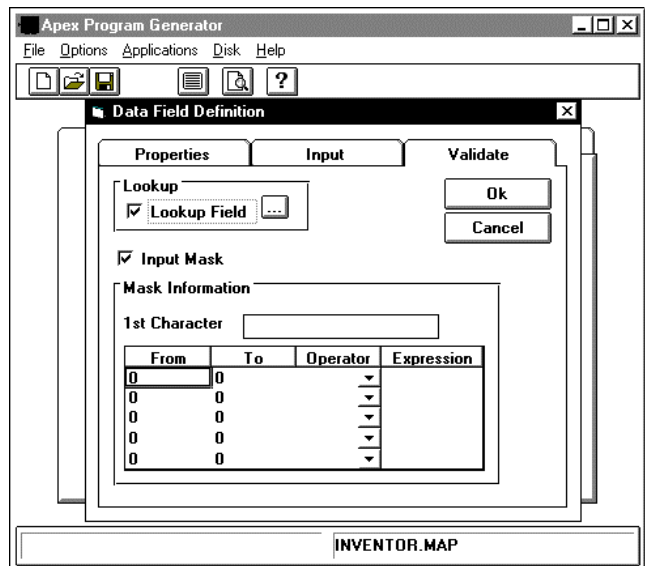
Select the Validate tab from the Data Field Definition window.

Then select the Lookup Field option.

The "... " button to the right of the Lookup field option becomes activated (changes color).

Select the "... " button.

The Lookup Definition window is displayed.

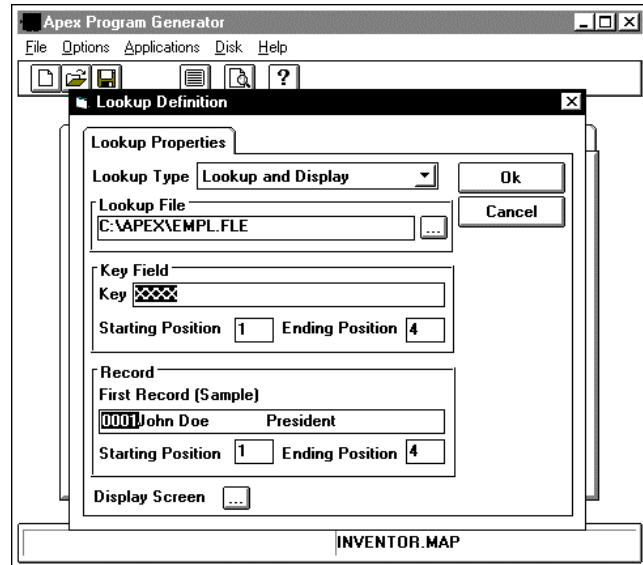


105

Using the lookup file planning on page 13 and this figure, select the lookup type and file name.

Then select all four X's representing the four characters in the key field (Employee number). This tells the program to look up the complete field.

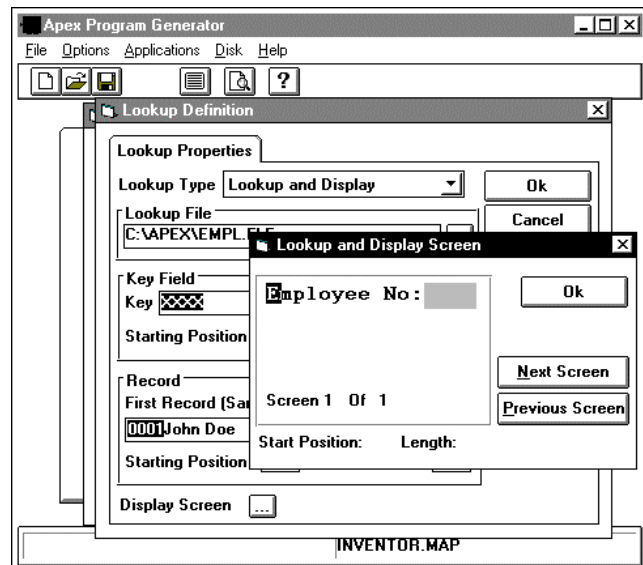
Finally, select the key field in the first record (0001).



123

Since we want to display employee information contained in the lookup file, select the Display Screen "..." button at the bottom of the screen.

The Lookup and Display Screen window is displayed.



122

We are now designing the handheld unit screen that is displayed after a person enters a valid employee number.

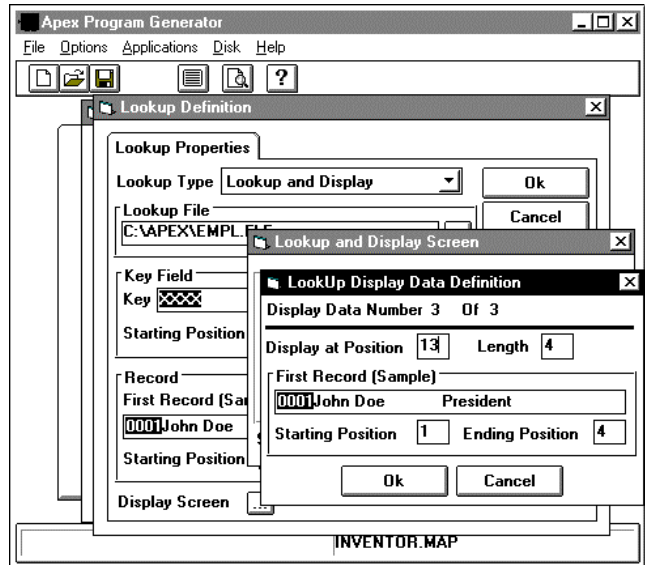
Type "Employee No:" (no quotation marks). Then click and drag the mouse across the four spaces on the right of the first line.

The Lookup and Display Data Definition window is displayed.

The Lookup and Display Data Definition window prompts you to define what is displayed in the four spaces we just highlighted on the Lookup and Display screen window.

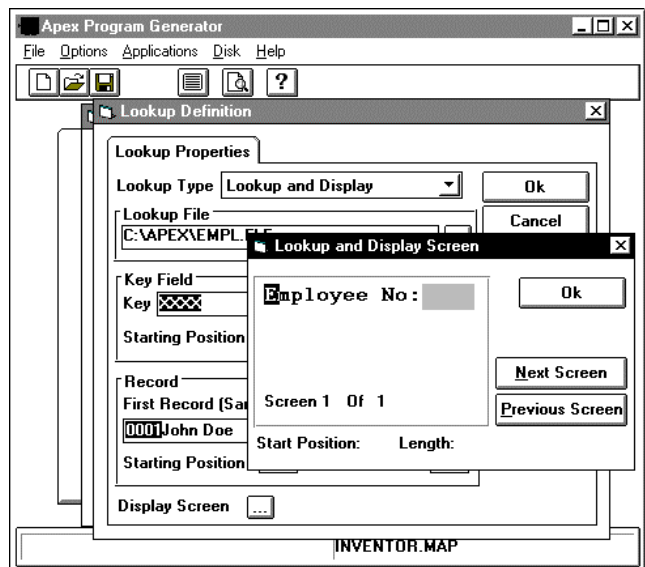
Select the first four characters in the first record (0001).

Then select OK.



121

The Lookup and Display screen window is again displayed.



122

We want to display the rest of the employee information.

The program prompts us to define which positions in a lookup file record we want displayed on which lines of the Apex II screen.

There are 16 characters on each line of the Apex II display.

The Employee Lookup File is 32 characters long.

We have marked the first four characters to be displayed on the first line of the Apex II display.

To decide how to set up the rest of the display, we need to study the lookup file.

### NOTE

A lookup file must have the same number of characters in each line. If it does not, the program will not be able to use it. We have constructed the lookup files for you. Notice that each line ends in a character. The program does not recognize spaces at the end of a line. Also notice that the file does not end with a carriage return. If it did, this would be a line of zero length, and the program could not use this lookup file.

### Employee Lookup File

0001	John Doe	President	X
0002	Mary Smith	Engineer	X
0003	Frank Sanchez	Engineer	X
0004	Carry Mahoney	Purchasing	X
0005	Frances Coyle	Production	X
0006	Marilyn Hill	Tech. Sup.	X
0007	Ron Jones	Sales Mgr.	X
0008	Claudia Anderson	Engineer	X
0009	Lou Goldman	Office Mgr.	X
0010	Lee French	Software M	X

In this lookup file, the first four characters are the employee number. The longest employee's name (Claudia Anderson) uses characters 5 through 20. The longest employee's title (Office Mgr.) uses characters 22 through 32. Character 33 is an X to identify the end of the line.

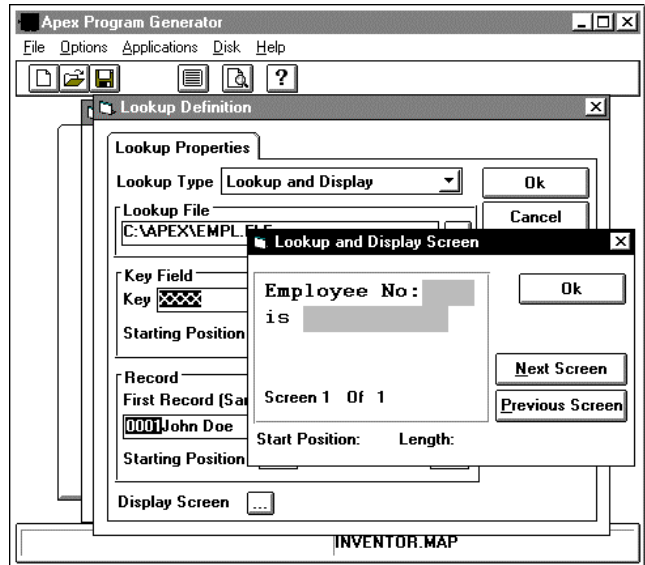
We are going to put the employee's title on the second line of the display and the employee's name on the fourth line of the display.

### NOTE

An X at the end of each line (30th character) in the SKU description file assures that each line is the same length. The program does not recognize spaces at the end of the line.

Type "is" (no quotation marks) on the second line of the Lookup and Display screen window. Then click and drag the mouse on the right of the second line, beginning at space 4 with a length of 11.

The Lookup and Display Data Definition window is displayed.



142

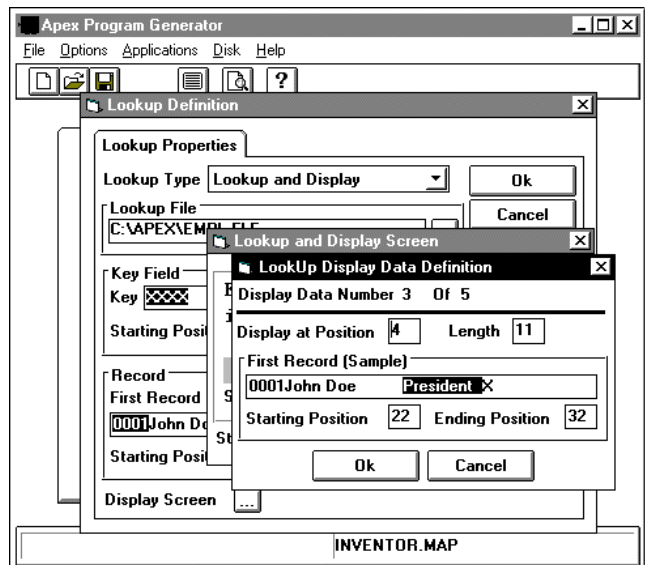
The Lookup and Display Data Definition window prompts you to define what is displayed in the 11 spaces we just highlighted on the Lookup and Display screen window.

Select the characters 22 through 32 in the first record. Begin with the first letter of the word "President" and drag the highlighting to, but not including, the X at the end of the line.

We identify characters 22 through 32 to accommodate the longest employee title.

Then select OK.

The Lookup and Display screen window is again displayed.

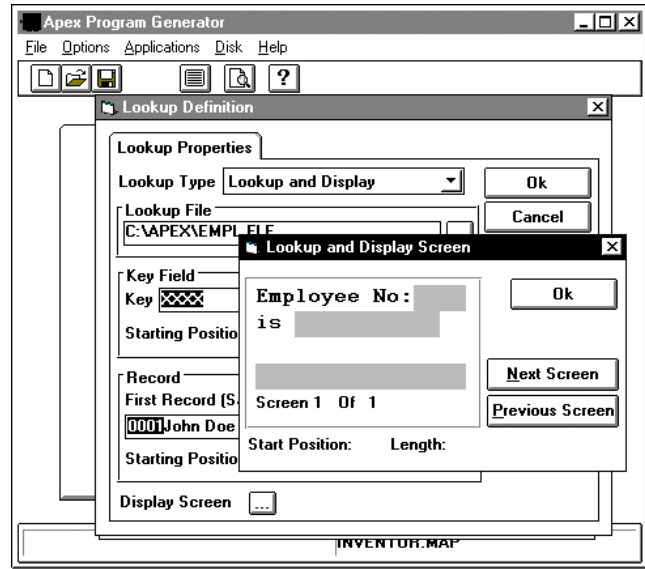


141

Click and drag the mouse across the bottom line, across spaces 1 to 16.

The Lookup and Display Data Definition window is displayed.

The Lookup and Display Data Definition window prompts you to define what is displayed in the 16 spaces we just highlighted on the Lookup and Display screen window.



144

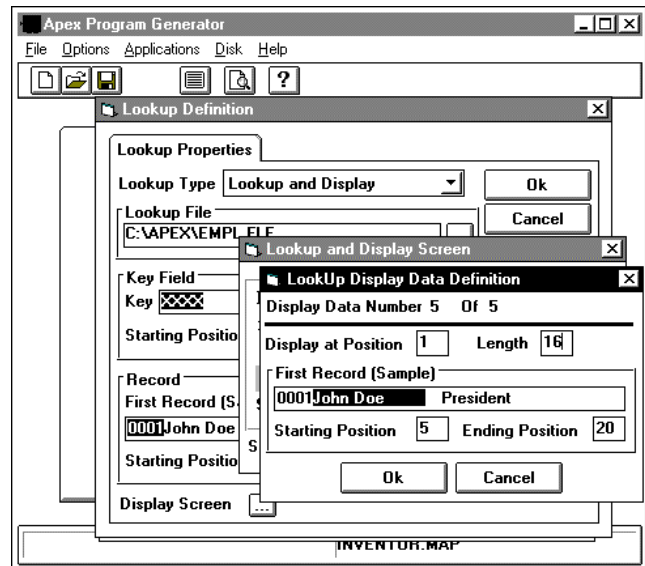
Select the characters 5 through 20 in the first record. (The words “John Doe” and 8 spaces after them).

We identify characters 5 through 20 to accommodate the longest employee name.

Then select OK.

The Lookup and Display Screen window is again displayed.

Select OK to return to the main menu.



151

You have now defined the Employee lookup file.

Save the project (select the File main menu option, then select Save).

You may want to download (page 36) and run the program again to see how you are doing so far.

Two more lookup files need to be defined: the AISLE.fle and the SKU\_DES.fle. The same windows are used in defining these lookup files.

We planned the lookup files on page 13.

Redefine the AISLE field as you redefined the EMPL field.

Select the "Must Exist" lookup type and the AISLE.fle file name.

Then select all five X's representing the five characters in the key field (Aisle number). This tells the program to look up the complete field.

Finally, select the key field in the first record (AA001).

**Aisle Lookup File**

```
AA001¶
AA002¶
AA003¶
BB001¶
BB002¶
BB003¶
CC001¶
CC002¶
CC003¶
DD001¶
DD002¶
DD003¶
```

Then select OK.

The Lookup and Display screen window is again displayed.

Redefine the SKU field as you redefined the EMPL field.

Select "Lookup and Display" lookup type and the SKU\_DES.fle file name.

Then select all five X's representing the five characters in the key field (SKU number). This tells the program to look up the complete field.

Set the program to display characters 1 through 5 on the first line, characters 7 through 20 on the third line, and Characters 21 through 29.

```
SKU No
Description:
```

**SKU Description Lookup File**

```
A-005 Apex Standard 128K X¶
B-010 Apex Standard 512K X¶
B-015 Apex Standard 1 Meg X¶
A-020 Apex Integ. 128K X¶
B-025 Apex Integ. 518K X¶
B-030 Apex Integ. 1 Meg X¶
B-050 Cab. Apx-PC 9 pin 6ft X¶
A-050 Cab. Apx-PC 25 pin 6ft X¶
A-055 Cab. Apx-QR 6ft X¶
B-055 Cab. Apx-QL 6ft X¶
A-060 Apex Battery Pack X¶
B-060 AA Alkaline 100 / Case X¶
```

**NOTE**

An X at the end of each line (30th character) in the SKU description file assures that each line is the same length. The program does not recognize spaces at the end of the line.

**Step 13**

**Set up loop, math registers, and branching.**

Select the Application’s main menu option.

Then select the Edit option.

Select the Setup Loop to Field edit option.

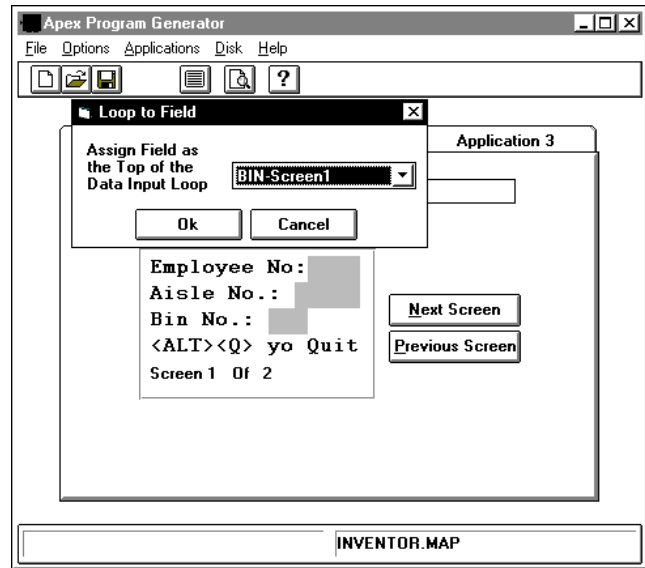
The Loop To Field dialog window is displayed.

We planned the application looping on page 15.

Assign the BIN field as the top of the data input loop.

Then select OK.

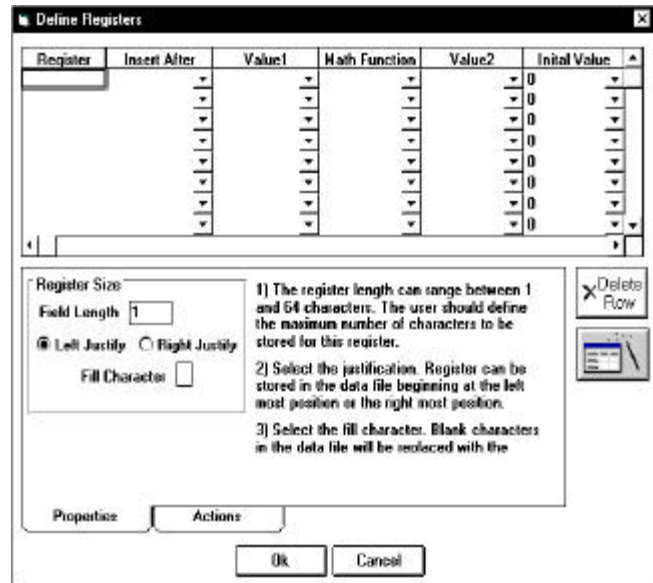
The main menu is displayed.



148

Select the Applications main menu option, the Edit option, and then select the Setup Math registers edit option.

The Define Registers Window is displayed. The field Parameters tab displays the data format information.



On page 7 and page 15, we planned to let people just enter the last three digits of the Bin Number and make the system always add the “B-“ at the beginning of the field. We can accomplish this by using a register. The Math Wizard will help us.

It is located on the right side of the screen directly below the delete icon. All registers can be created and edited using the Math Wizard or the grid. We are going to use both.



Place the cursor on in the blank register at the top left of the Math Register window.  
Then select the Math Wizard button.

The first Math Wizard window is displayed.

For item 1, name the register BIN\_Num.

For item 2, select the field button, and BIN from the drop down list.

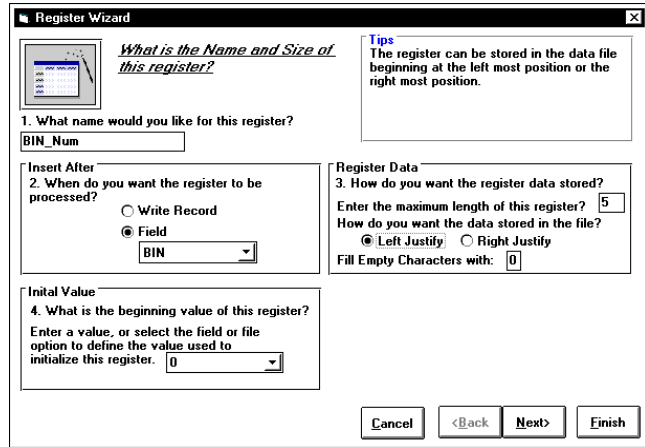
For item 3, type 5 as the maximum length of this register.

Leave the Left Justify button selected.

Leave the Fill Empty Character with: box empty.

For item 4, leave the beginning value at zero.

Select the Next button.



007

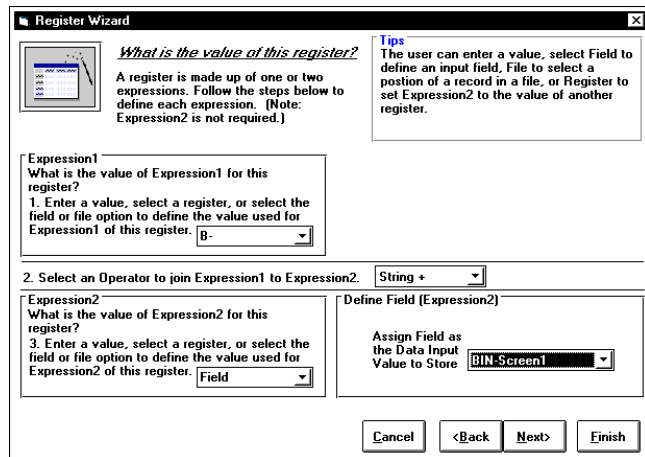
The second Math Wizard window is displayed.

For item 1, type a B and a hyphen (B-) in the data entry box.

For item 2, select "String +" from the drop down list. This tells the system to join the two values instead of performing a calculation.

For item 3, select the field option from the drop down list. An additional prompt is displayed. Assign BIN-Screen1 from the drop down box.

Select the Next button.



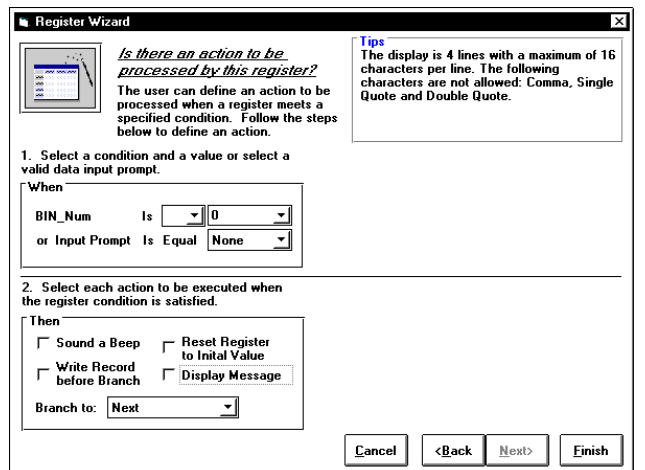
009

The last Math Wizard Window is displayed.

Since we enter a Bin Number for each record in the system, we do not need to tell the system to perform an action, such as reset the register, when the register meets a specified condition.

Therefore, leave items 1 and 2 as they are (displayed to the right).

Select FINISH to save the register information and return to the Math Registers window.

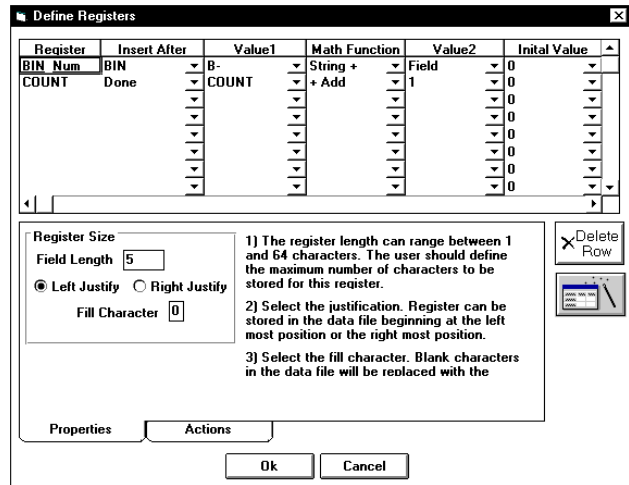


010

On page 15, we planned to have the program count each record that is entered into the Apex II and to display a message when 1000 or more records are stored in the handheld unit.

We want the program to have a register called COUNT, having an initial value of 0.

Each time a record entry is completed, we want the register to increase in value by 1. Complete the second line and bottom half of the Define Registers window as shown.



041

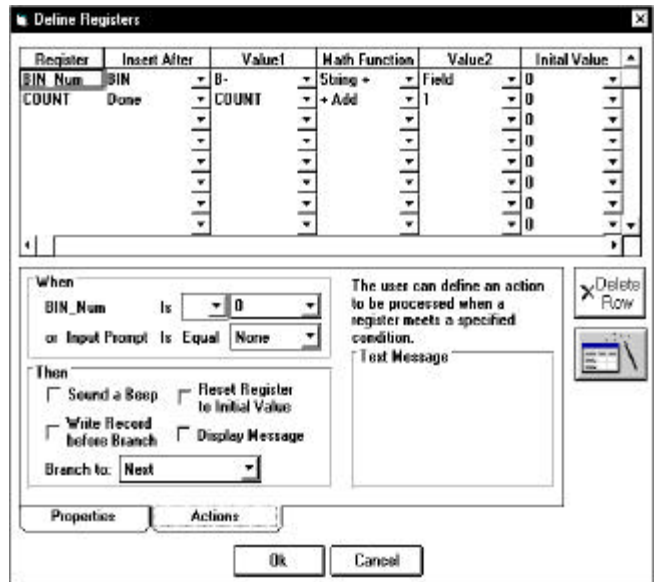
Then select the Actions tab at the bottom of the window. More prompts are displayed.

Select the arrow to the right of the words, “when COUNT is.” The equal to (=), greater than (>), and less than (<) symbols are displayed.

Select “>” and then type 999 in the “When Count is” field.

Select the Beep and Message option boxes and leave the Reset option box deselected. Then type the message we planned on page 7.

Select OK to return to the main menu.



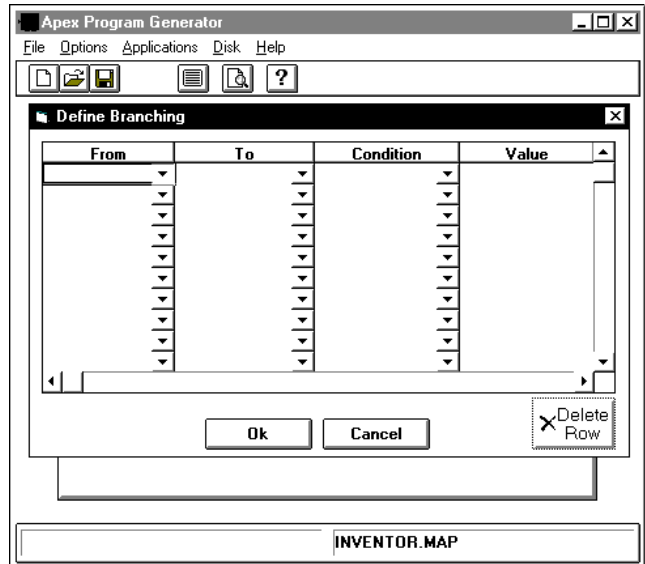
021s

We described branching on page 15.

To view the Define Branching window, select the Applications main menu option, the Edit option, and then the Define Branching edit option.

Since we decided not to use branching, (page 15) just view the window.

Select OK to return to the main menu.



024

## Step 14

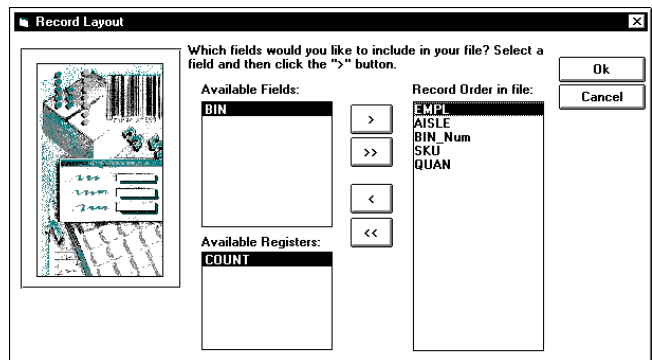
## Define record layout.

We planned the record layout on page 7.

Select the Application's main menu option.

Then select the Edit Option.

Select the Record Layout option to display the Record Layout window.



022

The available fields box displays all input fields within the system.

The available registers box displays all registers defined.

Our records are stored in the following order: EMPL, AISLE, BIN, SKU, and QUAN.

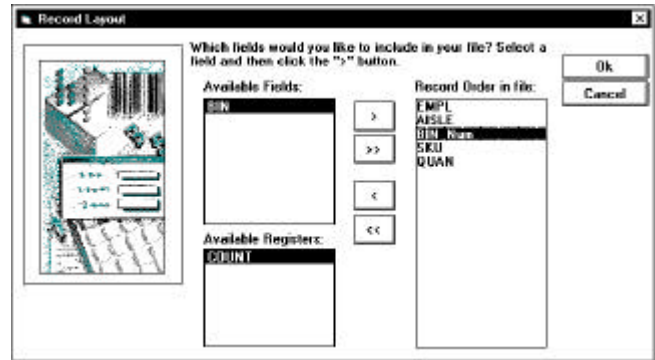
In our planning (page 7 and page 15), we decided to let people just enter the last three digits of the Bin Number and make the system always add the "B-" at the beginning of the field. We accomplished this in Step 13 when we defined the BIN\_Num register. We want to store the modified, rather than the entered, Bin Number in the file.

Select EMPL from the available fields and then press the > button to move the field to the record order list box. Follow the same steps to include the AISLE field.

Select Bin\_Num from the available register and then select the > button to move the register to the record order list box..

Select SKU and QUAN from the fields box to be included in our file.

Select OK to return to the main menu.



011

### **Step 15**                      **Download, run and retrieve data from the completed program.**

Some of us find it helpful to try downloading, running, and uploading each time we add a loop, branch, or register. We can check our work as we go, and are sometimes surprised at what we told the program to do.

After you become familiar with the Program Generator, this is not necessary.

To download the completed program, follow directions starting on page 36.

To run the completed program, just follow the directions you programmed!

To retrieve the inventory data, follow directions starting on page 40.

### **Step 16**    **Document and check your program.**

The Program Generator documents the projects you generate. This is especially helpful while you are learning to use the Program Generator. The document file generated is the project name with a "Rpt" extension (Inventor.Rpt).

To document your project, first assure that the project is currently open.

Then select the File main menu option and the Document Project option.

A message is displayed:

```
Documentation is complete.  
A text file named C:\APEX\Inventor.Rpt was created
```

Open Inventor.Rpt in any word processor. Setting the font to courier makes the file easy to read.

The documentation you print is similar to the planning described on pages 5 through 17.

If the program does not run on the Apex II in the way you expected it to, print the documentation and compare it to your planning.

## Step 17

## Create a diskette.

After you generate a project, you will probably want to distribute it to operators who use it to download the programs to the handheld unit and upload data from the handheld unit.

Creating an Application Manager Diskette facilitates distribution of the program.

When the Application Manager Diskette (or diskettes) are completed, people have access to the program you generated, without having access to the Program Generator.

Obtain three blank diskettes for this task.

Select the Disk main menu option. Then select Create Application Diskettes option.

The Application Manager Diskettes Setup window is displayed.

Type the Application Title (Inventor.map), the Local Application Prompt (Inventory), and the Prompt for Application 1 (Inventory App).

Then select Make a Disk.

147

A prompt asks if you want to save the current project setup and regenerate the program before creating the Application Manager Diskettes. Select Yes.

After the program saves the project and generates the code, the following prompt is displayed:

Insert disk 1 into drive A:\ and press OK to begin copying files.

Follow the directions on the screen. The screen prompts you to insert additional diskettes as needed. Label the diskettes appropriately.

The program displays a message when the Application Manager Diskettes are completed.

Next, use the diskettes to see how other people will work with the program you generated.

## Apex II Program Generator

---

Install the first Application Manager diskette.

Close all unnecessary Windows programs.

From windows or DOS, run the Setup.exe program on the diskette.

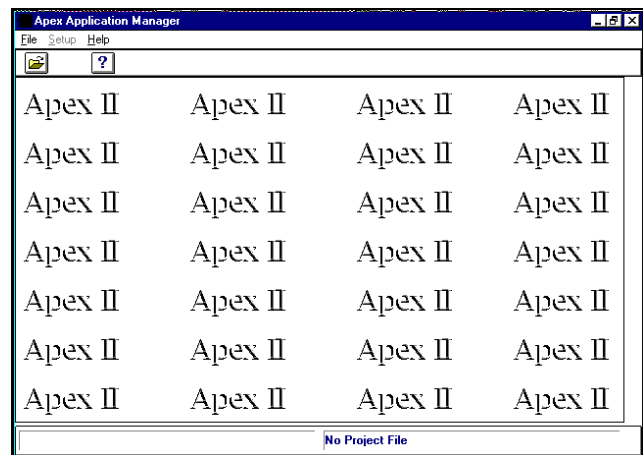
The Setup program prompts you to accept or edit the installation directory for the Application Manager files (C:\APEX3MGR).

Select continue. The program prompts you to insert additional diskettes.

When installation is complete, the program displays a message. When you select OK, the APEX3MGR icon (a barcode) is displayed.

Select the APEX3MGR (barcode).

The Apex Application Manager window is displayed.

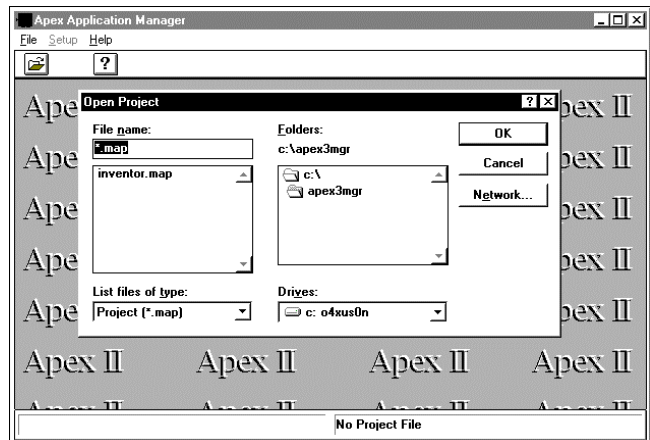


001

Select the File main menu option.  
Then select the Open Project option.

The application title we typed on the Application Manager Diskette Setup window (page 53) is displayed on the left side of the window.

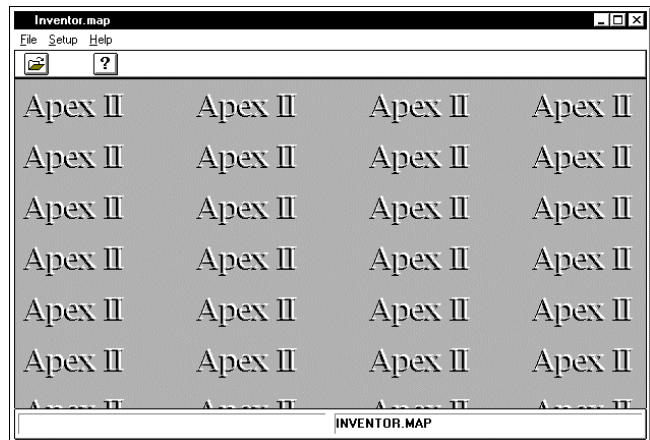
Select Inventor.map. Then select OK.



002

The Apex Application Manager window is displayed again.

Now the title bar and bottom line of the window display the application title (Inventor.map).



003

Select the Setup main menu option. A submenu is displayed, listing communications ports.

People use this menu to select the physical communications port for your program to use during data transfer.

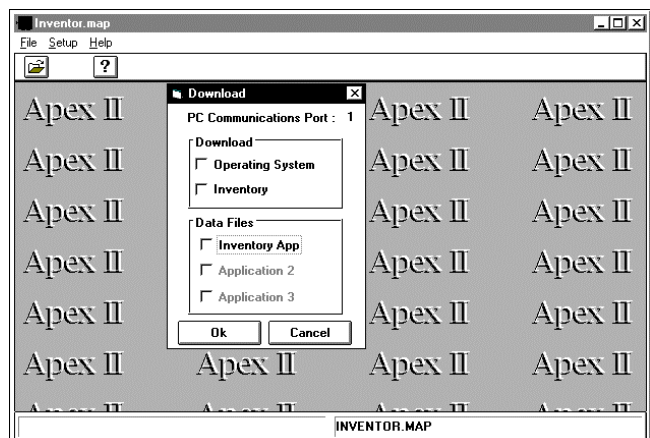
Select the File main menu option.

Then select the Download Files option.

The Download window of the Program Generator is displayed.

People use this window as we described Downloading on page 36.

Select cancel to close the window.



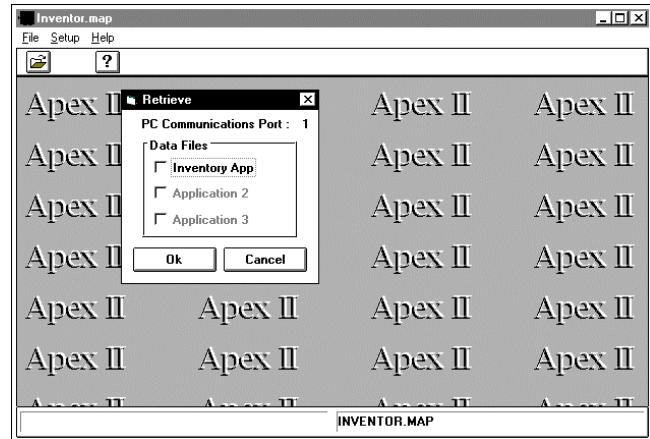
Select the File main menu option.

Then select the Receive Files option.

The Receive window of the Program Generator is displayed.

People use this window as we described Downloading on page 40.

Select cancel to close the window.





## **Empty Forms for Planning a Project**

---

---

Use the following forms as you plan your own project.

### **Step 1** **Decide what you want the project to do.**

Application \_\_\_\_\_

1. \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

### **Step 2** **Decide how you want the application to do it.**

Application \_\_\_\_\_

1. \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_


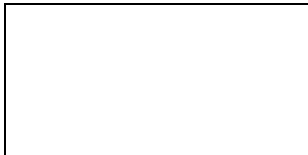




### **Step 3** **Decide how many applications you need to accomplish these tasks.**

- \_\_\_\_\_
- \_\_\_\_\_



**Step 4**

**Decide what you want the screens to look like.**

Application \_\_\_\_\_ Input Screens

Screen/Line	Handheld Unit Display
Screen _____ Line 1 Line 2 Line 3 Line 4	
Screen _____ Line 1 Line 2 Line 3 Line 4	
Screen _____ Line 1 Line 2 Line 3 Line 4	
Screen _____ Line 1 Line 2 Line 3 Line 4	
Screen _____ Line 1 Line 2 Line 3 Line 4	
Screen _____ Line 1 Line 2 Line 3 Line 4	

**Field Lookup and Display Screen**

	<b>Handheld Unit Display</b>
Line 1 Line 2 Line 3 Line 4	
	<b>Handheld Unit Display</b>
Line 1 Line 2 Line 3 Line 4	

**Step 5                  Decide how many input fields are needed and the program layout.**

Application \_\_\_\_\_

- \_\_\_\_\_
- \_\_\_\_\_
- \_\_\_\_\_
- \_\_\_\_\_
- \_\_\_\_\_

**Step 6                  Decide how to collect the data.**

Application \_\_\_\_\_

Data Collected in the \_\_\_\_\_ File


**Application** \_\_\_\_\_

**Data Collection Planning**

Operating Parameter	Value Set
Collection File Name	
Do you want to overwrite the contents of your data collection file, or do you want to append new records to it?	
<p>What type of PC File do you want for data collection?</p> <p>Fixed Length ASCII file.</p> <p>Fixed Length ASCII File with Carriage Returns Term Records.</p> <p>Comma Quote Delimited File (separate each record with a comma).</p> <p>Custom Quote Delimited ASCII File. (separate each record with a character that you define).</p> <p>Dbase File.</p> <p>Access Database File.</p>	
<p>Data Collection - Handheld unit:</p> <p>Do you want to allow people to create duplicate records; that is, with identical data in one of the fields?</p> <p>If you do not want duplicate records, which field do you want to be unique?</p>	
Do you want the date for each record, without having to key it? Do you want to include the century in the date?	
Do you want the time for each record, without having to key it?	
Do you want to delete the data from the handheld unit after it is transferred to the PC text File?	
Do you want power to the Laser turned off between scans?	

**Step 7**

**Design the input fields.**

Application \_\_\_\_\_

Design Consideration	Our Field Decisions				
What's a short, easy to remember field name?					
Which screen is it on?					
How long is this field including spaces and special characters?					
Do you want this field to accept letters? Do you want this field to accept numbers? Do you want this field to accept Special Characters?					
Do you want this field to be required? Do you want this field to display a default value when the screen is first displayed? If so, what?					
Do you want the program to only allow certain first characters? Which ones? Do you want the field to accept only values in a certain range? Minimum Range Value: Maximum Range Value:					
What are the smallest and largest number of characters you want people to enter for this field? Minimum Field Length: Maximum Field Length:					
After people scan or key data, should the cursor automatically advance to the next input field? Do you want people to be able to scan Input? Do you want people to be able to key Input?					

**Step 8** Define the bar code scanning abilities of each input field.

Application \_\_\_\_\_

<b>Bar Code/Options</b>					
Codabar Minimum length? Codabar pairs?					
Code 3 of 9 Full ASCII?					
Code 11 Number of check digits? (1 or 2)					
Code 93					
Code 128					
EAN - 8					
EAN - 13 Require 2 or 5 Character Supplemental Code?					
Identicon 2 of 5 Checksum? Fixed Length? Length in characters?					
Industrial 2 of 5 Checksum? Fixed Length? Length in characters?					

Bar Code/Options					
Interleaved 2 of 5  Checksum?  Fixed Length?  Length in characters?					
Matrix 2 of 5  Checksum?  Fixed Length?  Length in characters?					
MSI Plessey					
UPC - A  Require 2 or 5 Character Supplemental Code?					
UPC - E					
How many seconds do you want the bar code to be displayed on the handheld unit after a person scans it?					

**Step 9**

**Design the input masks.**

Application \_\_\_\_\_

1. \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

**Step 10**

**Design the lookup files.**

Application \_\_\_\_\_

1. \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_



**Lookup File Planning**

Parameter			
Look up type			
Look up file name			
Each record contains			
Key lookup field			

**Step 11** **Design the look and function of the application sub menu.**

Application \_\_\_\_\_

Sub Menu Operations	Default Title for Program	Enable	Title We Will Use
Enter data	Enter Data		
Edit data	Edit Data		
Verify data	Verify Data		
Transmit data via program	Xmit Data		
Delete data	Delete Data		
Transmit data via wedge	Xmit Wedge		

**Step 12** **Design the application looping pattern.**

Application \_\_\_\_\_

1. Loop from the last field back to the \_\_\_\_\_ field.

---



---



---



---



---

**Step 13** **Design the application branching pattern.**

Application \_\_\_\_\_

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Step 14** **Design the math registers.**

Application \_\_\_\_\_

register; named \_\_\_\_\_ insert after the \_\_\_\_\_

start with the value \_\_\_\_\_.

then \_\_\_\_\_ each time \_\_\_\_\_

Application \_\_\_\_\_

register; named \_\_\_\_\_ insert after the \_\_\_\_\_

start with the value \_\_\_\_\_.

then \_\_\_\_\_ each time \_\_\_\_\_

**Step 15****Decide the global operating options.**

Application \_\_\_\_\_

<b>Operating Parameter</b>	<b>Our Decision</b>	<b>Enable</b>
Key Combinations For Basic Operations Keys to quit the operation or program Keys to clear data from the Apex II Keys to increase contrast of the display Keys to decrease contrast of the display		
Remap Key Options		
Hardware Options Disable Scanner Between Scans Idle time in seconds before the Apex II automatically powers off Data Time Out (Communications Abort Time)		
Banner Information Banner Line 1 Banner Line 2 Banner Line 3 Banner Line 4		
Parameters Options Set the date and time on the handheld unit View the available memory in the Handheld Unit Parameters Menu Title Date/Time Menu Title Memory Menu Title		

<b>Operating Parameter</b>	<b>Our Decision</b>	<b>Enable</b>
General Options  On exiting a program, save changes in project file automatically without being prompted  When generating a program, save changes in the project file  Always overwrite the previous version when generating a program automatically without being prompted  When downloading, save the program in the handheld unit's FLASH  When downloading, set the handheld in Run Mode		

**Step 16                                      Decide whether you want to make a distribution diskette.**

**Application** \_\_\_\_\_

Yes\_\_\_\_\_ No\_\_\_\_\_