

TinyC for the SD-8516

Introduction

This TinyC is more like the “B” language in it's feature set, and in the idea that it is intended to compile a version of itself written in C.

Current Feature Set:

- Expressions with precedence
- Local variables
- if/else/else if
- Comparison operators
- while
- break/continue
- Multiple functions + function calls
- Function arguments
- Pointers (read, write, address-of)
- Array indexing
- char* (8-bit access)
- String literals
- Hex literals
- Character literals
- Comments (//)
- Global variables
- Builtins (putchar, getchar, halt, yield)

This is enough that I was able to start writing functions in C like abs, min, max, strlen, strcmp, atoi/itoa, and so on.

History

A self-hosting C compiler had been my goal for quite some time. But I didn't know how to write one. I hacked out a BASIC first, then followed some guides and source code to get a Forth running. But C eluded me for several months. Eventually I was able to leverage what I had learned doing BASIC to get a parser that could compile int main(void) { return 0; } and emitted LDA #0 and RET. The debugger was simply printing the return value. The rest of the story is found in the [TinyC Developer Diary](#). Mainly so I don't forget what I learned, as this project has gotten quite big.

From:
<https://www.appledog.ca/wiki/> - **Appledog**

Permanent link:
https://www.appledog.ca/wiki/doku.php?id=sd:tinyc_for_the_sd-8516&rev=1776434660

Last update: **2026/04/17 14:04**



