

LLVM Backend for SD-8516

February to April 2026

I started by copying the Lanai files and renaming everything to SD8516. When it compiled I moved forward.

Yes, this took three months.

April and May 2026

During this time I worked on the back-end. There are many tutorials on the 'net such as the CPU0 tutorial. I chose LLVM because I heard the documentation was more useful. Thank God, because I was going to need everything I could get. It's impossible to describe this process. I should have kept more notes on the wiki about it. I may have been able to save myself several days worth of grepping to find random stuff in the target's source tree. This period I call the insanity days. 4 and 8 hour days of doing nothing but trying to lower LLVM code into SDA assembly.

This period is binary. It compiles or it does not. There is no incremental improvement here. I learned a lot and actually changed my ISA a bit because of what I learned doing this. But, I do not remember anything I did here. It is a black hole. I've probably blocked it out of my memory. It was that horrifying.

Notes from April/May

- That stupid string that looks like e-p:32:8-i16:8-i32:8-i64:8-n16 is in:
 - `llvm/lib/TargetParser/TargetDataLayout.cpp`

Total cost of this note alone: Probably 6 hours.

- COMMON files I need to find:
 - `llvm/lib/Target/SD8516/SD8516InstrFormats.`
 - `llvm/lib/Target/SD8516/SD8516InstrInfo.`
 - `llvm/lib/Target/SD8516/SD8516ISelLowering.`

Common advice:

- Look very carefully at the error message. 90% of the time you're missing a header, a function declaration, or something's type is wrong. The solution is often easier than you think.
- Use git or you will mess stuff up and be unable to go back.

June 2026

A sudden thrust of energy, cola and Doritos and the back-end was in sight. A lot of this was fueled by

excitement over many small wins. Such as getting something to work. Once I saw it start to emit SDA assembly I was hooked. Many small changes to the assembler were made too, to support LLVM stuff that I didn't want to try and lower out. One example of an extreme change I made was to remove MOV touching the zero flag because I couldn't figure out how to set it in the files. Its supposed to be [SR] (status register). But there are multiple places you're supposed to add it and I started hallucinating that I had added it in certain places then when I went back to check it was gone. Whatever, I'll just remove the flags from MOV! Now i'm done with that part. HAHA. Hehe hoho.

Suddenly I was out of the tunnel and I didn't even realize it. I decided to skip some stuff that would have made the generated code "better". Like jumptables and some function calls (that are instead inlined, which bloats code).

- Remember: jumptables for large switches
- Remember: mul and div calling functions versus inlining
- [Notes to Self from June](#)
- [Runtime Library Notes](#)

From:

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

Permanent link:

https://www.appledog.ca/wiki/doku.php?id=sd:llvm_backend_for_sd-8516&rev=1781146436Last update: **2026/06/11 02:53**