The Use of Forth in a Portable, Point of Sale Environment

Laslo Szasz
Robert Marshall
Anthony Kobykin
QuadROM Software Systems, Inc.
1000 St. Antoine St., W.
Montreal, Quebec H3C 3R7 Canada

Abstract

Our introduction to Forth has been dictated by the hardware, the Panasonic HHC. This machine is highly portable (only 2 pounds) and is widely used in the insurance environment to do sales illustrations of products.

We have had to implement strict project management structures to make our Forth environment productive, and to keep the lines of communication clear with our clients.

The hardware environment and the EPROMing methodology limits some of the well known advantages of Forth. Overall, we find Forth to be special because it allows for very small code and the easy access to hardware controls and the assembler.

We have developed many routines and structures to enhance the programming environment provided by Panasonic. Some of these are:

Input with Validation,
Non-checking Floating Point,
Assembly Routines,
Mathematical functions,
LCD special characters (French & Spanish characters),
Printer routines,
Program swapping,
Storage of Numeric Data - Space considerations,
Storage of Strings,
Multiple Module Systems.

Draft of the Presentation

On the Insurance Application

- Point of Sale environment
- Nature of insurance application
  - Form fill in with complex validation
  - Complex question sequencing
  - Format of output
  - Types of calculations done

On the Project Development Structure

- Full and Clear specifications
  - Take what is received from client and transform to our form
  - Single specification document to serve to be signed off by client and to give to programming

On use of Forth

- Hardware - describe dimensions, processor, and accessories
- Snapforth and Apple development system
  - host is a dialect of Forth 79;
  - can program host to do file manipulations;
  - host is merely used as editor and cross-compilor;
  - source code from editor is not directly interpretable on the Apple, so in order to test, a download must be done (several minutes);
  - test cycle is 10 to 20 min.;
  - Against the original design of Forth.;
  - Interpretability is wasted.
- Features of Snap
  - Short and Long Tags

When a program is compiled, word references generate a long (2 bytes) or short (1 byte) tag. The number of short tags available are limited (64) but their use speeds execution and reduces program size.
There is only one short tag table, but there can be multiple long tag tables. Long tags have a table number as their first byte.

Tag tables simplify the linkage of routines in multiple modules.

Floating Point

Floating Point numbers take 8 bytes, with 13 digits of precision. The HHC has built in many floating point words. (In fact, there are more floating point words than long integer ones, so we use floating point almost exclusively.) In the insurance application, the accuracy is required and the use of floating point takes away the scaling considerations on very wide ranging inputs and extended projections.

What we have done to improve on system processes.

Input with Validation

We have implemented arrow key control, line editing.
The validation requirements are quite severe in this application environment and we have developed the appropriate routines for numeric and alpha fields.

Non-checking Floating Point

Addition, multipl, subtr, and div. do not check for error conditions such as overflow, so use them only when routines are debugged. This is possible in the application because input strictly controlled, and after validation, only programming error can cause such conditions. Thus we obtain improvement in speed.

Assembly Routines

For conversion between integer and floating point and also for scaling for powers of 10.

Truncation.

Mathematical functions

Exponential and Logarithm routine.

LCD special characters (French & Spanish characters)
The nature and flexibility of the LCD
compared to other (e.g. Vmarc)
Capability of overlaying characters.
You can set up your own fonts.

Printer routines
We have not been able to do the same on the printer as on the LCD. We are currently in the development of this.

Ofcourse, we have developed the required formatting routines to obtain the full range of numeric formats for our illustration environment.

Program swapping
From a particular application system, using a hot key access is gained to the built in calculator of the HHC. This works simply applications systems, but complications arise in more complex systems and this extension is still in the development phase.

More General System Considerations

Storage of Numeric Data - Space considerations
We use Dbase-II for manipulation of rate tables and other numerical constants in order prepare them for download to an EPROM. We have certain compaction procedures to reduce the space taken. The overhead in time for decompression and the overhead in space for utility to keep track of information have to be considered.

Storage of Strings
We use a word table and there can be multiple phrase tables that point into the same word table. We had to do this with consideration of a bilingual application environment. Substantial space savings are achieved by the non-replication of words.

Multiple Module Systems
We have data modules and we can have data with program libraries on them. We have simplified the linkage of routines to other separately compiled program modules.