

```
graph TD
    subgraph Compiler
        SC[Source Code] --> P[Parser]
        P --> AST[Abstract Syntax Tree]
        AST --> SSR[Syntax Sugar Removal]
        SSR --> TI[Type Inference]
        TI --> TE[Type Environment]
        SSR --> CSSA[Conversion to SSA]
        TE --> CSSA
        CSSA --> IR[Intermediate Representation]
        IR --> CC[Closure Conversion]
        CC --> ASFF[Analyse Stateful Functions]
        ASFF --> ST[State Tree]
        ASFF --> CG[Code Generation]
        ST --> CG
        CG --> LLVMIR[LLVM IR]
    end
```

The diagram illustrates the Runtime architecture, showing the interaction between the Scheduler, the LLVM Execution Engine, and the Audio Driver (RtAudio).

Runtime

Scheduler

Internal Time

Task Queue
[{100, &fn1},
{200, &fn2}, ...]

LLVM Execution Engine

Audio Driver (RtAudio)

Interactions:

- addTask():** The LLVM Execution Engine sends tasks to the Task Queue in the Scheduler.
- runTask():** The Scheduler runs tasks from the Task Queue.
- tickTime():** The Scheduler updates the Internal Time.
- dsp():** The Audio Driver sends data to the LLVM Execution Engine.
- now:** The Scheduler provides the current time to the LLVM Execution Engine.