# Glitch

A Visual Compiler

# Table of Contents

# Introduction

Glitch is a visual compiler. The software interprets block diagrams (blueprints) and translates the diagrams into Arduino intermediate source. Glitch is extensible with other frameworks, for example, the C programming language.

Glitch should be functional on any operating system where Qt 5 LTS or Qt 6 LTS is supported. Qt 5.5.1 is supported for PowerPC and other operating systems. Qt 4.8.x is considered obsolete and is not supported.

The source of Glitch is available at https://github.com/textbrowser/glitch.

# Arduino Special Functions

The Arduino programming interface requires two special functions, loop() and setup(). The functions are automatically assigned to an Arduino diagram after a diagram is created.

## Creating New Diagrams

New diagrams may be created via File → New Diagram → Arduino. After a diagram is initialized, editing may begin. To add an object, drag-and-drop it from the Categories tree widget. Objects may also be added from the copy buffer via paste events. Diagram information is recorded in a portable SQLite database.

## Editing Diagrams

Existing objects may be edited via direct interactions. A context menu is also available for each object. The context menu contains generic and specific properties. Copying and pasting objects are also allowed. A single redo / undo stack provides rich redo / undo behavior.

## Operating Systems

Glitch supports Android, FreeBSD, Linux, Mac OS X, OS/2, OpenBSD, and Windows. Generally, the application should be compatible with any operating system where a modern Qt is supported. The software has also been tested on a variety of architectures, including AMD, ARM, PowerPC, and SPARC.

## SQL Injections

All Glitch SQL queries are parameterized. Prepared SQL statements are resilient against SQL injections.

# Wiring Objects

Wired objects designate a graphical relationship between the wired objects. For example, a variable object wired to a function object suggests one of two things. For a main-diagram function, a wired variable connected to the function implies that the function has one parameter. For a non-main-diagram function, a wired variable (or another object type) suggests that the function be issued with the wired input.


```
long rotl(long x, long b)
{
 return(((((x) << (b))) | (((x) >> (((64L) - (b)))))));
}
```

In this example, the order of the objects on the diagram describes the function’s parameter order.

# Index