

Digital Synth VRA8-N mode-VC for Arduino Uno

ISGK Instruments 2019/8/3-4 Maker Faire Tokyo 2019

Arduino Unoの限界に挑戦する「Digital Synth VRA8」シリーズ第6弾。8ビットCPUひとつで音を合成するモノフォニック（単音）シンセサイザー（MIDI音源）です。スケッチはフリーで公開しており、Arduino Unoと抵抗、コンデンサ、オーディオジャックだけで製作できます。新モード「VRA8-N mode-VC」では電圧制御に対応し、ポテンショメータを使って演奏できます。

1 Digital Synth VRA8-N (v2) 仕様

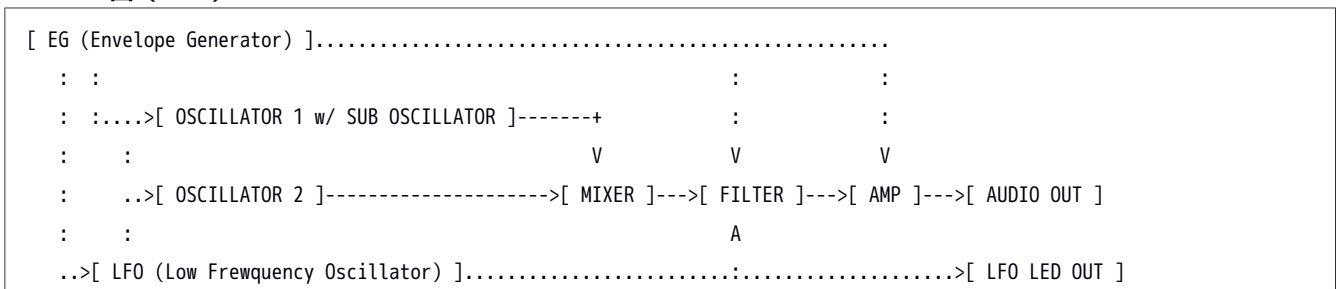
- Arduino Uno（8ビットCPU）だけでサウンド生成する，モノフォニック（単音）シンセサイザー（MIDI音源）
- 「mode-VC」で動作させることで電圧制御に対応，ポテンショメーター等を使って演奏可能
- MIDI入力：USBシリアル（38.4 kbps）またはMIDI（31.25 kbps） ※「mode-VC」でもMIDI制御可能
- オーディオ出力：単極PWM（62.5 kHz，ピンD6 or D5）+RC回路（推奨，fc：15.9 kHz，R：100 Ω，C：100 nF）
- オシレーター：ノコギリ波・矩形波・ノイズ（2のみ） ● サブオシレーター：サイン波・矩形波・ノイズ
- LFO：三角波・三角波2・ノコギリ波・ランダム・矩形波（LED出力：ピンD5 or D6）
- フィルター：2次LPF（fc：784.0 Hz～12.5 kHz）
- サンプリング周波数/ビット深度：31.25 kHz/8 bit ● 位相分解能：24 bit ● 音域：C-1～C9（中央ハがC4）
- ポルタメント・ピッチベンド・モジュレーション・エクスプレッション（またはベロシティ）：対応
- Arduino IDE 1.8.5 を強く推奨 ● スケッチはフリーで公開 ● Webサイト：<http://risgk.github.io/>

コントローラー一覧 (v2.4)

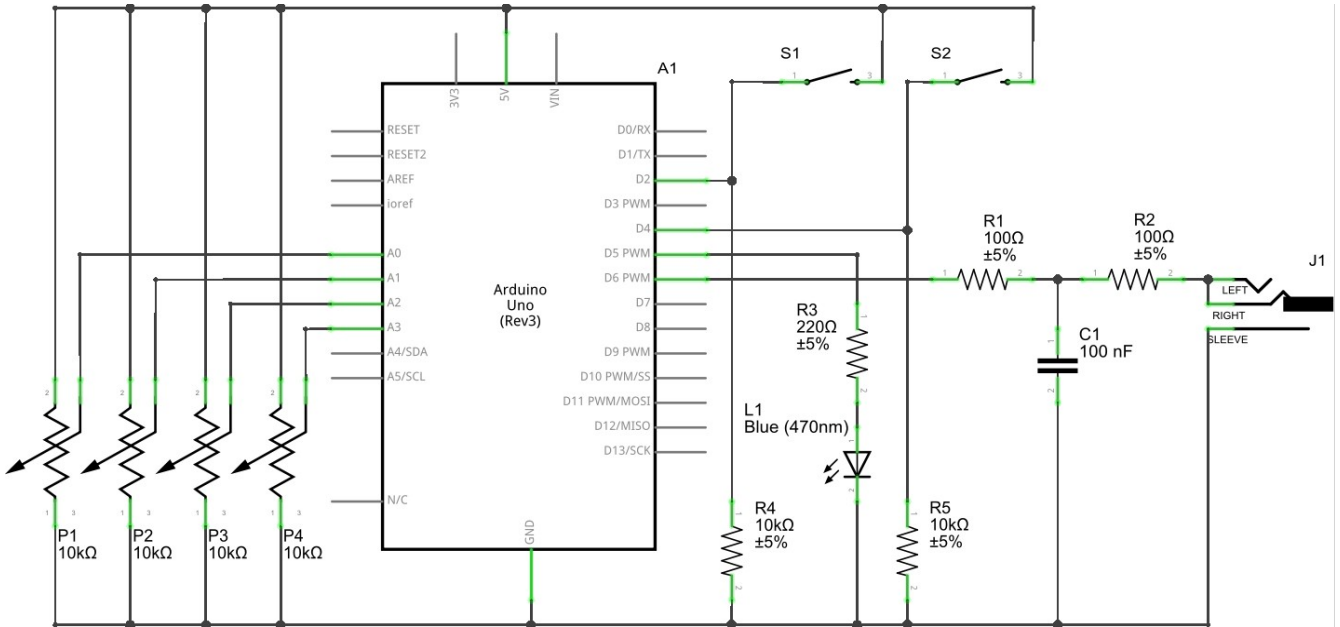
OSC1/2 WAVE (SAW/NOI/SQ)	OSC MIX (1/2)	OSC2 COARSE TUNE (-/+)	OSC2 FINE TUNE (-/+)
SUB WAVE (SIN/NOISE/SQ)	SUB LEVEL	EG > PITCH (-/+)	EG > PITCH TARGET (1&2/2)
CUTOFF FREQUENCY	RESONANCE	EG > CUTOFF (-/+)	AMP EG (OFF/ON)
ATTACK	DECAY	SUSTAIN	RELEASE (OFF/ON)
LFO WAVE (T/2/SA/RND/SQ)	LFO RATE	LFO > PITCH (-/+)	LFO > CUTOFF (-/+)
LFO DEPTH	EG > LFO RATE (-/+)	LFO > PITCH TARGET (1&2/2)	LFO FADE TIME
PITCH BEND RANGE ※	PITCH > CUTOFF ※	EXPRESSION > CUTOFF (-/+)	EXPRESSION > AMP LEVEL
PORTAMENTO TIME	LEGATO (OFF/ON) ※	KEY ASSIGN (L/L/P/H/LAST) ※	EXP BY VELOCITY (OFF/ON)
OSC LEVEL	RESONANCE LIMIT	AMP LEVEL	DAMP AND ATTACK

※ 「mode-VC」では無効

ブロック図 (v2.4)

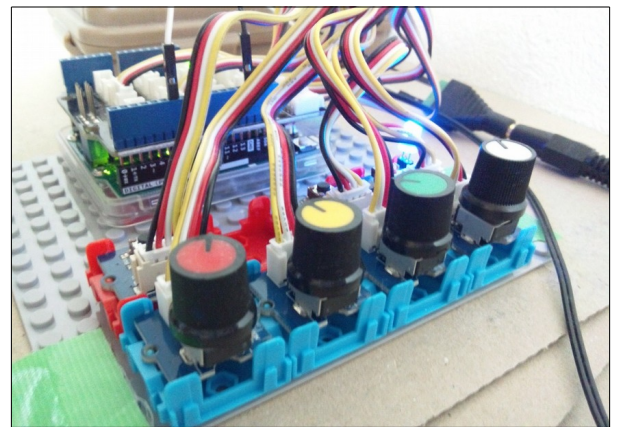


回路図 (v2.4)



fritzing

実装例 (Grove システムを使用)



2 「mode-VC」仕様

電圧制御 (voltage control) に対応し、ポテンショメーターやアナログシーケンサーを使って演奏可能. Arduino 用シンセの名作「Auduino (オーデューイノ)」へのリスペクトを含む. スケッチ書き込み前に #define でモードを選択する.

具体的には、0 ~ 5 V の制御電圧 (control voltage) によって音色や音高を調整する. デフォルトの設定は以下の通り.

(A4 や A5 の使用や GATE 入力対応の改造も可能)

A0 : CUTOFF 調整, **A1** : RESONANCE 調整, **A2** : OSC MIX 調整, **A3** : PITCH 調整 (いわゆる CV 入力, 0V で発音オフ),

D2 : 音色 PROGRAM 変更 (途中から RANDOM CTRL), **D4** : SCALE MODE 変更 (MODE 0 「C Major (2Oct / 5V)」, MODE 1 「Chromatic (2Oct / 5V)」, MODE 2 「Linear (5Oct / 5V)」)

3 「mode-VC」設計ノート

● **AD 変換処理の効率化** 開始・結果取得関数 (約 250 μ 秒後に呼ぶ) を用意して、変換待ちがビジーになるのを回避.

● **AD 値の「ゆらぎ」対策** 測定誤差やノイズによって調整値がズれるのを防止するため、AD 変換結果の値 (0 ~ 1023) を前回値と比較して「±1 以内の変化」は無視. ただし、Arduino を AC アダプターで動かさないと安定しない.

● **ボタン入力のチャタリング防止** カウンターを使って、「デジタル入力値 High/Low の変化後、約 50 ms 経過するまで」は電圧をチェックしないようにしている.

過去作品の資料

- ・「Digital Synth VRA8-N (v1) 説明資料」 (2018 年) http://risgk.github.io/digital-synth-vra8-n/mft2018_isgk_instruments.pdf
- ・「Digital Synth VRA8-Px 説明資料」 (2017 年) http://risgk.github.io/digital-synth-vra8-px/mft2017_isgk_instruments.pdf
- ・「Digital Synth VRA8-P 説明資料」 (2016 年) http://risgk.github.io/digital-synth-vra8-p/mft2016_isgk_instruments.pdf
- ・「Ruby x Arduino でシンセサイザーを作ってみた」 (2015 年) <https://gist.github.com/risgk/0db52ea683530652d933>