

## ■機能概要

Scratch2 用の HTTP 拡張プログラム(C 言語ソースコード)です。

本プログラムは Scratch からハードウェアを制御するためのものです。Scratch からの制御要求を、HTTP 通信を介してハードウェアに伝えます。通信に必要な HTTP ヘッダは自前で用意していますので、HTTP サーバ無しで動作します。

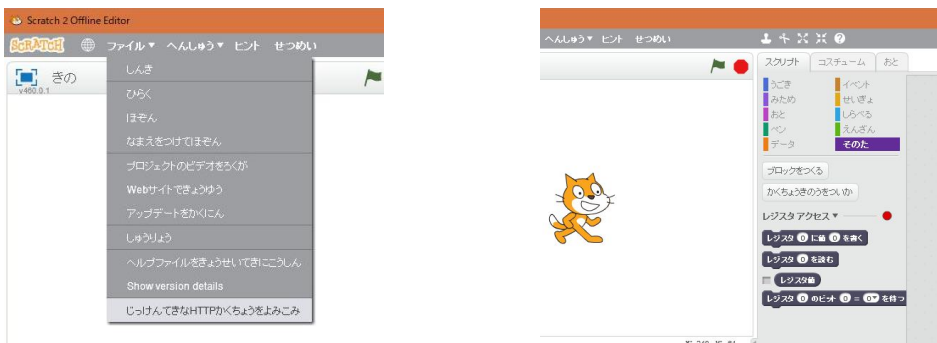
本プログラムには実際のハードウェアレジスタにアクセスするコードなどは含まれていません。ハードウェア制御の際に、物理アドレス空間へのアクセスが必要な場合には、別途販売中の「Physical Memory Driver」をご使用ください。

※Scratch (スクラッチ) は、MIT メディアラボが開発した、ビジュアルプログラミング言語です。

## ■Scratch の準備

Scratch2 offline Editor にて「Shift」キーを押しながら、「ファイル」メニューを選択すると、「実験的な HTTP 拡張を読み込み」というメニューが表示されます。

このメニューを選択後、本プログラム同梱の「reg\_acc.json」を指定してください。「スクリプト」タブの「その他」に新たなブロックが表示されます。



## ■プログラムのコンパイル

Linux 版、Windows 版のそれぞれのディレクトリに makefile が格納されていますので、そちらを使用してコンパイルしてください。

Linux 版の場合、gcc 開発環境でコンパイル可能です。

Windows 版の場合は VisualStudio の開発者コマンドプロンプトから「nmake」を実行してください。

## ■プログラムの起動

本プログラム(regsv)を起動後、Scratch との通信が開始されると、「レジスタアクセス」の右側のインジケータが赤から緑に変わります。



## ■通信内容

以下に、Scratch と本プログラム間の通信内容について記します。ソースコードとあわせて参照ください。

Scratch の送信コマンドです。

コマンド	説明
poll	規定のコマンドです。1/30 秒周期で送信します。
reset_all	規定のコマンドです。赤色の停止ボタン●を押した場合に送信します。
write_reg	追加コマンドです。ブロック <code>レジスタ 0 に値 0 を書く</code> 実行時に送信します。
read_reg	追加コマンドです。ブロック <code>レジスタ 0 を読む</code> 実行時に送信します。
wait_reg	追加コマンドです。ブロック <code>レジスタ 0 のビット 0 = 0x を待つ</code> 実行時に送信します。

本プログラムのレスポンスです。

レスポンス	説明
_busy	規定のレスポンスです。wait_reg コマンドによる待ち合わせ中に送信します。
reg_value	追加レスポンスです。ブロック <code>レジスタ値</code> の値に反映されます。

Scratch Extension の詳しい仕様については [https://en.scratch-wiki.info/wiki/Scratch\\_Extension](https://en.scratch-wiki.info/wiki/Scratch_Extension) を参照ください。