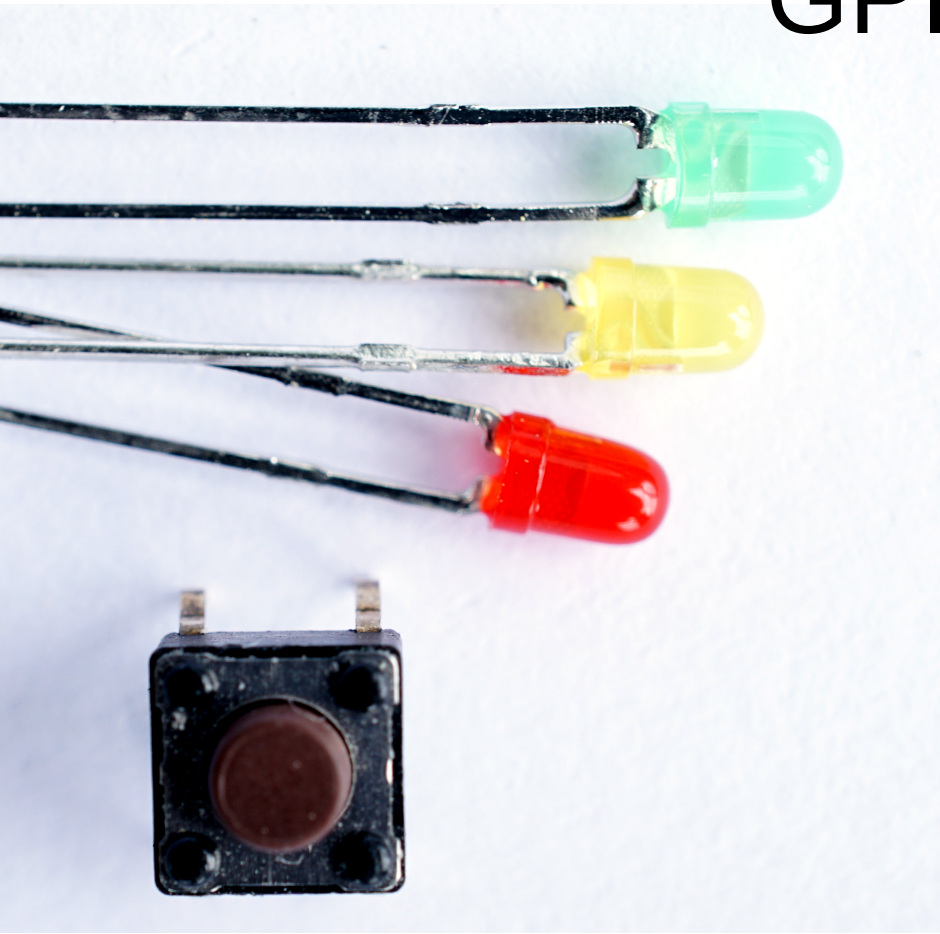


Pi and More 9

Workshop Raspberry Pi
Scratch
GPIO



Gerhard Hepp

Raspberry Pi

Populärer Microcomputer

ARM-basiert

Linux

Microcontroller-Peripherie GPIO, SPI, I2C, Seriell, USB

Scratch

Integrierte Entwicklungsumgebung 'IDE'
Insbesondere für Programmieranfänger entwickelt
und geeignet.

In Version 1.4 auf Raspberry Pi verfügbar.



Scratch hatte ursprünglich keine Hardwareanbindung, seit Version 1.3 mit dem „Remote Sensors Protocol“



Navigation

[Scratch Wiki Home](#)
[Random Page](#)

Tools

[What links here](#)
[Related changes](#)
[Special pages](#)
[Printable version](#)
[Permanent link](#)
[Page information](#)

Help the wiki!

The Scratch Wiki is made by and for Scratchers. Do you want to contribute?

Remote Sensors Protocol

This article or section documents an outdated version of Scratch (version 1.3/1.4). For this article in the current version, see [Scratch Extension Protocol \(2.0\)](#).

The **Remote Sensors Protocol** or **Scratch Extension Protocol** is an experimental extension feature that allows interaction between Scratch (1.3+) and other programs.

Contents

[\[hide\]](#)

- [1 Introduction](#)
- [2 Protocol](#)
- [3 Message Types](#)
- [4 Common Message Types](#)

Introduction

Three kinds of interaction are supported:

- sharing [broadcasts](#) (in both directions)
- virtual sensors
- Scratch's global [variables](#) are made visible

Protocol

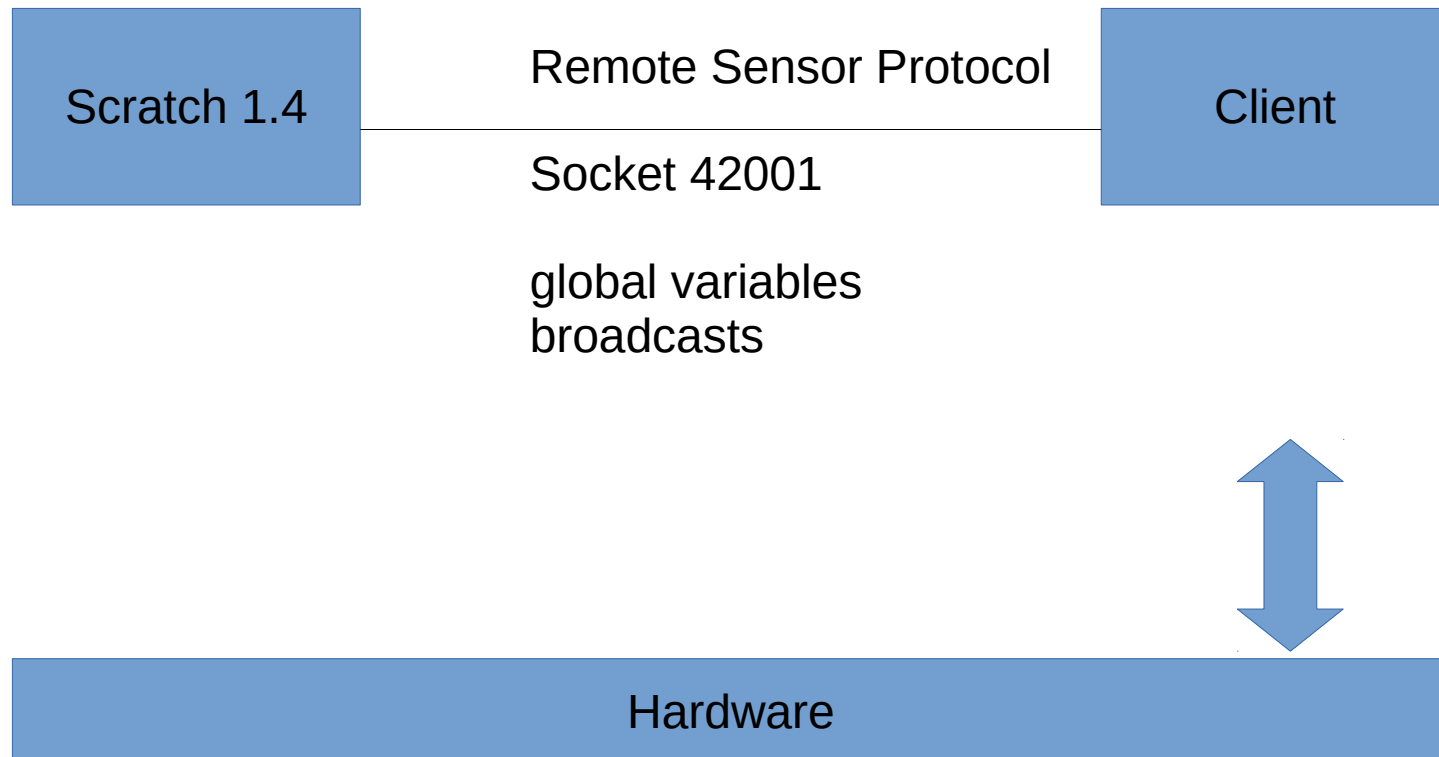
The experimental extension feature is enabled using the right-button menu on one of the two [sensor blocks](#). When remote sensors are enabled, Scratch listens for connections on TCP port 42001.

Once a connection is established, messages are sent in both directions over the socket connection

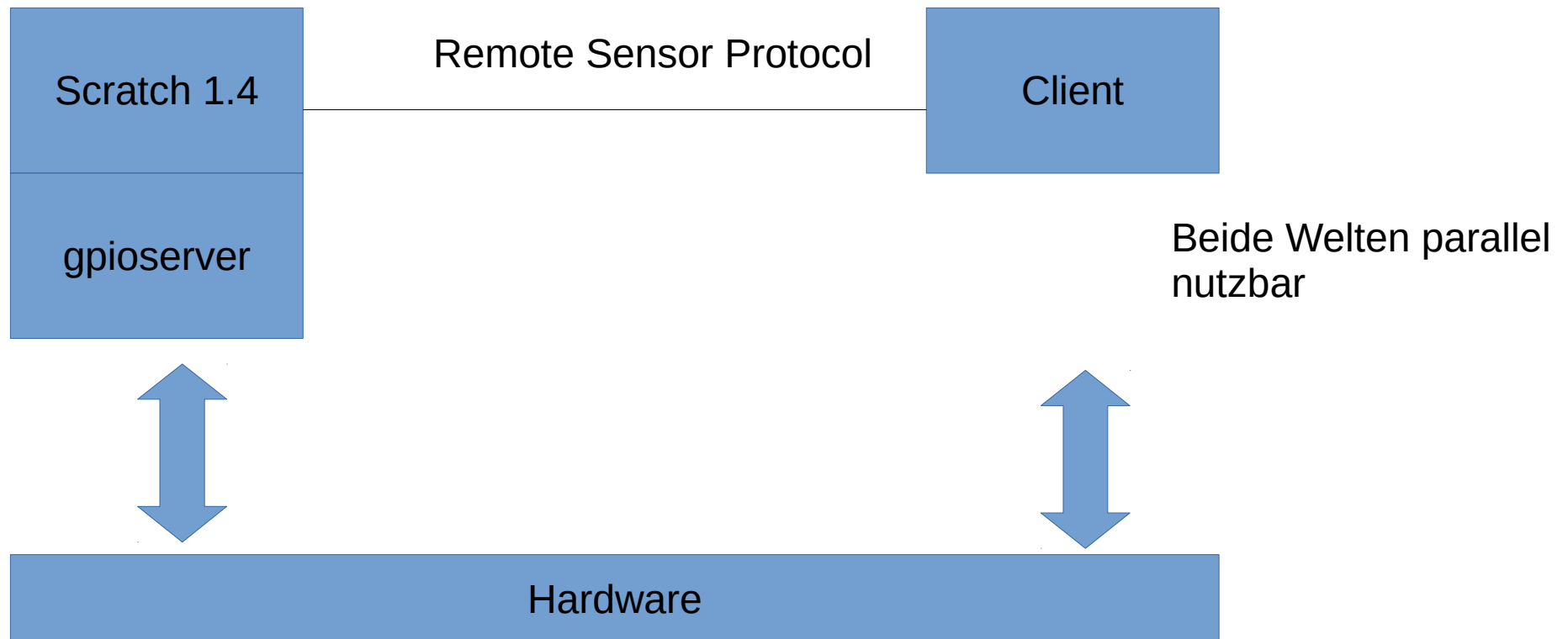
Client Programme mit Scratch

Beispiele für Client-Programme

- scratchGPIO
- **scratchClient**
- scratchpy API



'gpioserver' zur Hardwareansteuerung



ScratchClient

- „hardware abstraction“
Logische Namen für Peripherie
- Skalierungen, Linearisierungen, komplexe Devices
- Adapter -orientiert und dadurch einfach erweiterbar.
- Für die Anwendung in der Schule entwickelt
mit passenden Konfigurationen.

Pro and Con

gpioserver

- + no additional installations
- limited support for cards, devices
- + foundation supported
- limited development support

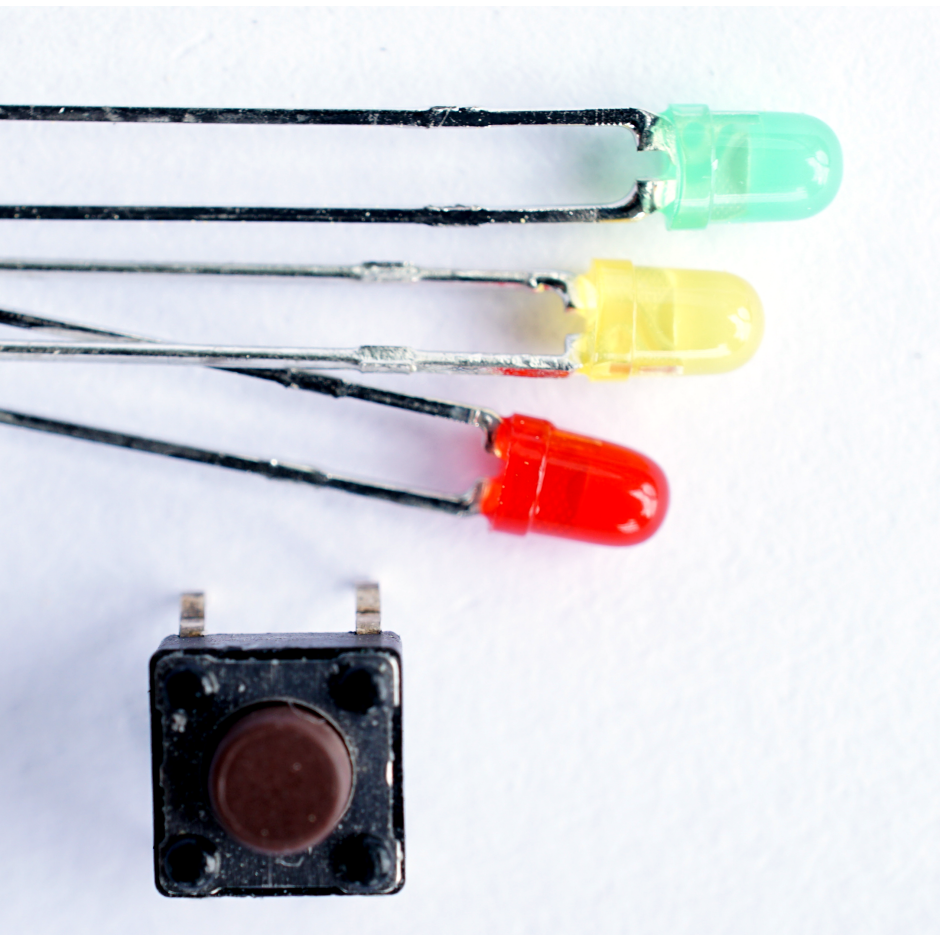
scratchClient

- + - additional installation needed
- + - config files, preconfigured setups possible
- + monitoring
- + extensible, python based

Workshop

Ansteuerung LED mit Taster

- einmal mit gpioserver
- einmal mit scratchClient



Zeitplan

10 min	Einleitung
10 min	Hardware Aufbau
30 min	gpioServer
40 min	scratchClient
10 min	Aufräumen
	Diskussion

110 min

Die Unterlagen sind auf dem Desktop des Raspberry Pi verfügbar.

Weitere Informationen

ScratchClient: heppg.de

Raspberry pi: raspberrypi.org

Scratch: scratch.mit.edu