

[HOME](#)[SHOP](#)[DEVELOPER](#)

GRIP PATTERNS

March 30, 2016

OVERVIEW

This post details how grips are produced and how grips can be customised in [Artichoke V1.2](#).

This tutorial is for:

- Hand: [Ada V1.1](#)
- Board: [Almond V1.2](#)
- Firmware: [Artichoke V1.2](#)

You will need:

- Ada hand with Almond board, running Artichoke V1.2
- [12V DC power supply](#)
- Micro USB
- Computer running [Arduino](#)

DEFAULT GRIPS

Artichoke V1.2 incorporates 6 predefined grip patterns;

- G0 Fist Grip
- G1 Palm Grip (Fist Grip but with thumb extended)
- G2 Thumbs Up
- G3 Point
- G4 Pinch
- G5 Tripod

When using Serial control, the grip is selected based on the grip number (G#) entered, when using muscle control, the grips are cycled through from 0-5 when the outer forearm muscle is tensed for a set duration, and when using HANDLE control, the grips can be cycled through by pressing either of the buttons.

(EACH OF THE CONTROL METHODS ABOVE WILL LINK TO THEIR SPECIFIC TUTORIAL)

FINGER POSITIONS

Each grip pattern is simply a combination of the individual finger positions, and each grip movement is a conversion of the desired grip position into the desired position of each finger within the grip. The function 'gripMovement()' maps the input grip position (0 - 100) to individual finger positions defined by the values within the 'animation[][]' array.

SIMPLE

Without
needed to
know how the
grip positions
are processed
(discussed in
COMPLEX),
you can

```

let animation[GRIP][FINGER][STEP][FINGER] = { // 3D array holding animation sequences
  { // FIST
    //COUNT
    [0], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //OPEN
    [10], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //NEEDPOINTA
    [20], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //NEEDPOINTB
    [30], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //CLOSED
  },
  { // PALM
    //COUNT
    [0], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //OPEN
    [10], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //NEEDPOINTA
    [20], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //NEEDPOINTB
    [30], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //CLOSED
  },
  { // THUMB
    //COUNT
    [0], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //OPEN
    [10], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //NEEDPOINTA
    [20], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //NEEDPOINTB
    [30], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //CLOSED
  },
  { // PINCH
    //COUNT
    [0], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //OPEN
    [10], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //NEEDPOINTA
    [20], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //NEEDPOINTB
    [30], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //CLOSED
  },
  { // TRIPOD
    //COUNT
    [0], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //OPEN
    [10], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //NEEDPOINTA
    [20], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //NEEDPOINTB
    [30], FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, FULLY_OPEN, //CLOSED
  }
}

```

modify the
positions of
the fingers
within each grip by modifying the values within the
'animation[][]' array within the 'AnimationPositions.h'
file.

The array is split into 6 segments, where each
segment represents one of the 6 grips. Each grip
segment is then split into 6 columns, a grip position
column (COUNT) and 5 finger position columns (F0 -
F4).

The array can be read by looking at the grip segment
you wish to modify, then read down the grip position
column to the grip position at which you wish to
modify the finger positions, the finger position at that
grip position can then be changed.

For example, if you wish to modify the position of a
finger when the hand is closed in the tripod grip, you
would look at the bottom segment (TRIPOD), look
along the bottom row (COUNT = 100) and modify the
finger position of the desired finger.

Both FULLY_OPEN and FULLY_CLOSED are defined
presets within FingerLib.h, and are the maximum and
minimum positions of the motors (50 - 973).

COMPLEX

As detailed above, the array is read by first reading
from the segment relating to the grip, the grip position
is then used to determine the row of finger positions
to use, however the actual function of this grip control
is slightly more complex.

	// FIST						
	//COUNT	F0	F1	F2	F3	F4	
Row 0	{0,	FULLY_OPEN,	FULLY_OPEN,	FULLY_OPEN,	FULLY_OPEN,	FULLY_OPEN,	//OPEN
Row 1	{10,	750,	BLANK,	BLANK,	BLANK,	BLANK},	//MIDPOINTA
Row 2	{40,	750,	BLANK,	BLANK,	BLANK,	BLANK},	//MIDPOINTB
Row 3	{100,	FULLY_CLOSED,	FULLY_CLOSED,	FULLY_CLOSED,	FULLY_CLOSED,	FULLY_CLOSED},	//CLOSED

The segment of the array that applies to fist grip

The function 'gripMovement(int grip position)' starts by looks through the COUNT elements of the desired grip pattern, and searches for two elements that the grip position is between. For example, in the above image, a grip position of 5 would be between COUNTA = 0 (Row 0) and COUNTB = 10 (Row 1), and a grip position of 50 would be between a position of COUNTA = 40 and COUNTB = 100.

After determining the two count values either side of the grip position (COUNTA and COUNTB), the position of the finger is determined by reading the finger position along the same row as the COUNT values. If the finger position reading (POSA) along the COUNTA row is read as BLANK, then the finger position from the row above is used, and if the finger position reading (POSB) along the COUNTB row is read as BLANK, then the finger position from the row below is used. The grip position (0 - 100) is then mapped from the range of COUNTA - COUNTB, to POSA - POSB, which calculates the finger position values between the ones in the array.

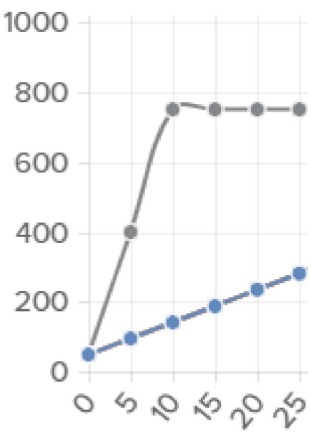
The Fist grip is designed so that when closing, the thumb moves across quickly to oppose the fingers, it then stays in the opposing position for a bit whilst the fingers close further, when the grip gets to a certain position, the thumb then continues closing to apply similar force to the opposing fingers. The graph below shows the how the positions of the fingers change (Y-axis) as the grip position is changed (X-axis) in Fist grip.

Using the Fist grip
above as an example,
when the grip position
0, we can generate the
COUNT and POS values

FIST GRIP MOVEMENT

for each finger, as detailed below;

- FINGER0
 - COUNT A = 0
 - COUNTB = 10
 - POSA = FULLY_OPE N (50)
 - POSB = 750



F0 F1
F2 F3
F4

- FINGER1 to FINGER4
 - COUNTA = 0
 - COUNTB = 100
 - POSA = FULLY_O PEN (50)
 - POSB = FULLY_CLO SED (973)

Note that F1 to F4 are all represented by F4

Note that for FINGER1 to FINGER4, the finger positions for both Row 1 and Row 2 are BLANK, therefore COUNTB = 100.

When the grip position of 0 is mapped using the above values, the target position of each finger is calculated as

Grip Pos	F0	F1	F2	F3	F4
0	50	50	50	50	50
5	400	96	96	96	96
10	750	142	142	142	142
15	750	188	188	188	188
20	750	235	235	235	235
25	750	281	281	281	281
30	750	327	327	327	327
35	750	373	373	373	373
40	750	419	419	419	419
45	769	465	465	465	465
50	787	512	512	512	512
55	806	558	558	558	558
60	824	604	604	604	604
65	843	650	650	650	650
70	862	696	696	696	696

50. When the grip position is 5, the COUNT and POS values are still the same, but the mapped values are as follows;

75	880	742	742	742	742
80	899	788	788	788	788
85	917	835	835	835	835
90	936	881	881	881	881
95	954	927	927	927	927
100	973	973	973	973	973

- FINGER0 = 400
(almost halfway closed)
- FINGER1 to FINGER 4 = 96 (slightly closed)

When the grip position is at 25, the COUNT and POS values are different, but produce the following results;

- FINGER0 = 750 (3/4 closed)
- FINGER1 to FINGER 4 = 235 (1/4 closed)

We can see from the graph that fingers 0 - 4 move in a linear motion, whilst finger 0 moves quickly from 50 to 750, and remains at position 750 for a while, it then moves slowly from 750 to 973.

In Tutorial Tags Active Tutorial

Share

8 Likes

← Artichoke V1.1 Firmware
User Guide

Ada V1.1 Printing
Instructions →

[Home](#) [Shop](#) [Returns](#) [Investors](#)
[Developers](#) [Downloads](#) [Tutorials](#)
[Blog](#) [Contact Us](#)

© Open Bionics 2018