Native compiler for different platforms

Building native compiler with necessary C libs could be tricky. Firstly there must be crosscompiled temporary compiler for compiling its native version as second stage compiler. This must be done from PC, which possess different machine code, than target-native machine code (for example some small microcontroller with narrow ISA set). To make it useful, one need hexdump, asembler, linker, and many other tools, including some C library newlib. Described tutorial enables trivial building toolchains for platforms:
-ARM Cortex M0+; floating operations in software;
-ARM Cortex4F; floating operations in hardware;
-Intel Pentium (i586) 32bit;
-MIPS;
-AVR many MCU's; with avr-libc
Just analyze and run shell script:

#./build_nativecompiler.sh