Here's a streamlined guide for launching Quartus and setting up a System on Chip (SoC) project, which includes licensing, creating a new project, adding Verilog code, and compiling.

1. Launch Quartus

- **Open Quartus**: Launch the Quartus software from your desktop or applications menu.

2. License Quartus (if required)

- **License Setup**: If you are in a university lab, ensure that Quartus is licensed. Typically, the university IT department will have set this up. If not:
- **Open License Setup**: Go to `Tools` > `License Setup`.
- **Enter License Info**: Follow the instructions to enter your license information.

3. Start a New Project

- **Create New Project**:
  - Go to `File` > `New Project Wizard`.
  - Click `Next`.

4. Project Name and Location

- Project Name: Enter a name for your project.
- Project Location: Choose the directory where you want to save your project files.
- Top-Level Entity: Enter the name of your top-level Verilog module.
  - Click `Next`.

5. Add Files

- Add Existing Files: If you have existing Verilog files, add them here. If not, you can skip this step.
  - Click `Next`.

6. Select Correct FPGA

- Select Device: Choose the FPGA device you are targeting.
  - Family: Select the appropriate FPGA family (e.g., Cyclone, Arria).
  - Device: Select the specific FPGA device.
  - Click `Next`.

7. EDA Tool Settings

- Tool Settings: Configure any EDA tools you are using (optional).
  - Click `Next`.
  - Click `Finish`.

8. Add Verilog Code

- Create Verilog File:
- Go to `File` > `New`.
- Select `Verilog HDL File` and click `OK`.
- Write or paste your Verilog code.
- Save the file with a `.v` extension in your project directory.

9. Compile the Project

- Compile:
  - Go to `Processing` > `Start Compilation`.
  - Wait for the compilation to complete. Fix any errors that may arise.

10. Assign Pins

- Pin Planner:
  - Go to `Assignments` > `Pin Planner`.
  - Assign the correct pins to your FPGA according to your board's specifications.
  - Save your pin assignments.
- Compile Again:
  - Re-compile the project to ensure that the pin assignments are correct.
  - Go to `Processing` > `Start Compilation`.

11. System on Chip (SoC) Configuration

- SoC Integration:
  - Depending on your SoC requirements, you might need to configure additional IP cores and system settings.
  - Use the Platform Designer (Qsys) for integrating various components.

Additional Tips

- Simulation: Use ModelSim or any other simulator integrated with Quartus for verifying your Verilog code.
- Debugging: Use SignalTap Logic Analyzer for in-system debugging.
- Documentation: Refer to the Quartus user manual and FPGA board documentation for detailed steps and pin configuration.

This guide provides a basic workflow for setting up and compiling a Quartus project. Depending on your specific requirements, additional steps may be necessary.