# ARPIE USER GUIDE – FIRMWARE VERSION 5.X

## Quick Start

OK so I know you just want to get started... So connect up your MIDI keyboard to MIDI in, and a sound module to MIDI out. Switch on the ARPIE.. after a short delay (1-2 seconds) the 16 blue data display LEDs should light and the CLK LED should start blinking.

Hold down a chord on the MIDI keyboard. The RX LED should blink as notes are received, and the TX LED should blink rapidly as notes are sent. Press the HOLD button once, the HOLD LED should illuminate. Now when you play a chord, the Arpeggiator continues to play after you have released the keys.1

If you don't have a keyboard handy, turn on HOLD, press INST and then press the small button below the right-most blue LED. This should start playing a C major chord arpeggio.

OK, now we've broken the tension lets move on!

## Introducing ARPIE

ARPIE is a MIDI arpeggiator based around open-source software and hardware. The Arduino-based firmware allows maximum hackability and the hardware is designed for easy DIY assembly and customisation.

Despite its simple construction and appearance, ARPIE is full featured with various chord arpeggiation, tempo synchronisation and standalone features to enable ARPIE to be used creatively with or without an attached MIDI keyboard.

A distinctive feature of ARPIE is the minimal control surface. While this has only simple LEDs for visual feedback, it is easy to use once you get familiar with it.

## Basic Operations

When you play a chord into ARPIE, it takes the notes of that chord and builds them into an "arpeggio" based on the selected arpeggio mode (MODE button).

The argeggio is extended over multiple octaves, based on the selected octave span (SPAN), and transposed up or down by whole numbers of octaves selected by the octave shift (SHFT) setting, and by a number of notes based on the note transpose (TRAN) setting.

This argeggio is then extended based on the selected note insertion (INST) mode, for example by inserting the lowest note of the sequence between all other notes.

ARPIE plays the arpeggio sequence through the MIDI out port at a rate based on the tempo and synch settings (SYNCH) and the playback rate (RATE) setting.

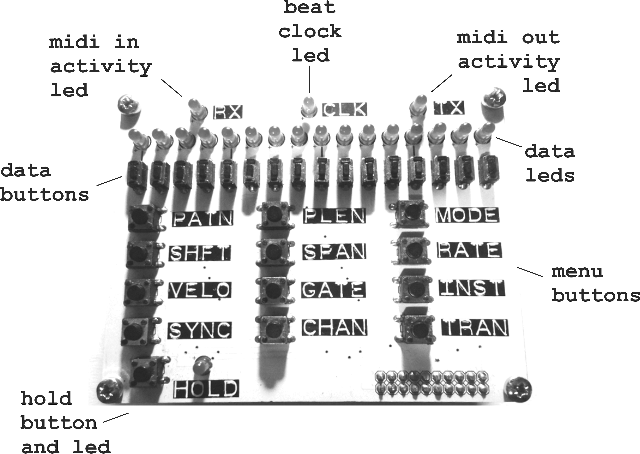
The output channel, volume and duration of the note are controlled by the MIDI channel (CHAN), note velocity (VELO) and gate length (GATE) settings.

Notes in the arpeggio sequence can be selectively muted using the rhythmic pattern (PATN) which loops, independently of the arpeggio sequence, at a selectable rhythmic pattern length (PLEN). The interplay between the arpeggio and the pattern can create interesting variations and grooves.

ARPIE has loads more features up it's sleeve such as accent and glide patterns, polyphonic chord gating and force-to-scale. Read on to find out more!

# Know Your ARPIE

## Control Surface



- Twelve \*\*menu buttons\*\* select different functions. Most buttons have different functions depending on whether you press and release the button, or hold it down for a number of seconds.

- Sixteen \*\*data leds\*\* and adjacent \*\*data buttons\*\* have functions that depend on the selected menu function. These are described in the following seconds.

- An additional \*\*hold button\*\* at the lower left controls how ARPIE responds when you release all keys on the controller keyboard. The button has additional functions that can be accessed by holding it for several seconds. The \*\*hold led\*\* indicates the selected function (off, steady or blinking)

## Rear Panel Connections

The rear panel of ARPIE is shown below:

<img class="wide" src="img/arpie\_back.png">

- The \*\*power\*\* connector accepts a 2.1mm barrel connector of a 9-12V DC supply. The supply can be either polarity (centre positive or centre negative). When a plug is inserted, the internal battery is disconnected.

- The \*\*midi in\*\* connector accepts MIDI input from a standard 5-pin DIN connector, generally from a controller keyboard. This input is opto-isolated.

- The \*\*aux midi synch\*\* connector accepts MIDI clock messages (only) via a standard 5-pin DIN connector. This input can be used to synchronise ARPIE to a drum machine etc. when the MIDI in connector is in use. This input is opto-isolated.

- The \*\*midi out\*\* connector accepts a standard 5-pin DIN connector and is generally connected to a synth or sound module.

- The \*\*reset button\*\* performs a hard reset of ARPIE, equivalent to a power off/on cycle. This is typically most useful when loading new firmware.

## Left Side

<img class="wide" src="img/arpie\_left.png">

- The \*\*on/off switch\*\*... turns the power on and off (battery or external DC), leading us to the \*\*power led\*\* !

- The \*\*battery holder\*\* accepts a 9V PP3 battery. Alkaline batteries recommended. To replace the battery you need to remove the four screws that secure the control surface and pull the control surface connector pins from their socket.

## Right Side

<img class="wide" src="img/arpie\_right.png">

- The \*\*hack header\*\* breaks out 5V power and four spare I/O lines from ARPIE's microcontroller. It allows DIY add-ons and offers a lot of fun for those brave enough to experiment with it... more about this later!

- \*\*Diagnostic LED 1\*\* is connected to the main ATMEGA328 microcontroller - ARPIE's "brain". When the brain is alive, this LED should blink about once per second.

- \*\*Serial Programmer Header\*\* allows new program code to be loaded into ARPIE's brain using a suitable programmer (a 6-pin USB-TLL serial programmer).

## Front Side

<img class="wide" src="img/arpie\_front.png">

- \*\*Diagnostic LED 2\*\* is connected to the PIC16F1825 microcontroller that manages the aux midi synch port. This LED should blink every second or two.

- \*\*PIC icsp header\*\* can be used to update firmware on the PIC chip, using a PICKit2 type programmer. Usually it would be rare to need to do this, and typically there is no need to solder a header here

- \*\*Control Surface Connector\*\* electrically joins the control surface to the main board. When inserting the connector in the socket be careful to make sure all the pins are aligned correctly.

## General Use

- When you press one of the twelve main menu buttons, the function of the sixteen data entry buttons is changed according to the selected function.

- Most menu buttons have dual functions: press and release the button to access the primary function, press and hold the button for a few seconds to select the secondary function.

- To exit from a secondary menu function, press the same (or a different) menu button. There is no need to exit from primary menu function (you can just press another button)

- By default ARPIE returns to the pattern edit mode (PATN function) after a few seconds of inactivity. You can turn this off if you want.

- Sections in the manual describe how the 16 data buttons are assigned in each mode. A yellow colour is used for primary menu function and a purple colour for secondary.

## Chord

The first thing ARPIE needs to make music is a **chord**. ARPIE will take that chord, stretch and twist it but it firsts need you to tell it what notes to play! There are a few different ways of doing this

* You can play a chord into ARPIE via the MIDI input. Usually this would be by playing a chord using an attached MIDI keyboard
* You can manually enter a chord by toggling individual notes on and off. This feature is available by long-pressing the INST menu button
* You can select one of a set of standard chord shapes or a randomised chord. These options are available by clicking the INST menu button

### MIDI Note Input & HOLD Mode

To play chords into the ARPIE via MIDI you need to connect a MIDI cable between the MIDI controller (keyboard, sequencer, DAW etc) and ARPIE’s main MIDI IN socket (Notes cannot be received on the SYNC socket)

You need to make sure that ARPIE is listening to the correct MIDI channel to receive the notes that you are sending. This is set by long-pressing the CHAN button then any one of the 16 step buttons to select a MIDI input channel from 1 to 16. If you press the button twice, ARPIE will enter OMNI mode, which means it will receive notes sent on any MIDI channel (this is the default) which is fine is you have only a single keyboard connected.

When ARPIE receives MIDI notes, it will immediately start to arpeggiate them and sent notes to the MIDI output. The arpeggiation will stop as soon as all notes are released and will then restart with the next note. When playing on a keyboard it may be difficult to time chord changes precisely and this can result in playback being off the beat. Also you may want the arpeggiation to continue after you have released the keys on the keyboard.

The way around these two issues is to use HOLD mode. HOLD mode can be turned on and off by pressing the HOLD button. The HOLD LED will be lit when hold mode is active.

When playing chords manually against a beat you’ll usually get the best result using HOLD.

Turning HOLD off will stop the held arpeggio from playing.

### Manual Chord Insertion

Long press the INST button to go into manual chord insertion mode. Toggle notes on and off by pressing the data buttons. Each data button is mapped to a note as follows

TODO

As soon as the first note of the chord is set, ARPIE starts arpeggiating. HOLD mode is enabled automatically. Continue toggling notes on and off to change the chord as it plays.

### Preset/Random Chord Insertion

When INST is clicked, the following data buttons are mapped to chords

TODO

By pressing one of these data buttons you automatically set ARPIE up to play one of the following chords

* C Major
* C Minor
* C7
* C Maj7
* C Min7

There is also the ability to have ARPIE play a “random” chord. The chord has 2,3 or 4 notes which are always within a single octave from MIDI Note 48

ARPIE automatically enters HOLD mode when a chord is selected this way

Use the TRAN button to transpose playback of the chord. This is how you can use this feature to insert chords that have root notes other than C

### The Arpeggio

Once ARPIE has been fed a chord, it can get to work building a sequence of notes from that chord – the **arpeggio**.

The main difference between the chord and the arpeggio is that while a chord is a set of notes that are played together, the arpeggio is a sequence of notes that do have a particular order to be played in.

There are several ways in which you can control the creation of the arpeggio:

* The type of the arpeggio, controlled by the MODE button – for example if notes go up in pitch or down in pitch (see below)
* The range of octaves over which the arpeggio is spread, controlled by the SPAN button – each note in the chord is transposed up one octave in pitch for every additional octave to be spanned, up to a maximum of 4 octaves (i.e. the original pitch plus 3 higher octaves)
* Inserted notes which morph and twist the arpeggio in different ways, controlled by the INST button (see below)

### Arpeggio Types

The notes of the chord are turned into an arpeggio based on the selected arpeggio type, which you access via the MODE button.

TODO

There are the following arpeggio types

* UP – ARPIE sorts the notes of your chord into ascending order of pitch, extending the arpeggio over the number of octaves you select with SPAN button
* DOWN – ARPIE sorts the notes of your chord into descending order of pitch, extending the arpeggio over the number of octaves you select with SPAN button
* UP/DOWN – This is like the arpeggio of DOWN mode being added to the end of the arpeggio of UP mode, so the notes play up over the octave span then back down again. The highest and lowest notes are only played once each cycle.
* RANDOM – ARPIE places the notes of the chord into a random order, extending the arpeggio over the number of octaves you select with SPAN button
* KEYBOARD ORDER – The notes of the chord are not sorted before arpeggiation – maintaining the order the notes were played in to the MIDI input. This order is preserved as the arpeggio is extended over the number of octaves you select with SPAN button
* GATED CHORD – This is not really mode of arpeggiation but rather a special way of playing the chord back. Instead of playing the notes one-by-one in a sequence, ARPIE plays them all at the same time. The SPAN button can still be used to extend the chord over multiple octaves. Inserted note modes do not work in this mode.

### Inserted Notes

Once the notes of the chord have been sorted and extended, the next step is to allow some further extension and twisting of the arpeggio sequence. These modes are accessed via the INST menu option

TODO

* NONE – There is no further change to the arpeggio sequence
* HIGHEST – The highest note in the arpeggio is inserted between every other pair of notes
* LOWEST – The lowest note in the arpeggio is inserted between every other pair of notes
* 3 STEPS FORWARD, 1 STEP BACK – The arpeggio is extended by jumping back one step after every three steps forward. For example ABCDEF becomes ABCBCDCDEDEF
* 4 STEPS FORWARD, 2 STEPS BACK – The arpeggio is extended by jumping back two steps after every four steps forward. For example ABCDEF becomes ABCDCDEFEF

These options aren’t applicable in gated chord mode.

### Transpose

The notes of the arpeggio can be changed in pitch during playback

By clicking the SHFT button you can access the octave shift feature which allows the entire arpeggio to be transposed up or down by full octaves (up to three octaves in each direction)

By clicking the TRAN button you can transpose the entire arpeggio by single semitone increments (chromatically), from 3 semitones down to an entire octave up.

These two transpose features work together, so you can transpose (for example) down by 2 octaves then up by 3 semitones.

The best results for chromatic transposition can usually be got by using it with the Force To Scale mode so that notes in the arpeggio remain in key and the musical intervals between notes in the arpeggio will change depending on the transpose amount, allowing new and interesting chord progressions to be found. With an arpeggio locked down in HOLD mode, the TRAN feature can be played like an instrument in itself!

### Sequenced Transpose

How good would it be if ARPIE could remember a sequence of your transpose actions and play them back in time like a chord progression? Well it can!

With an arpeggio held and playing, enter transpose mode by clicking TRAN. Keep the TRAN button held down and press a sequence of transpositions using the data button. Release the TRAN button and now your transpositions will be replayed repeatedly.

Each time you add a new transposition to the sequence, that transposition will usually apply for 16 steps of arpeggio playback (corresponding to the movement of the current step LED when in PATN mode during playback). You can change the rate at which transpositions happen by using the PLEN feature (see later).

If you want a given transposition to play for longer, just press the button multiple times when recording the sequence. For example if you want transpose 0-0-3-5 then hold down TRAN, press button for 0 transpose twice followed by 3 then 5

### MIDI Keyboard Transpose

It is also possible to transpose a held arpeggio using an attached MIDI keyboard.

You can do this when an arpeggio is locked down using the hold button.

Hold down the HOLD button again until the HOLD LED blinks. Now playing MIDI notes into the ARPIE on the selected input channel will transpose the sequence. The amount of transposition depends on the difference between the pitch of the lowest note in the chord and the note you play on the controlled keyboard.

Click HOLD to exit from MIDI transpose mode and return to normal HOLD mode.

Note that the “long press” function of the HOLD button can be set to do two different things (MIDI transpose or MIDI lockout). You select between these functions by long-pressing the MODE button to access the performance options menu. Ensure data LED #1 is ON in order to use MIDI transpose

### Force to Scale

ARPIE’s force to scale modes allow you to make sure all the notes in the arpeggio are “in key”.

While you can use this feature to correct chord playing, the real power comes when combining it with the chromatic transpose features.

This is because when a sequence of notes that are all “in key” is transposed up or down by an amount that is not a whole number of octaves, some of those notes are pushed out of key. If one of those notes is “corrected” by sharpening or flattening it to bring it into key again, the musical interval between that note and others will be changed and therefore the whole “feel” of the arpeggio will be changed.

TODO

### The Rhythmic Pattern

So far we’ve seen how a chord (a set of notes in no particular order) has been tue

ARPIE's **rhythmic pattern** is a sequence of up to sixteen steps that runs against the arpeggiated sequence and modifies the way that the sequence is played. This is most obviously seen in the way that ARPIE steps over the 16 data LEDs during arpeggiation.

While an arpeggio is playing, ARPIE repeatedly loops over the steps of the rhythmic pattern. The pattern usually has sixteen steps but it is possible to change this to any number between 1 and 16 using the **PLEN** (Pattern Length) button

Each step in the pattern will affect the note in the arpeggio that is playing at that step. The arpeggio may contain more, or less notes than the rhythmic pattern. Both the pattern and the arpeggio loop together, lining up in different ways each time they repeat, resulting in interesting variations.

The pattern has several **layers available** which do different things

|  |  |
| --- | --- |
| **Mutes** | This is the layer that is shown and edited when the PATN button is clicked. Also by default, the ARPIE will return to this layer when no menu button is pressed.  Each step in the mute layer decides whether the corresponding note in the arpeggio will play or not. Notes are muted when the pattern step LED is OFF.  It is possible to select if notes at a “mute” step will be skipped over, or will be played at the next “play” step. This selection is made in the p**erformance options** menu (Skip over muted notes in pattern) |

The following layers can be accessed one at a time by selecting one of them in the **performance options** menu (Second pattern layer) then long-pressing the PATN button. Note that the step must be active in the mutes layer for any other layer to apply to the note.

|  |  |
| --- | --- |
| **Accent** | Selected steps play at the accent velocity (selected by long-pressing the VELO button) where other steps play at the normal velocity (selected by clicking the VELO button)  Accent velocity can be more, less or equal to normal velocity. |
| **Full Duration** | Selected steps play at a full step duration (i.e. the time between two steps at the current tempo) where other steps play at the normal duration (selected by clicking the GATE button) |
| **Legato** | Selected steps are “tied” to the next active note meaning that the note will not end until the next note has started. This typically results in a “glide” type of playing. |
| **Octave lift** | Selected steps are transposed up one octave |
| **Octave drop** | Selected steps are transposed down one octave |
| **Fourth drop** | Selected steps are transposed down by interval of a fourth |
| **Polyphonic play-through** | If the following step also has play-through selected, the next note will play at the same time as the current note. This will repeat for a sequence of adjacent selected steps. |

### The Transpose Sequence

### Force To Scale

### Tempo and Rate

## PLEN

<a name="plen">

## Rhythmic Pattern Length

Press the PLEN button to view and change the loop length of the rhythmic pattern (from 1-16 steps). The current pattern length is indicated with a brighter LED and can be changed by pressing the corresponding data button.

The rhythmic pattern (and accent/glide) loops for the number of steps defined by PLEN. Setting a shorter PLEN does not clear the later steps in these patterns, so you can bring them back later.

<img class="wide" src="img/plen.png">

<a name="plen2">

## Preferences

Press and hold PLEN to access the configuration preferences

<img class="wide" src="img/prefs.png">

The first eight LEDs control built-in functionality for the <a href="#hh">hack header</a> (the small expansion header on the ARPIE base board). This is quite a big subject and has it's own section later in the manual!

- \*\*Auto Revert\*\* controls whether ARPIE will time out to the PATN menu when no buttons are pressed for a about 10 seconds. Turn this off if you prefer - you can always press the <a href="#patn">PATN</a> button to get back!

- \*\*Long Press Time\*\* controls how long you need to hold down one of ARPIE's menu buttons to access it's second function.

<center>

<table class="data">

<tr style="font-weight:bold"><td width="50">B1</td><td width="50">B0</td><td width="150">Long Press Time</td></tr>

<tr><td>ON</td><td>ON</td><td>1.5 seconds</td></tr>

<tr><td>OFF</td><td>ON</td><td>1 second</td></tr>

<tr><td>ON</td><td>OFF</td><td>0.5 seconds</td></tr>

<tr><td>OFF</td><td>OFF</td><td>0.25 seconds</td></tr>

</table>

</center>

- \*\*LED Profile\*\* sets the how the 16 data LEDs are controlled. Different types of LEDs have different characteristics; some are much brighter than others and need to be driven at shorter "duty cycles" to get decent contrast between ARPIE's three different LED brightnesses (bright, medium, dim). The LED profile shown below are recommendations but your may prefer to use a different ones (don't worry, it won't damage the LEDs to use the "wrong" profile!)

<center>

<table class="data">

<tr style="font-weight:bold"><td width="50">B1</td><td width="50">B0</td><td width="300">Recommended for LED type</td></tr>

<tr><td>OFF</td><td>OFF</td><td>Very high intensity (e.g. bright white)</td></tr>

<tr><td>OFF</td><td>ON</td><td>High intensity (e.g. bright blue/green)</td></tr>

<tr><td>ON</td><td>OFF</td><td>Medium intensity (e.g bright red)</td></tr>

<tr><td>ON</td><td>ON</td><td>Low intensity (e.g standard red/green)</td></tr>

</table>

</center>

All of these settings are saved in EEPROM. This means they are remembered when ARPIE is switched off.

# MODE

<a name="mode">

## Arpeggio Mode

<img class="wide" src="img/mode.png">

This option allows you to select between ARPIE's basic arpeggiation modes:

- \*\*Up\*\* mode plays notes of the chord in ascending order

- \*\*Down\*\* mode plays notes of the chord in descending order

- \*\*Alternate\*\* plays notes of the chord in ascending then descending order. If the sequence spans multiple octaves, all octaves are spanned on the way "up" before returning "down". The highest note is not repeated.

- \*\*Random\*\* plays the notes of a chord in a random order. This randomised order is determined when the arpeggio is built and is repeated until the arpeggio is rebuilt (e.g. press the Random Mode button again)

- \*\*Manual\*\* plays the notes of the chord in the order they were received on the MIDI input.

<a name="polygate">

- \*\*Poly Gate\*\* plays all the notes of the chord at the same time. Thicken up with <a href="#span">SPAN</a> and add some groove with <a href="#patn">PATN</a> to get dancey chord stabs!

## Pattern Fill

The far right data entry buttons can be used to reset the <a href="#patn">PATN</a> and <a href="#plen">PLEN</a> information.

- \*\*Random\*\* sets steps to a randomised combination of play and mute steps and sets PLEN to a random value.

- \*\*All Mute\*\* sets all steps to mute (LED off) and sets PLEN to 16 steps

- \*\*All Play\*\* sets all steps to play (LED on) and sets PLEN to 16 steps

Accent and Glide patterns are not affected.

<a name="mode2">

## Performance Options

<img class="wide" src="img/opts.png">

Press and hold MODE to access the Performance Options. This menu contains settings that control other ARPIE functions, changing ARPIE's musical performance.

- \*\*Hold Fn2\*\* controls the long-press function of the <a href="#hold">HOLD</a> button. When this option in OFF, the second function of HOLD is \*\*MIDI lock\*\*. When the option is ON the second function is \*\*transpose by MIDI input\*\*. See <a href="#hold2">here</a> for more information.

- \*\*Skip Rest\*\* controls what happens when there is a rest (LED off) at the current position in the <a href="#patn">PATN<a> sequence. If the option is OFF, the arpeggiator note at that position is skipped over and does not play. If the option is ON, the arpeggiator inserts a rest and plays the note at the next step.

- \*\*Glide Tie\*\* controls the operation of the <a href="#patn2">glide</a> function. If the option is OFF, the arpeggiator note coinciding with a glide point is played for one whole step. If the option is ON, then the note is "tied" to the next playing note.

- \*\*Patn Fn2\*\* controls the long-press function of the <a href="#patn2">PATN</a> button. When this option in ON, the second function of PATN is \*\*accent\*\*. When the option is OFF the function is \*\*glide\*\*

# SHFT

<a name="shft">

## Octave shift

<img class="wide" src="img/shft.png">

This function transposes the arpeggiated sequence up or down by up to three whole octaves. The default, no shift, position is indicated by a slightly brighter LED.

<a name="shft2">

## Force To Scale: Root Note

<img class="wide" src="img/froot.png">

Hold the SHFT button to access the Force to Scale Root note option.

This option works with <a href="span2">force to scale mode</a> to set the root note of the musical scale to which notes are being forced. For example if force to scale root note is set to D# and force to scale mode is Locrian, notes will be forced into a D# Locrian mode. See <a href="span2">force to scale mode</a> for more information.

# SPAN

<a name="span">

## Octave Span

<img class="wide" src="img/span.png">

This option determines the number of octaves the arpeggiated sequence will be extended over. This also works in <a href="#polygate">polyphonic gate mode</a> to layer additional octaves worth of notes on the chord.

<a name="span2">

## Force To Scale: Mode

<img class="wide" src="img/fscale.png">

The force to scale function is accessed with a long press to the SHFT button. When enabled, this feature makes sure that all notes output from ARPIE fits within a specific musical scale (or "key"). This can sound especially good when transposing held arpeggios, as the note intervals within the chord change to keep it in key.

A musical key is made up of a root note (for example "C") and a set of note intervals which together give us a scale (e.g. "C minor").

These sets of note intervals have wierd names derived from ancient Greek tribes! Don't let that put you off though - they are just the sequences of note intervals you get when you play along the white notes of a piano starting at different points

For example if you play white notes C through C you play a Major scale, which happens to be the same as the "Ionian" mode. If you play white notes A through A you are playing a Minor scale, or "Aeolian" mode. The other modes are just the sets of intervals you get by playing sequences of white notes with different start points... simple! but each mode has a different "feel" and we can have fun playing with them!

- The \*\*chromatic\*\* scale includes all notes. Selecting this option effectively turns off force-to-scale. This is the default setting.

- The \*\*Ionian\*\* mode is equivalent to the intervals between all the white notes on a piano keyboard from C to C. It is the same as the normal major scale.

- The \*\*Aeolian\*\* mode is equivalent to the intervals between all the white notes on a piano keyboard from A to A. It is the same as the natural minor scale.

- The other modes follow the same pattern of white notes from D to D (Dorian), E (Phrygian), F (Lydian), G (Mixolydian) and B (Locrian). Each is a different way of dividing up the 12 semitones of an octave into 8 notes and has a different feel.

As well as changing musical mode, force-to-scale lets us select the <a href="#shft2">root note</a> (we need both a musical mode and a starting note to define a scale) and control how out of scale notes are corrected.

## Force To Scale: Note Correction

So - if you want to make an omelette you need to break a few eggs - and if you want to force a sequence of notes into a musical scale some of them just might not fit!

This option is accessed by a long press to the SPAN menu. It lets you decide what to do with these outlying notes. Each option has a different musical feel, so have a play!

- \*\*Skip\*\* out-of-scale notes are simply ignored and the next valid note is played. There is no rest introduced.

- \*\*Mute\*\* out-of-scale notes are not played, but a rest of a single step is added instead.

- \*\*Flatten\*\* out-of-scale notes are played, flattened by 1 semitone to bring them back into key.

- \*\*Sharpen\*\* out-of-scale notes are played, sharpened by 1 semitone to bring them back into key.

- \*\*Alternate\*\* out-of-scale notes are alternately sharpened and flattened. Each time the sequence is rebuilt, the action taken with the first corrected note toggles. You can hear the effect of this by pressing the Alternate button again when you have a sequence which is forced to scale.

# RATE

<a name="rate">

## Time Division

This option sets the length of notes in the arpeggiated sequence, as a fraction of a beat. While the BPM stays the same, the arpeggio will play faster or slower as the beats are divided up differently.

The default "normal" note time (at least in 4/4 time) is one sixteenth of a beat, so for example selecting eigths will play notes half the speed for a given BPM.

<img class="wide" src="img/rate.png">

- \*\*(dot)\*\* or \*\*(d)\*\* represent "dotted time" (one and a half times normal note duration)

- \*\*(tr)\*\* represents triplet time (two thirds normal note duration)

# VELO

<a name="velo">

## MIDI Velocity

This menu controls the MIDI velocity of the notes output by ARPIE. There are two velocity modes, described as follows:

## Select MIDI Velocity

<img class="wide" src="img/velo.png">

This mode allows you to override the velocity of all arpeggiated notes to one of 16 specific levels (0,9,17,26,34,43,51,60,68,77,85,94,102,111,119,127). In this mode the original velocity information from the controller keyboard is ignored.

To switch to original velocity mode, press the first data button (zero velocity) two times.

## Original MIDI Velocity

Original velocity mode plays each arpeggiated note using the velocity of the note as it was played on the controller keyabord. This mode can be identified by the left and right-most LEDs being on and all the others off.

To switch back to selected velocity mode, press any velocity button, including zero.

# GATE

<a name="gate">

## MIDI Note Gate Length

<img class="wide" src="img/gate.png">

Controls the gate length of the MIDI notes played by ARPIE. You can select a proportion of a "full step" or you can "tie" notes together.

- \*\*Tie Notes\*\* means that a note is not stopped until the next note is started, with the MIDI note off message being sent after the next MIDI note on. This allows notes to play right across rests in the rhythmic pattern allowing for some nice grooves!

# INST

<a name="inst">

<img class="wide" src="img/inst.png">

## Arp Note Insert

Extends the length of the arpeggiated sequence by repeating notes from the chord according to specific modes:

- \*\*None\*\* just plays the arpeggio sequence without adding any extra notes to it. This is the default.

- \*\*Highest\*\* inserts the highest note of the arpeggio sequence in between all the other notes

- \*\*Lowest\*\* inserts the lowest note of the arpeggio sequence in between all the other notes

- \*\*3F, 1B\*\* staggers the arpeggio sequence and introduces repetation of notes by playing forward three places then moving back one place and continuing.

- \*\*4F, 2B\*\* staggers the arpeggio sequence by moving forward four places then back two.

## Chord Insert

To the right of the screen are some options for inserting chords without a keyboard. \*\*The HOLD function must be on for this to work!\*\*

- \*\*Random\*\* creates a chord from randomised notes.

- The other buttons insert minor, minor seventh, major seventh, seventh. minor, major chords. The root note is C but you can use the TRAN function to change this.

# SYNC

<a name="sync">

## Clock Source And BPM

<img class="wide" src="img/sync.png">

This screen controls the ARPIE's MIDI synch mode and it's internal metronome (BPM).

- The \*\*Source\*\* button controls the clock source. When the LED in OFF, ARPIE runs off its own internal clock source. When the LED is ON, ARPIE runs on external MIDI clock via the MIDI IN or AUX SYNCH IN sockets

- The \*\*Send\*\* button controls whether ARPIE sends a MIDI beat clock to the MIDI OUT port. The clock is sent when the option is ON. When this option is switched from OFF to ON, ARPIE also sends a "start" command to the OUT port, which can be used to synchronise MIDI slave devices. Conversely a MIDI "stop" command is sent when the Send function is switched off.

- When ARPIE is running from it's internal clock source, the \*\*BPM\*\* buttons can be used to set the tempo to a specific value. BPM defaults to 120 when ARPIE is switched on. The data display LED's indicate the approximate BPM.

- Pressing the \*\*Tap Input\*\* button two or more times allows you to set the tempo interactively.

- The \*\*Slower\*\* and \*\*Faster\*\* buttons can be used to adjust the BPM by steps of 1.

The internally generated BPM is approximate (but good enough for most stuff). However, an external clock source is recommended if a precise BPM is needed or if you need to sync to other gear as a slave.

## MIDI Routing Options

<img class="wide" src="img/midi.png">

These options give you control over how ARPIE internally routes MIDI information.

- \*\*Forward Channel Message\*\* can be switched ON so that non-note channel messages (CC change, pitch bend etc) which are received on ARPIE's configured input channel, can be sent through to ARPIE's output channel. This lets these performance parameters be sent from your controller keyboard to your synth.

- \*\*Thru Note\*\* - If you have ARPIE listening on a specific input channel (not omni mode) then this option controls whether note messages for other channels will be passed directly through to ARPIEs MIDI output.

- \*\*Thru Channel Message\*\* - This option controls whether non-note channel messages (CC, bend etc) for other channels will be passed through to ARPIEs MIDI output.

- \*\*IN Clock\*\* - determines whether ARPIE will listen for external MIDI clock ticks on the MIDI IN port. Source must also be set to external for these to be used.

- \*\*SYNC Clock\*\* - determines whether ARPIE will listen for external MIDI clock ticks on the AUX MIDI SYNC port. Source must also be set to external for these to be used.

- \*\*Flt.Ch.Mode\*\* - When ON, ARPIE will remove MIDI "Channel Mode Messages" (CC numbers 120 thru 127) from the data thru'd from input to output. Some controllers might send unwanted channel mode messages such as "all notes off" when releasing keys. ARPIE can filter these out if desired.

Note that MIDI thru is handed in ARPIEs firmware (there is no hardware thru) which can cause latency if the arpeggiator engine is busy - and possibly even dropped messages in situations of very high thru volume.

# CHAN

<a name="chan">

## MIDI Output Channel

When pressed, selects and displays the MIDI channel for output

<img class="wide" src="img/chan.png">

## MIDI Input Channel

<img class="wide" src="img/ichan.png">

If the CHAN button is pressed and held for a couple of seconds, you can Select The MIDI Input Channel. By default ARPIE will accept input from any MIDI channel (OMNI mode) but you can also select a specific channel, which affects MIDI Thru behaviour (see below)

OMNI mode is indicated by all the LEDs being ON. Pressing any single button will select an individual input channel. Pressing the same button again will restore OMNI mode.

# TRAN

<a name="tran">

## Chromatic Transpose

Transpose the MIDI output. Buttons represent -3. -2, -1, 0, +1, +2 through to +12 semitones. The default (no shift) position is indicated with a brighter LED.

<img class="wide" src="img/trans.png">

# HOLD

<a name="hold">

## Normal Operation

Before we look at the HOLD function, lets look at how things happen when when the HOLD function is switched off:

- ARPIE does not produce any output unless notes are currently active on the MIDI input (i.e. held down on the controller keyboard)

- When the first note of a chord is played on the input, ARPIE starts to play immediately (there is no delay until the next beat). The rhythmic pattern also restarts from the first step.

- Notes in a chord can be released, causing the arpeggio sequence to change. As long as at least one note remains held, playback will continue without interruption.

- When the last key of the chord is released, playback stops.

## Hold Function

The HOLD function can be activated by pressing the HOLD button, lighting the HOLD LED. When the HOLD function is enabled:

- Playback of the arpeggio will continue after the last note of the chord is released

- When the first key of a new chord is pressed, the held chord is cleared, but playback continues without any change to beat timing or position within the rhythmic pattern.

- You can stop playback by turning HOLD off by pressing the button again.

HOLD can be activated or deactivated while notes are already held.

<a name="hold2">

## Hold Button Secondary Function

When the HOLD button is held for a few seconds, the HOLD LED starts to blink, indicating that the hold button secondary function is active.

There are two functions available and you can select between them from the performance options menu (long press <a href="#mode">MODE</a>). These are described below.

## MIDI Lock function

This function can be used to "lock" the arpeggiator engine and start passing MIDI notes from input to output. This allows you to hold an arpeggio and then play over it.

MIDI lock can also be useful if you chain together multiple ARPIEs and want to play into a single one at a time while the others are playing sequences in hold mode.

Exit MIDI lock by pressing HOLD again. ARPIE will return to the previous HOLD mode.

## MIDI Transpose function

This function works a bit like MIDI lock, except that instead of passing input notes through to the output, a held arpeggio is transposed according to the interval between the note you play on the input and the lowest note in the arpeggiated chord.

This function is designed to work with the HOLD feature. You can have great fun when using this with force-to-scale options.

# Clock/Message Routing

This block diagram shows how data is routed around ARPIE

<a target="\_new" href="img/schematic.png"><img class="full" src="img/schematic.png"></a>

# Hack Header Modes

<a name="hh">

## Control Pots / Switch

You can connect up to three potentiometers (100kOhm - 1Mohm recommended) to ARPIE's hack header, configured as voltage dividers between 0 and 5V, and have them control various internal or MIDI parameters. You can also connect an active low switch to PB3 and have it trigger some predefined specific actions.

The following table shows how to configure hack header LEDs B7 - B0 on the system preference menu. 1 indicated LED is ON and 0 indicates OFF. A dot means this LED is not relevant to this setting.

<center>

<table class="data">

<tr style="font-weight:bold"><td width="50"></td><td width="250"></td></tr>

<tr><td>00......</td><td>PB3 Switch MUTE MIDI output</td></tr>

<tr><td>01......</td><td>PB3 Switch Restart Bar</td></tr>

<tr><td>0.00....</td><td>PC0 Pot Disabled</td></tr>

<tr><td>0.01....</td><td>PC0 Pot ARPIE BPM</td></tr>

<tr><td>0.10....</td><td>PC0 Pot ARPIE Gate Length</td></tr>

<tr><td>0.11....</td><td>PC0 Pot MIDI CC#</td></tr>

<tr><td>0...00..</td><td>PC4 Pot Disabled</td></tr>

<tr><td>0...01..</td><td>PC4 Pot ARPIE Velocity</td></tr>

<tr><td>0...10..</td><td>PC4 Pot MIDI Pitch Bend</td></tr>

<tr><td>0...11..</td><td>PC4 Pot MIDI CC#</td></tr>

<tr><td>0.....00</td><td>PC5 Pot Disabled</td></tr>

<tr><td>0.....01</td><td>PC5 Pot MIDI Mod wheel</td></tr>

<tr><td>0.....10</td><td>PC5 Pot ARPIE Transpose</td></tr>

<tr><td>0.....11</td><td>PC5 Pot MIDI CC#</td></tr>

</table>

</center>

The default MIDI CC numbers assigned to the three pots when in MIDI CC# mode are as follows

<center>

<table class="data">

<tr><td>PC0 Pot</td><td>CC#16</td><td>General purpose</td></tr>

<tr><td>PC4 Pot</td><td>CC#17</td><td>General purpose</td></tr>

<tr><td>PC5 Pot</td><td>CC#18</td><td>General purpose</td></tr>

</table>

</center>

Of course if you are prepared to dirty your hands in the code, you can make it do anything you like!

## Turning it off

The hack header should always be disabled, unless you specifically want to connect something to it. Otherwise you may get random things happening if pots are not connected ... or maybe you want that!

<center>

<table class="data">

<tr style="font-weight:bold"><td width="50"></td><td width="250"></td></tr>

<tr><td>00000000</td><td>Hack header disabled</td></tr>

</table>

</center>

## Pulse clock

Alternatively you can connect a 5V pulse clock to the header. To use pulse clock you need to set the hack header mode LEDs to the following

<center>

<table class="data">

<tr style="font-weight:bold"><td width="50"></td><td width="250"></td></tr>

<tr><td>10000000</td><td>Pulse Clock</td></tr>

</table>

</center>

The default settings are designed to work with 15ms high-pulse-per-step (e.g. Korg Volca) but you can edit and recompile the ARPIE source code to support other time formats.

The pin assignments on the header in this mode are

<center>

<table class="data">

<tr style="font-weight:bold"><td width="50"></td><td width="250"></td></tr>

<tr><td valign="top">PC5</td><td>Enable pulse clock (active low)</td></tr>

<tr><td valign="top">PC4</td><td>Pulse clock OUT</td></tr>

<tr><td valign="top">PC0</td><td>Pulse clock IN</td></tr>

<tr><td valign="top">PB3</td><td>Not used</td></tr>

</table>

</center>

You'll need to restart ARPIE after changing the pulse clock settings

## Other Possibilities

The hack header brings out 4 of the ATMEGA328 I/O pins. These pins can support some interesting functions that open up other possibilities for experimentation (as long as you are able to get your hands dirty in the code!). In future some of these might be implemented in the main firmware releases.

- \*\*PC5\*\* this is Arduino analog input 5. It is also the clock line for I2C and is a general purpose digital input or output (Arduino digital pin 19) which can support interrupt on pin change (PCINT13)

- \*\*PC4\*\* this is Arduino analog input 4. It is also the data line for I2C and is a general purpose digital input or output (Arduino digital pin 18) which can support interrupt on pin change (PCINT12)

- \*\*PC0\*\* this is Arduino analog input 0. It is also a general purpose digital input or output (Arduino digital pin 14) which can support interrupt on pin change (PCINT8)

- \*\*PB3\*\* this is Arduino digital pin 11, supporting a PWM (Arduino analogWrite) output and interrupt on pin change (PCINT3)

- \*\*VCC\*\* and \*\*GND\*\* regulated +5V supply from internal LDO regulator. Do not draw more than a few tens of milliamps from this supply!

Since the header breaks out the I2C bus, peripherals such as DACs, EEPROMs, Accelerometers etc could be supported with additional firmware code.

The presence of a PWM output allows generation of a control voltage that could be used to drive external equipment(although only 8 bit resolution is supported by Arduino analogOut)