IntelliJ IDEA Setup for Chisel

Xiaokun Yang

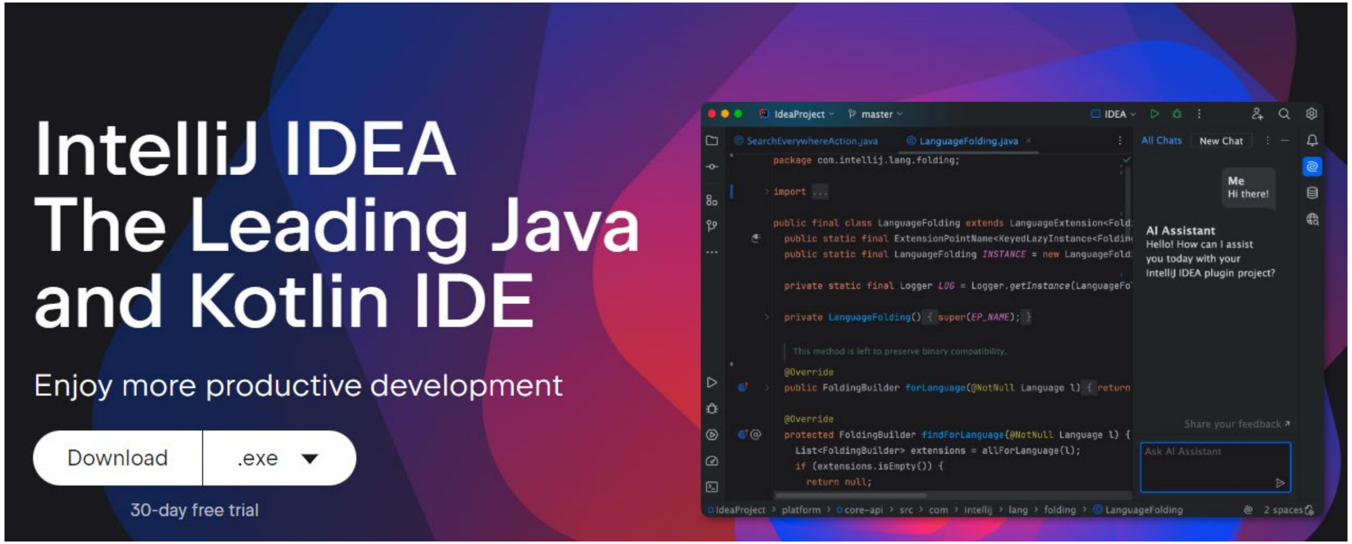
9/23/2024

BERKELEY LAB

University of Houston Clear Lake

1. Install IntelliJ

Please download and install IntelliJ
Community edition is free-licensed version
Or apply for student license which is also free


IntelliJ IDEA
The Leading Java and Kotlin IDE
Enjoy more productive development
Download
.exe
30-day free trial
IdeaProject
master
IDEA
SearchEverywhereAction.java
LanguageFolding.java
All Chats
New Chat
Share your feedback
Ask AI Assistant
IdeaProject › platform › core-api › src › com › intellij › lang › folding › LanguageFolding
2 spaces

```
package com.intellij.lang.folding;

import ...

public final class LanguageFolding extends LanguageExtension<Fold
  public static final ExtensionPointName<KeyedLazyInstance<Foldin
  public static final LanguageFolding INSTANCE = new LanguageFold

  private static final Logger LOG = Logger.getInstance(LanguageFo

  private LanguageFolding() { super(EP_NAME); }

   This method is left to preserve binary compatibility.
  @Override
  public FoldingBuilder forLanguage(@NotNull Language l) { return

  @Override
  protected FoldingBuilder findForLanguage(@NotNull Language l) {
    List<FoldingBuilder> extensions = allForLanguage(l);
    if (extensions.isEmpty()) {
      return null;
```

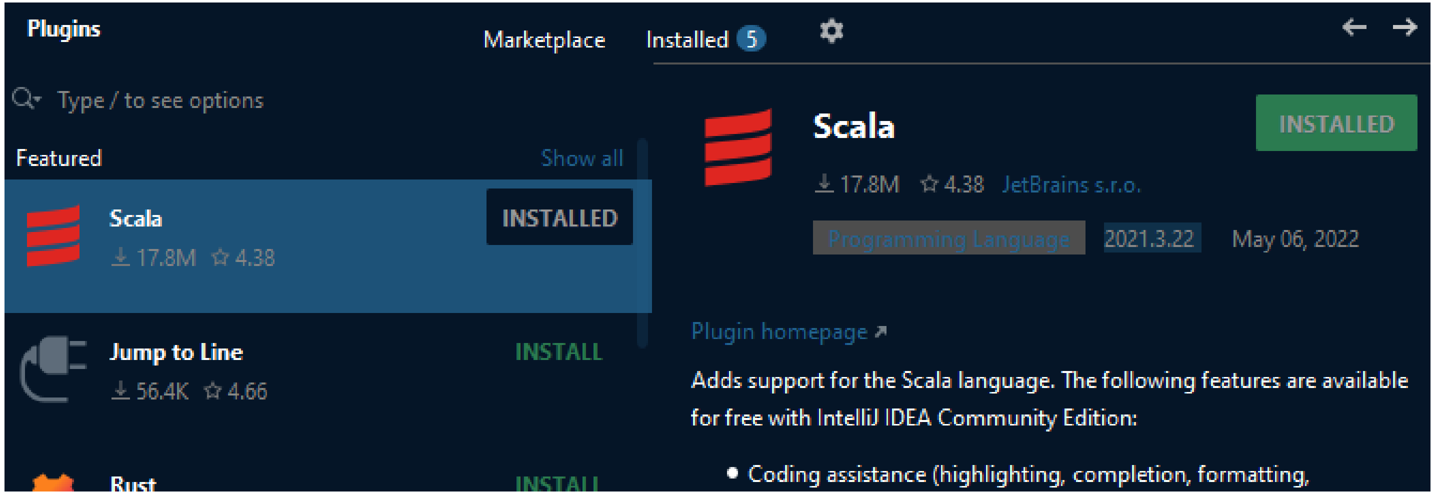
Me
Hi there!

AI Assistant
Hello! How can I assist you today with your IntelliJ IDEA plugin project?

2. Install Scala Plugin

Chisel is based on **Scala**, so you'll need to install the **Scala plugin** in **IntelliJ** to work with Chisel code.

File -> setting -> plugin -> **Marketplace**



Adds support for the Scala language. The following features are available for free with IntelliJ IDEA Community Edition:

- Coding assistance (highlighting, completion, formatting,

3 JDK (Java Development Kit) for SBT (Scala Build Tool)

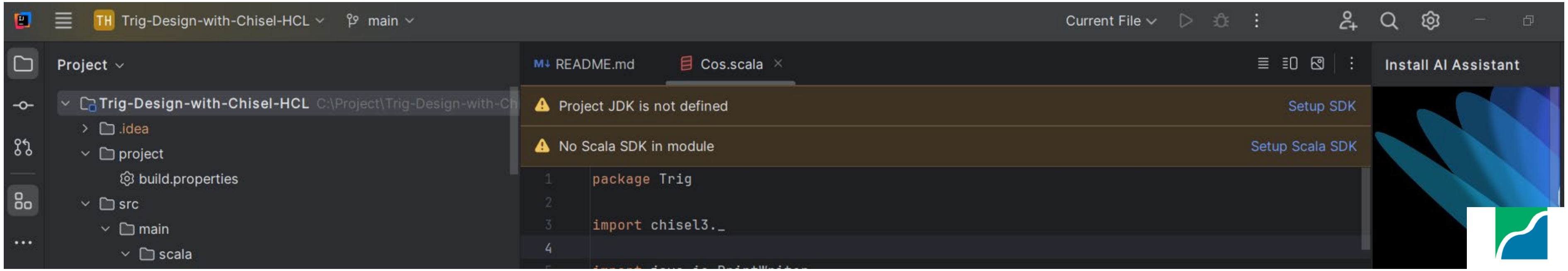
3.1 Install JDK (Java Development Kit)

To set up Chisel in **IntelliJ**, we use **sbt (Scala Build Tool)**, and sbt requires the **JDK** to function correctly. Chisel is based on Scala, and Scala, in turn, runs on the **Java Virtual Machine (JVM)**.

3.2 Configure **IntelliJ**:

After installing the **JDK**, you'll need to configure it in **IntelliJ** by going to:

•File -> Project Structure -> Project Settings -> Project -> Select the JDK under "Project SDK."



Project JDK is not defined

No Scala SDK in module

```
package Trig

import chisel3._

```

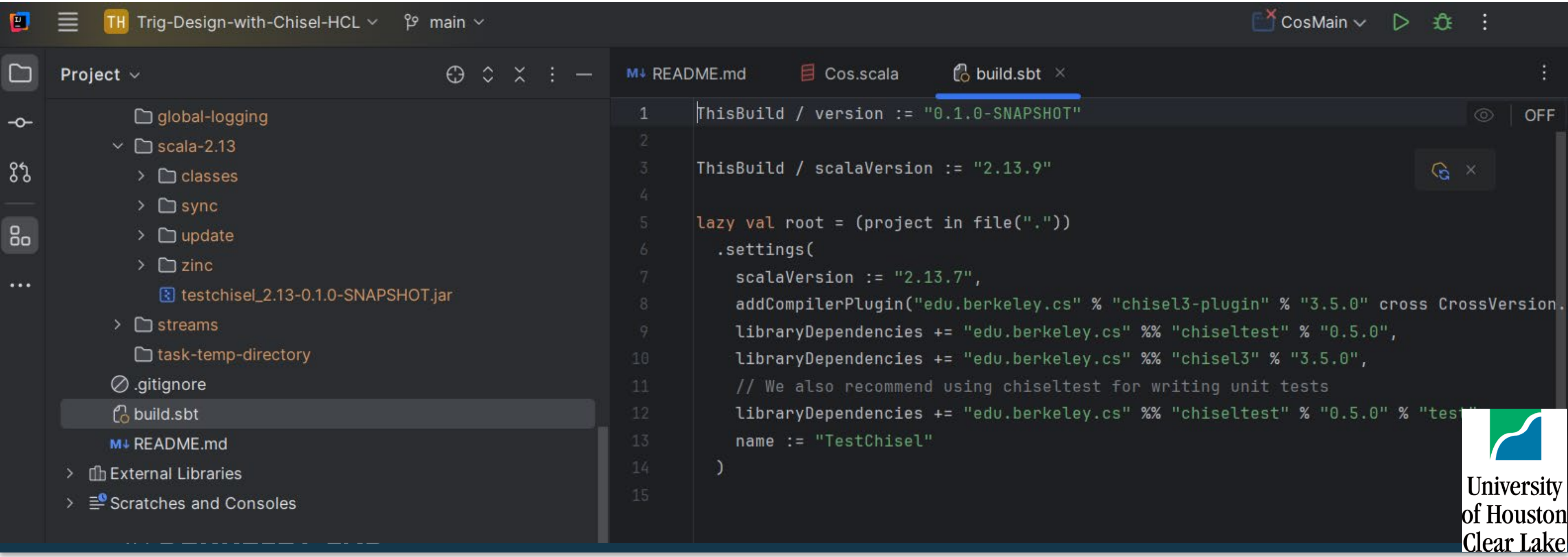
BERKELEY LAB

University of Houston Clear Lake

4. SBT (Scala Build Tool)

Chisel projects are typically managed using **sbt**, which simplifies building Scala (and Chisel) projects.

Before running sbt, please create/check the Chisel dependencies in your build.sbt file which is shown in the screenshot. You can skip this step by clone the Git Repository in step 5.



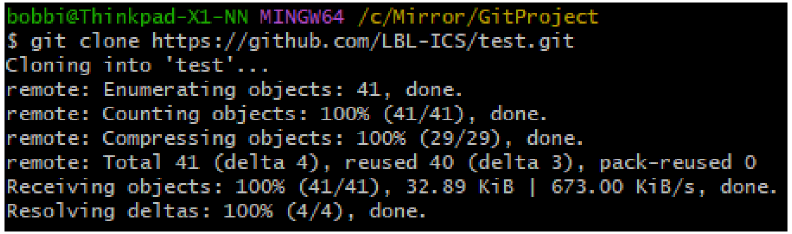
```
ThisBuild / version := "0.1.0-SNAPSHOT"

ThisBuild / scalaVersion := "2.13.9"

lazy val root = (project in file("."))
  .settings(
    scalaVersion := "2.13.7",
    addCompilerPlugin("edu.berkeley.cs" % "chisel3-plugin" % "3.5.0" cross CrossVersion.
    libraryDependencies += "edu.berkeley.cs" %% "chiseltest" % "0.5.0",
    libraryDependencies += "edu.berkeley.cs" %% "chisel3" % "3.5.0",
    // We also recommend using chiseltest for writing unit tests
    libraryDependencies += "edu.berkeley.cs" %% "chiseltest" % "0.5.0" % "tes
    name := "TestChisel"
  )
```

University of Houston Clear Lake

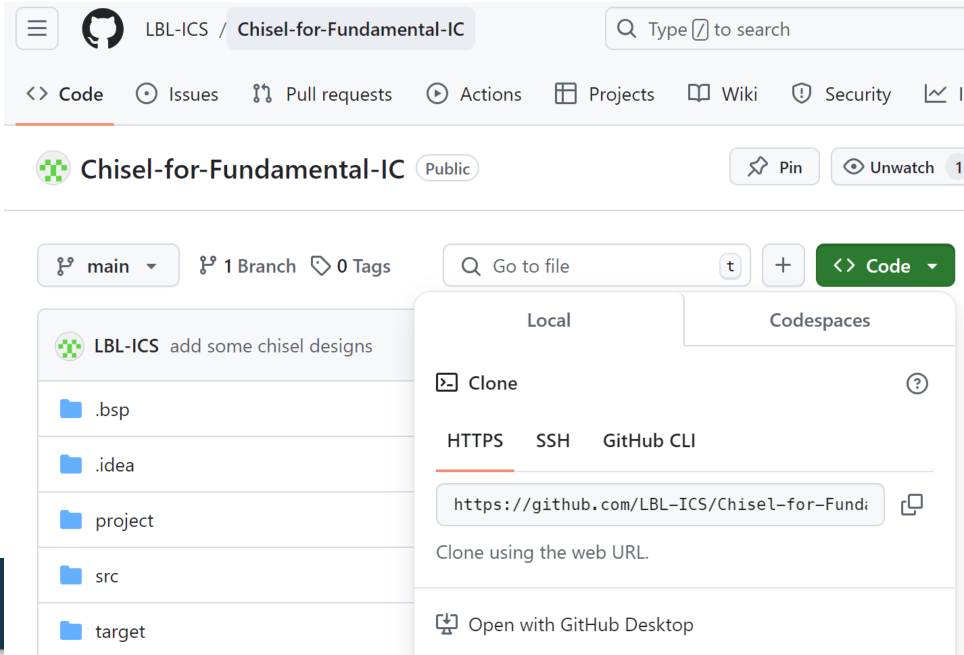
5. Clone Git Repository for Chisel Env and Project

Please clone the Git repository: https://github.com/LBL-ICS/Chisel-for-Fundamental-IC.git

1) Locate to your local directory and open Gitbash
2) Typing: git clone https://github.com/LBL-ICS/Chisel-for-Fundamental-IC.git



```
bobbi@Thinkpad-X1-NN MINGW64 /c/Mirror/GitProject
$ git clone https://github.com/LBL-ICS/test.git
Cloning into 'test'...
remote: Enumerating objects: 41, done.
remote: Counting objects: 100% (41/41), done.
remote: Compressing objects: 100% (29/29), done.
remote: Total 41 (delta 4), reused 40 (delta 3), pack-reused 0
Receiving objects: 100% (41/41), 32.89 KiB | 673.00 KiB/s, done.
Resolving deltas: 100% (4/4), done.
```



LBL-ICS / Chisel-for-Fundamental-IC

Chisel-for-Fundamental-IC Public

LBL-ICS add some chisel designs

.bsp
.idea
project
src
target

Clone

HTTPS SSH GitHub CLI

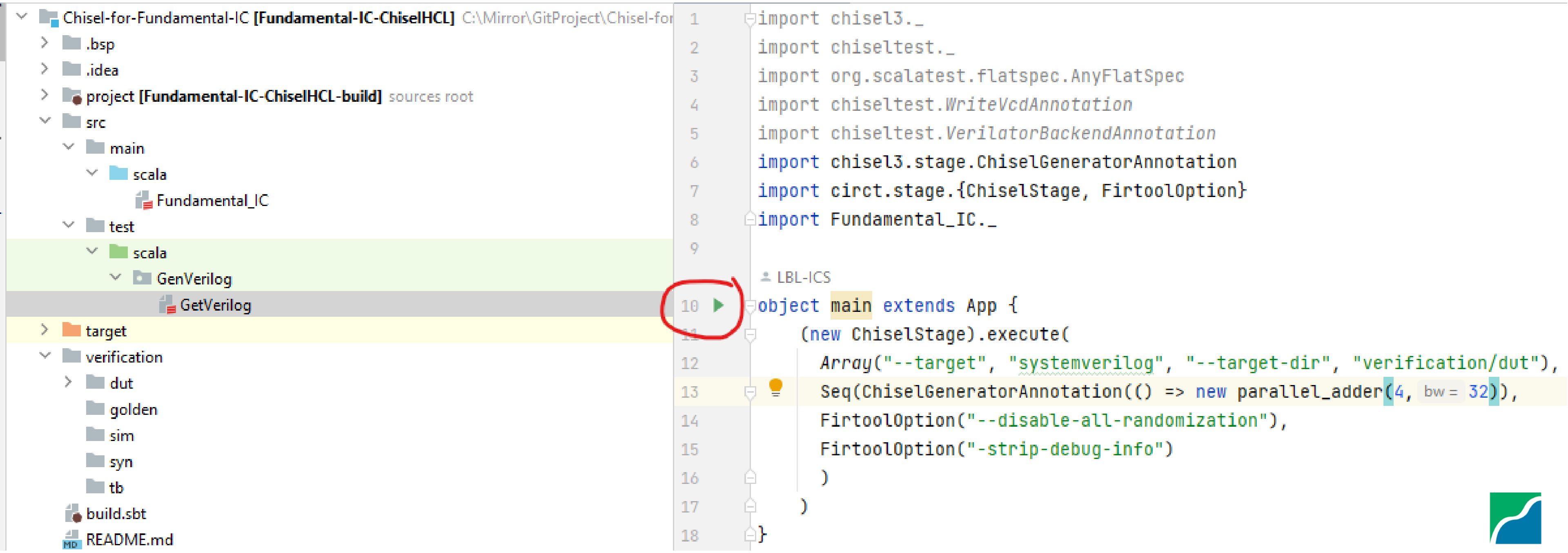
`https://github.com/LBL-ICS/Chisel-for-Fund`

Clone using the web URL.

6. Run Chisel Project

After setting up sbt, you can compile and run your Chisel project by clicking the green triangle!

When successfully run the sbt, you will find the generated Verilog code in verification/dut



```
Chisel-for-Fundamental-IC [Fundamental-IC-ChiselHCL] C:\Mirror\GitProject\Chisel-for
  .bsp
  .idea
  project [Fundamental-IC-ChiselHCL-build] sources root
  src
    main
      scala
        Fundamental_IC
    test
      scala
        GenVerilog
          GetVerilog
  target
  verification
    dut
    golden
    sim
    syn
    tb
  build.sbt
  README.md
```

```scala
import chisel3._
import chiseltest._
import org.scalatest.flatspec.AnyFlatSpec
import chiseltest.WriteVcdAnnotation
import chiseltest.VerilatorBackendAnnotation
import chisel3.stage.ChiselGeneratorAnnotation
import circt.stage.{ChiselStage, FirtoolOption}
import Fundamental_IC._

LBL-ICS
object main extends App {
  (new ChiselStage).execute(
    Array("--target", "systemverilog", "--target-dir", "verification/dut"),
    Seq(ChiselGeneratorAnnotation(() => new parallel_adder(4, bw = 32)),
    FirtoolOption("--disable-all-randomization"),
    FirtoolOption("-strip-debug-info")
    )
  )
}
```

References

1. Chisel Design – XY. Pdf
2. Chisel and IC Projects Structuring – XY. pdf
3. Chisel Bootcamp
4. Chisel3 Cheat sheet
5. Chisel textbook, version 5
6. Chisel Designs for Fundamental IC, Git repository: https://github.com/LBL-ICS/Chisel-for-Fundamental-IC.git
7. Xiaokun Yang, "Integrated Circuit Design: IC Design Flow and Project-Based Learning", CRC Press -Taylor & Francis Group, ISBN: 978-1-032-03079-1 (ebook ISBN: 978-1-003-18708-0), First edition
8. Project Structuring for IC Design and Simulation, Git repository: https://github.com/LBL-ICS/IC-Design.git