Subject: Re: How to link assembler compiled file
Posted by mirek on Thu, 22 May 2025 07:59:24 GMT
View Forum Message <> Reply to Message

frederik.dumarey wrote on Sun, 18 May 2025 11:51Hello,

I added a assembler file named my_strlen.S to my package, the contents of this file are below:

```
 .global _my_strlen   # Export the symbol _my_strlen
 .text      # code section

_my_strlen:

 #function prologue
 pushq %rbp
 movq %rsp, %rbp
 movq %rdi, %rsi    # copy the string pointer to %rsi
 xorq %rax, %rax    # zero %rax to use as a counter

strlen_loop:
 cmpb $0, (%rsi)    # compare byte at %rsi with 0 (null terminator)
 je strlen_end    # if zero, end of string reached
 incq %rax     # increment counter
 incq %rsi     # move to next character
 jmp strlen_loop    # repeat loop

strlen_end:

 #function epilogue
 popq %rbp
 ret
```

Since this is AT&T assembler code, it compiled without any problem to a object file using right_click, build and compile option in TheIDE GUI. So far so good.

I then added a C++ file, which has the following content:

```
#include <iostream>

//declare the assembler function
extern "C" size_t my_strlen(const char* str);

int main(int argc, const char *argv[])
{
 const char* message = "Hello from Assembler";
 size_t length = my_strlen(message);
 std::cout << "Message: " << message << std::endl;
 std::cout << "Length: " << length << std::endl;
 return 0;
}
```

which can also be compiled using the same method as stated before.

Now my question: how do I link those two object files in the GUI? I suppose I have to go to Project menu item, then Custom build steps, but what do I enter in all these fields?

Thanks.

Well, this is very exotic issue so it made me check the code and interestingly gcc builder (used with clang) simply treats .s files just like any other source (.c, .mm), including adding the .o to the linker. To be sure, you can check commandlines with Verbose.