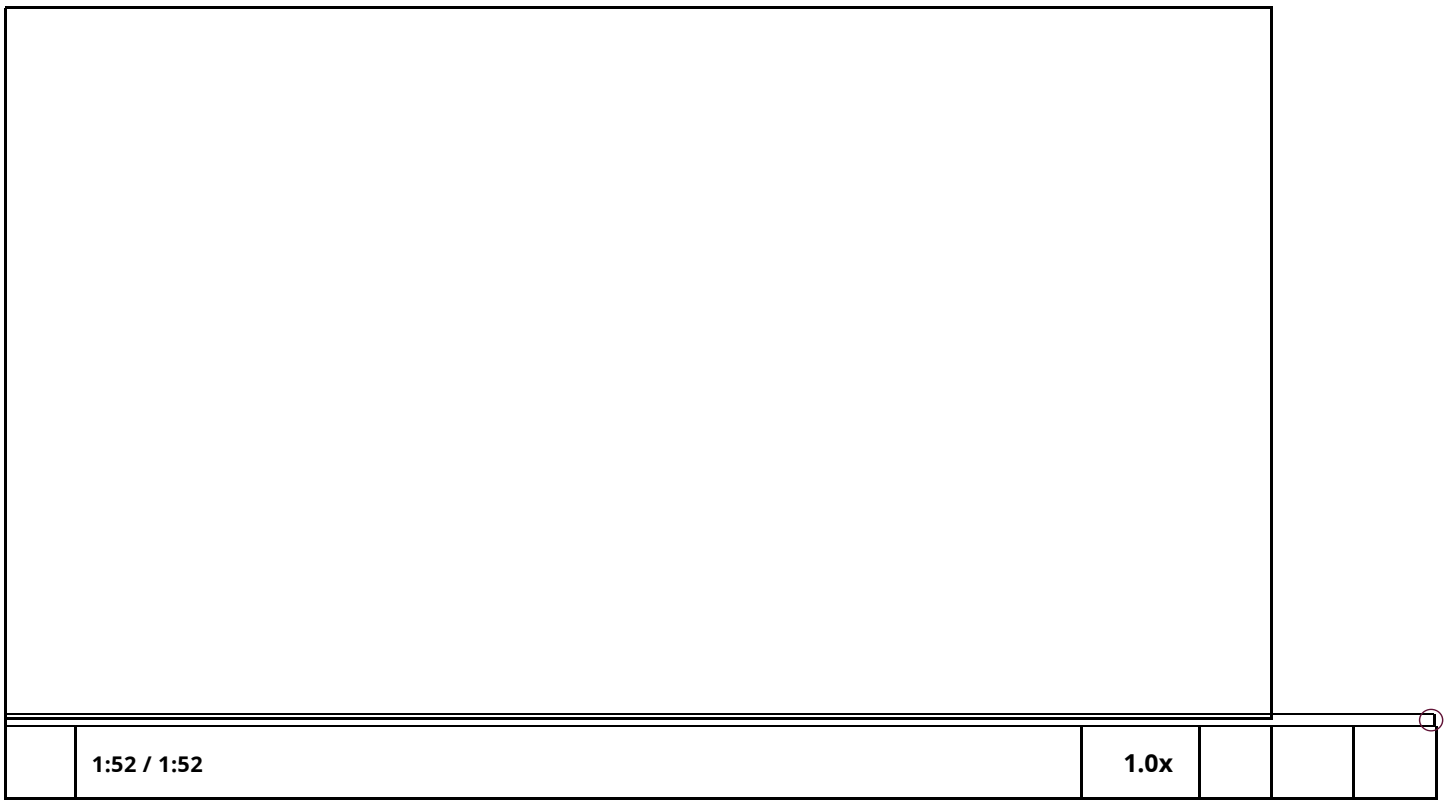
In this module, we will learn about assembly language, a human-readable form of machine code. It provides us, as programmers, an easier way to understand and talk about programs instead of looking at binary 1s and 0s. We will begin with the straightforward mapping of opcodes to instructions and introduce some features of assembly language: directives, TRAP codes, labels, and comments. Then we will describe how an assembly language program is translated by an assembler into machine code. Finally, we will briefly talk about how the program is loaded into memory for execution.

At the end of his module, students will be able to:

- Apply assembly language, directives, TRAP codes, labels, and comments.
- Describe how an assembler translates an assembly program into machine code.
- Describe how a linker can tie together multiple compiled sections of a program into one block of code.
- Describe the role a loader plays in running a program.

INTRODUCING ASSEMBLY



Download transcript .txt

Show Discussion

New Post

Help

EdX offers interactive online classes and MOOCs from the world's best universities. Online courses from MITx, HarvardX, BerkeleyX, UTx and many other universities. Topics include biology, business, chemistry, computer science, economics, finance, electronics, engineering, food and nutrition, history, humanities, law, literature, math, medicine, music, philosophy, physics, science, statistics and more. EdX is a non-profit online initiative created by founding partners Harvard and MIT.

Terms of Service and Honor Code

Privacy Policy (Revised 10/22/2014)

POWERED BY OPENedX

About edX

About

News

Contact

FAQ

edX Blog

Donate to edX

Jobs at edX

Follow Us

Facebook

Twitter

LinkedIn

Google+

Tumblr

Meetup

Reddit

Youtube