

A Guide to Forth on the Parallax Propeller™
for the complete beginner

HOWTO learn Forth for use on the Parallax Propeller™ from scratch – 2012

Author – G. Herzog, aka Loopy Byteloose at forums.parallax.com

Date – Nov 8th, 2012

Version 1.01 typo corrections, minor edit.

Version 1.00 Draft

Recommended resources for the complete beginner.

“Starting Forth” on line edition (2012), Forth, Inc. at www.forth.com

Gforth www.gnu.org/software/gforth/ or

SwiftForth www.forth.com or

any ANS Forth 94 compliant Forth for a conventional personal computer

or **pfth** on the Parallax Propeller with some lexicon restrictions.

Any Propeller platform and a choice of any actively in development version of Forth on the Parallax Propeller.

Currently known-active versions of Forth on the Propeller are as follows:

1. **pfth** – a C version of ANS Forth 94, compiles in GCC (can be studied), binary images
2. **PropForth** – .Spin file images
3. **TachyonForth** – .Spin file images and binary images

Introduction

There are likely many good ways to learn Forth in 2012, and the purpose of this document is NOT to include all of them. It is merely to provide one suggested path for learning the fundamentals of Forth quickly and to then move on to a rather specialized variety of Forth as adapted for Parallax's Propeller, an eight parallel microprocessor that is able to support a variety of Forth interpreters. At present there are three Forths available for the Parallax Propeller, but all have required some degree of adaptation to the Propeller's rather unique architecture and small 32Kbytes of hub RAM.

While a lot of well-intentioned people promptly suggest reading and studying "Starting Forth", they may not be fully aware that the two out-of-print versions were intended to support earlier standards of Forth - Forth 79 and Forth 83 respectively; both of which have been replaced by the ANS Forth 94 standard. It seems that the third and later Forth standard, ANS Forth 94, has never had a published hard copy version of "Starting Forth", but ANS Forth 94 has been provided with “Starting Forth” as a free on line HTML version by Forth, Inc. for quite some time. Forth, Inc. has been at the center of Forth development since the very early days and directly involved with the creator of Forth, Charles H. Moore.

So it may be best to begin with the most recent text and a version of Forth that follows that most recent standard and is closely followed by the early pioneers. The suggested starting point herein is the on

line edition of “Starting Forth” and any free copy of ANS Forth 94. Involvement with Forth 79 and Forth 83 may just provide unnecessary distractions.

How does one proceed?

For the complete beginner, "Starting Forth" is a strongly recommended entry point.

Trying to skip ahead to Leo Brodie's second book, "Thinking Forth" without first mastering "Starting Forth" is likely to confuse and confound the new user. I do admit there are other good first texts, but "Starting Forth" is very short and rather a painless way to begin that assures a good foundation in the basics.

Other Forth tutorials are generally longer and have more detail, which I feel can be better studied as a second, follow-up textbook. My own preference is to start out a new subject of study with the shortest authoritative text that I can find, then move on to longer ones to learn more details and different points of view. And in this case in particular, Leo Brodie did an excellent job with “Starting Forth”. Moreover, Forth, Inc. still provides it in the free on line edition and strongly recommends it all and any newcomers to Forth even though they publish a pair of useful references that can be purchased.

For those that already know Forth, learners come with many different levels of ability and this HOWTO is intended to help all. There are optional ways to get get up to speed.

If you have already learned Forth in the past, maybe a quick review of "Starting Forth" is all that is required. For that purpose a "Starting Forth Lexicon" has been created with a listing of Forth words presented in each chapter.

In this way, the learner can easily identify the words that are most vague in his or her memory and review that chapter. Forth, Inc. not only provides the text of “Starting Forth”, but answers to all the tasks at the end of each chapter. So for the learner that is unsure of the answers, there is added assistance. The “Starting Forth Lexicon” (see Appendix A below) is my own excerpted list of the Forth words that “Starting Forth” presents and likely remains tied to “Starting Forth” and Forth, Inc. in copyright ownership. Please respect that.

Besides, Forth, Inc. we have Parallax, Inc. www.parallax.com and their Propeller microcontroller to consider. This approach presumes that the new user may not know much of anything about the Propeller, SPIN, or PASM. At some point, some basic knowledge of these becomes important. Parallax provides a substantial amount of material that can be downloaded for free.

The earliest point in which to involve the Propeller is pretty much defined by whether or not you want to first study Forth and "Starting Forth" outside of the Propeller or on the Propeller. It is easiest to study the whole text of “Starting Forth”, on a generic Windows, Apple, or Linux PC because Chapter Three applies to having the support of an OS and file system. There are several good versions of Forth available as free public domain versions or demo versions of proprietary software. Later, when you learn Forth on the Propeller, you will likely learn how to work with Forth without an editor via several easy alternatives.

The author has used Gforth on an Ubuntu Linux machine, and Gforth can be gotten for Windows or Apple OSes as well. It is free via a GNU license. Forth, Inc. provides demo copies of SwiftForth that are supposed to be amongst the best of today's state of the art Forths. PolyForth is an older Forth created by Forth, Inc. that some people frequently mention in high regard. One can use one of these alternatives, but it is not necessary as Gforth works quite well with "Starting Forth". And then there are a multitude of other Forth versions available and it is beyond the scope of the author to catalog and evaluate all of them. Discussion here is limited to what the author considers to be closest to working well with the on line edition of "Starting Forth".

The key point is that these versions are ONLY intended to get you up to speed with mainstream Forth and at a later point in your studies you will have to learn the differences required by using the Propeller as a Forth platform as the adapted versions of Forth intended. The first and most obvious difference is that lack of an OS and a file system to support the editor and file storage. The most obvious second is that you have eight Forth stack machines available to run in parallel at the same time.

This educational approach is intended to give you an brief overview of what the general Forth model is on today's personal computers and then to work into the differences that the Propeller creates due to being a microcontroller with limited RAM and due to having eight processors capable of operating in parallel.

As the learner progresses into Forth on Propeller, more and more study specific to the Propeller will become required - but these studies will also be quite rewarding as a great deal will be learned in a way that useful and transferable to any project in any language on the Propeller.

In other words, by studying Forth, one will have a deeper appreciation of PASM, Spin, and the hardware and will be able to progressively learn to do more with the Propeller. Forth is not a 'beginner's language' in the sense that Basic might be considered. One need not eventually abandon it for another more sophisticate language. Forth can be used to actually research questions of code and behavior that can be applied to a project in another language or Forth can evolve with the user's improved knowledge.

The suggested sequence of study.

A. If you have not already done so, first learn Forth through "Starting Forth" on a conventional personal computer or in some cases using a particular version of Forth on Propeller that supports the "Starting Forth" vocabulary.

At this time, **pfth** has created an ANS Forth 94 version of Forth to run on the Propeller, but it lacks the features that Chapter Three of "Starting Forth" mentions, the ?STACK word in Chapter Four, and the BLK word that is mentioned in Chapter Nine. One can merely omit study of Chapter Three and BLK if one chooses to learn on a Propeller with **pfth**. ?STACK may not be of critical importance.

The other versions, **PropForth** and **TachyonForth** may at any time decide to provide a special "Starting Forth" version or documents to help the new user identify differences. At this time, they are unsure that they will do so. Since development in all is evolving at a rather fast pace, this HOWTO assumes that these items are not yet available and may never become reality.

Resources - "Starting Forth" on-line edition, 2012; Forth, Inc.

And one ANS Forth 94 compliant version of Forth. But note that the author has specifically used Gforth and is not fully aware of how suitable other versions might be. Forth, Inc. has a demo copy of SwiftForth that is likely to be the most compliant as they own the copyright to "Starting Forth" and have republished the updated on line edition for many years.

Tasks – Study "Starting Forth" from beginning to end and work through all examples and the problems at the end of each chapter until satisfied that you have a good grasp of what it does and how it works.

B. Review "Starting Forth" in comparison with your chosen target Forth on Propeller for which words are included, omitted, or renamed.

Resources – The "Starting Forth Lexicon" - a chapter by chapter listing of all the Forth words presented in "Starting Forth" as of the 2012 on line edition. (included herein as Appendix A, below)

Tasks - Prepare an up-to-date comparison of the actual lexicon installed in your Forth on Propeller to the lexicon that you learned in "Starting Forth" to introduce you to what is available, what has been omitted, and what has acquired a different name.

You will need a word processor or spreadsheet application to do your work. Open a working file.

First, using the "Starting Forth Lexicon" as a starting point, list all the words in that lexicon in any sequence that you prefer in the first left hand column of a document or spreadsheet. This can be done by Chapter, by use, or merely by ASCII sort.

Second, acquire a complete list of the lexicon installed on you new Forth on Propeller to verify what is available.

Third, identify all the "Starting Forth Lexicon" words that appear to be missing in the second column of your document or worksheet.

Fourth, try to verify whether the omitted words are actually gone or have been renamed. When a renamed word is located, remove the note from the omitted list and enter the new name in the third column.

At this point, you have a clear picture of what is available and what is not. It is up to you to decide if you can now live without certain words or if you prefer to recreate any or all of the missing items.

~~~~~ | ~~~~~

It should be noted at this point, you are beginning to take responsibility for your own Forth dictionary as an important step towards mastering Forth. In general, everyone has different opinions of what they must have and what they can do without. And with limited space, the choices of what to do without are pretty much defined by the personal projects the user has in mind.

## C. Learning about the oddities and features of Forth on a micro-controller

### ***Introductory Discussion***

Originally Forth was a typical computer language that used a dumb terminal for all interactive i/o, a printer for output of results in hard copy, and a mainframe to store images of the ever evolving Forth dictionary.

But early on and in the background, there were also many instances where actual specialized robotic control of telescopes, sensors, networks, and process machinery did take place.

These unique aspects of I/O have always required some customization to the idealized standard core of Forth lexicon. Documentation of such aspects has never been a strong point of introducing Forth to the new learner. This is the turning point, where the user must accept that his version of Forth will evolve into a more specialized platform specific language and a function specific language. In other words, the 'honeymoon is over'. One needs to think as much about the architecture of the Propeller and of any devices attached to the Propeller as they do about the Forth language. The two need to be mated in a way that they serve you, the Forth programmer.

And for Forth on the Propeller, the customized and non-standard dictionary also offers the most interesting and challenging aspects of learning Forth. So try to enjoy going deeper and to acquire more knowledge of SPIN and PASM along the way. These non-standard dictionary schemes may also be the most rewarding in learning what the Propeller can do with eight processors available in parallel. The fact that Forth allows you to write a little and verify before you go on to the next item is in many ways more enjoyable than reading about libraries, compiling, loading, testing and debugging.

### ***Resources -***

*For hardware*, all of the Parallax Propeller hardware documentation and any documentation for attached hardware.

*For software*, documentation for all the IDEs and the compilers of .spin and binary images -- that you prefer to work with -- in order to load images of your version of Forth on the Propeller.

And of course, *your actual version of Forth on Propeller* as supported (which varies with each developer's stage of development, desire to document, and available time to document).

***Tasks*** - Read and work through any introductory information offered and apply what you have already learned about Forth to resolve non-documented features.

## D. Learning how edit and retain your own work of Forth on a micro-controller

***Resources*** - Your acquire knowledge of how your particular version of Forth on Propeller retains and modifies an image.

In some instances, you may be able to add to the core dictionary with a 'cut and paste' procedure in the Console. In other instances, you might be able to actually, cut and paste from your own additions to the

original code image and have the whole retained in EEPROM that will come up each time you reboot.

At this point, you are pretty much expected to acquire the skills and habits that make Forth housekeeping work for you. And you may also find that the version of your Forth evolving toward new means to storage and revise the dictionary space.

**Tasks** - Participate and keep up with the version of Forth that you prefer in all aspects of interest and importance to you. (Isn't this where you wanted to get to?)

### **In closing**

This document was created in a style where it would hopefully be useful for the longest possible period of time.

Many HOWTO documents are quickly dated and useless due to changes in available resources and links. In order to remain useful, the new user is asked to carry most of the burden to get up to a level that the Forth on Propeller versions provide. None of the Forth on Propeller versions can afford to dedicate a great deal of time and effort to teach the basics to a complete beginner, and the resources are already well-established to do so. So the bulk of the effort is to bring the new user up to speed and to ease the transition.

Every attempt is made to make the process as short and as simple as possible. Good luck and enjoy Forth on the Propeller.

**Appendix A** – The “Starting Forth Lexicon” as presented in the on line edition of Leo Brodie's “Starting Forth” at Forth, Inc. site [www.forth.com](http://www.forth.com) November 2012.

## Introduction

### Words

Nothing in the Introduction

## Chapter One

### Words

: xxx yyy ;  
CR  
SPACE  
SPACES  
EMIT  
." xxx"  
+  
.

## Chapter Two

### Words

+  
-  
\*  
/  
/MOD  
MOD  
SWAP  
DUP  
OVER  
ROT  
DROP  
2SWAP  
2DUP  
2OVER  
2DROP

## Chapter Three

### Words

USE xxx  
LIST  
LOAD  
( xxx)  
UPDATE  
EMPTY-BUFFERS  
BLOCK  
INCLUDE xxx  
FORGET xxx  
MARKER xxx

## Chapter Four

### Words

IF xxx  
ELSE yyy  
THEN zzz  
=  
-  
<  
>  
0=  
0<  
0<  
0>  
AND  
OR  
?DUP  
ABORT" xxx"  
?STACK

### Additional Words

INVERT  
TRUE  
FALSE

## Chapter Five

### Words

1+  
1-  
2+  
2-  
2\*  
2/  
ABS  
NEGATE  
MIN  
MAX  
>R  
R>  
I  
R@  
J  
\*/  
\*/MOD

## Chapter Six

### Words

DO ... LOOP  
DO ... LOOP+  
LEAVE  
BEGIN ... UNTIL  
BEGIN xxx  
WHILE yyy  
REPEAT  
U.R  
PAGE  
QUIT

## Chapter Seven

### Words

U.  
UM\*  
UM/MOD  
U<

HEX  
OCTAL  
DECIMAL

<#  
#  
#S  
c HOLD  
SIGN  
#>

<# ... #>  
<# ... ROT SIGN #>

D+  
D-  
DMAX  
DMIN

D=  
D<  
DU<  
D.R

M+  
SM/REM  
FM/MOD  
M\*  
M\*/

## Chapter Eight

### Words

CONSTANT xxx  
VARIABLE xxx  
CREATE xxx  
!  
@  
?  
+!  
ALLOT  
'  
C!  
C@  
FILL  
BASE  
2CONSTANT xxx  
2VARIABLE xxx

2!  
2@  
C,  
DUMP  
ERASE

## Chapter Nine

### Words

' xxx  
[]  
EXECUTE  
EXIT  
QUIT  
HERE  
PAD  
SCR  
BASE  
SP@  
TIB  
#TIB  
SP0  
>IN  
BLK

## Chapter Ten

### Words

TYPE  
-TRAILING  
MOVE  
CMOVE  
KEY  
ACCEPT  
WORD  
>NUMBER  
COUNT  
CMOVE>  
COMPARE  
BLANK

## Chapter Eleven

### Words

DOES>  
IMMEDIATE  
POSTPONE xxx  
LITERAL  
[  
]

## Chapter Twelve

### Words

No additional core words

3 study examples