

## Using the TTBasic Flash Memory

### Introduction

Newer versions of the TTBasic terminals can support optional flash memory for storage of the application program and Ram Disk files. This application note will describe how the flash memory is configured and how to use it.

### Architectural Overview

The memory architecture of the newer TTBasic terminals remains the same as previous versions. The total RAM (512KB) memory space is divided into Standard (lower 64KB) memory and Extended (upper 448KB) memory. Standard memory is reserved for BIOS, the application program and program variables. The Extended memory is primarily reserved for Ram Disk (file) storage. To this memory system has been added a flash (512KB) memory which can be used to store the program and/or Ram Disk memory.

### New Commands

New commands have been added to the TTBasic interpreter to store and recall data between the RAM system memory and the flash memory. This provides a very simple method to manage the flash memory without a great deal of change to the TTBasic engine or the user application program. The SET 100 command has been added to support the flash memory functions. Unique parameters for the SET 100 command are used to define the individual functions used to manage the flash memory.

#### SET 100,0

This command is used to store the terminal console parameters to the flash memory. These parameters include values for the serial port (baud rate, parity, etc.), keypad options, and the bar-code/mag-stripe decoding options. These options are reloaded from the flash memory during terminal reset.

#### SET 100,1

This command will store the application program currently in the program memory to the flash memory. Once a program has been stored in flash memory it will be automatically reloaded to the program memory during terminal reset. If the **AUTO** command is used to define a starting address other than zero (0) then the program will be executed automatically after terminal reset. Normally, this command is used once after the application program has been loaded.

#### SET 100,2

This command is used to recall the application program from flash memory. If no valid program has been stored or if the stored program has a length of zero (0) then this command will trigger an error.

#### SET 100,3

This command will store the entire Ram Disk directory and all data files to the flash memory. All flash memory has a write tolerance which limits the number of times each memory sector may be written. Therefore, this function should be reserved for applications that require look up files or segmented programs. It is not well suited to data collection application where the data file must be saved every few seconds. Depending on the amount of data stored in the Ram Disk this function may take several seconds.

#### SET 100,4

This command is used to recall the previously stored Ram Disk directory and data files from the flash memory. The entire contents of the Ram Disk will be overwritten and previously stored data will be lost. This command can be included at the start of a program to automatically reload the Ram Disk memory. Depending on the amount of data that was stored this function may take several seconds.

## Memory Map

The TTBasic memory map shown in Figure #1 illustrates the relationship between the RAM memory and the Flash memory. Arrows between the memory blocks demonstrate how the SET 100 commands are used to transfer information between RAM and Flash memory.

