Programming a Pivot-Wheel Drive

Clem McKown – Team 1640
9-August-2009

4-Wheel Independent Pivot-Wheel Drive describes a 4wd drive-train in which each of the (4) wheels are independently driven and may be independently pivoted for steering purposes. The design offers the potential for excellent drive-train performance and a solution to conventional (tank) drive-train design constraints. The design also brings some clear design and control challenges.

This arrangement provides the possibility to operate in several different modes:

1. Crab mode – pivots all wheels together and at common speed to steer the robot in any direction on the 2-d playing surface (true 2-d drive). The mode does not control chassis orientation.
2. Snake mode – pivots front and rear wheels in opposite directions to guide the robot through a turn
    a. X-bias – *drive direction aligned with the long-axis of the robot*
    b. Y-bias – *drive direction aligned with the short-axis of the robot*
3. Automobile mode – pivots front wheels only to guide robot through a turn
    a. X-bias
    b. Y-bias
4. Tank mode – Does not use pivots to steer (but can use pivots to change drive orientation *ala Twitch*). Steering accomplished by differential L & R drive speeds.
    a. X-bias
    b. Y-bias

Some important requirements for the programming are:

- Processing burden associated with steering and drive need be kept to a low level.
- Operation of the robot (driving) should be intuitive in all modes.

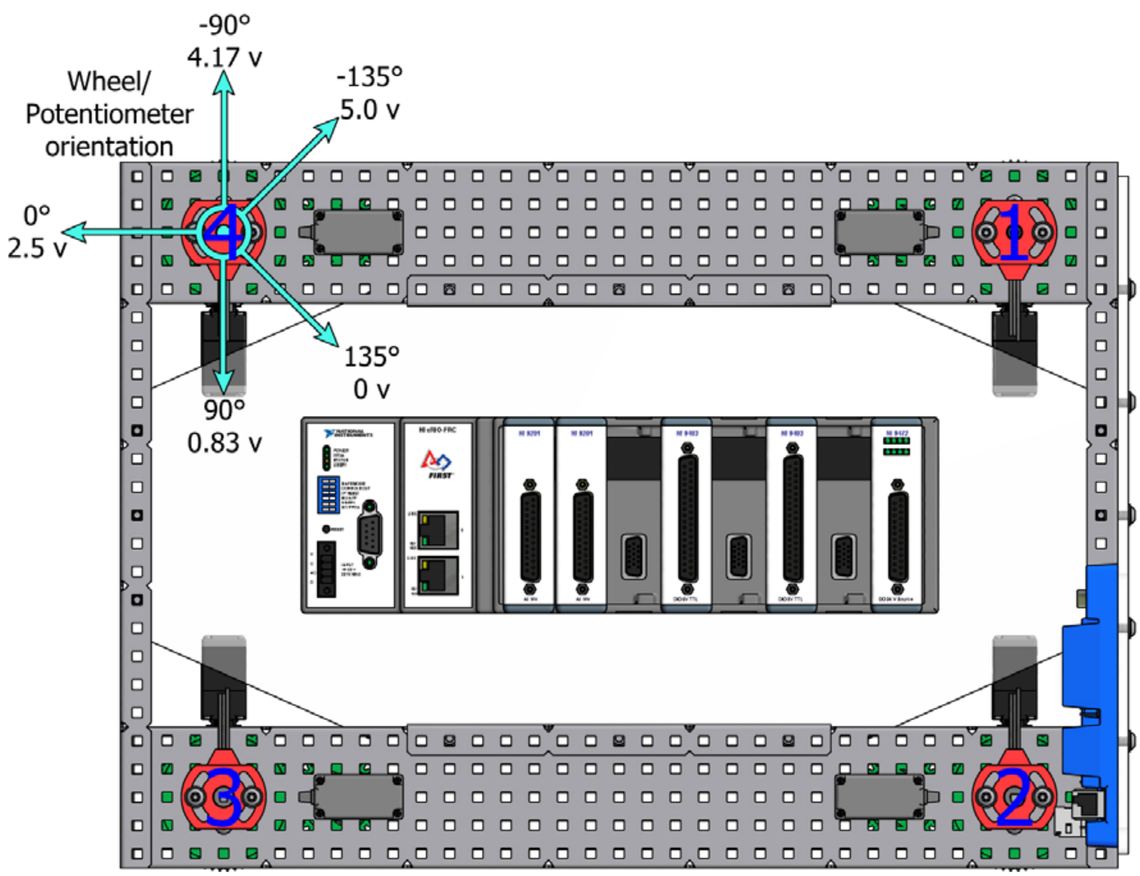
<u>Physical Configuration</u>

Configuration is shown in the following schematic. There are (4) Pivot Wheels with a wheel-base/pivot-base of 14 x 10 inches. Steering Motors are wired into digital sidecar ports 1-4, in the order right-rear, left rear, left front, right front. Drive Motors are wired into digital sidecar parts 5-8 in the same sequence.

There are (4) Potentiometers, one for each wheel pivot shaft. These are wired into analogue inputs 1-4 in the same sequence as the Steering Motors.

So, from there are clearly identified Steering & Drive Motors and Potentiometers 1, 2, 3 & 4. These will be consistently identified as such in this document. These identifications are physical. Changing the axis or direction of drive, or direction of turn does not change identities. These are marked in the following schematic.

Potentiometers should be set to mid-range (2.5 v or 512) when the wheels are aligned with the robot's long axis. Potentiometers need to be commonly oriented so that clockwise or counterclockwise pivot provides a Potentiometer output change in the same direction. The logic presented here assumes a clockwise pivot increases value (but either direction is okay – consistency is necessary).



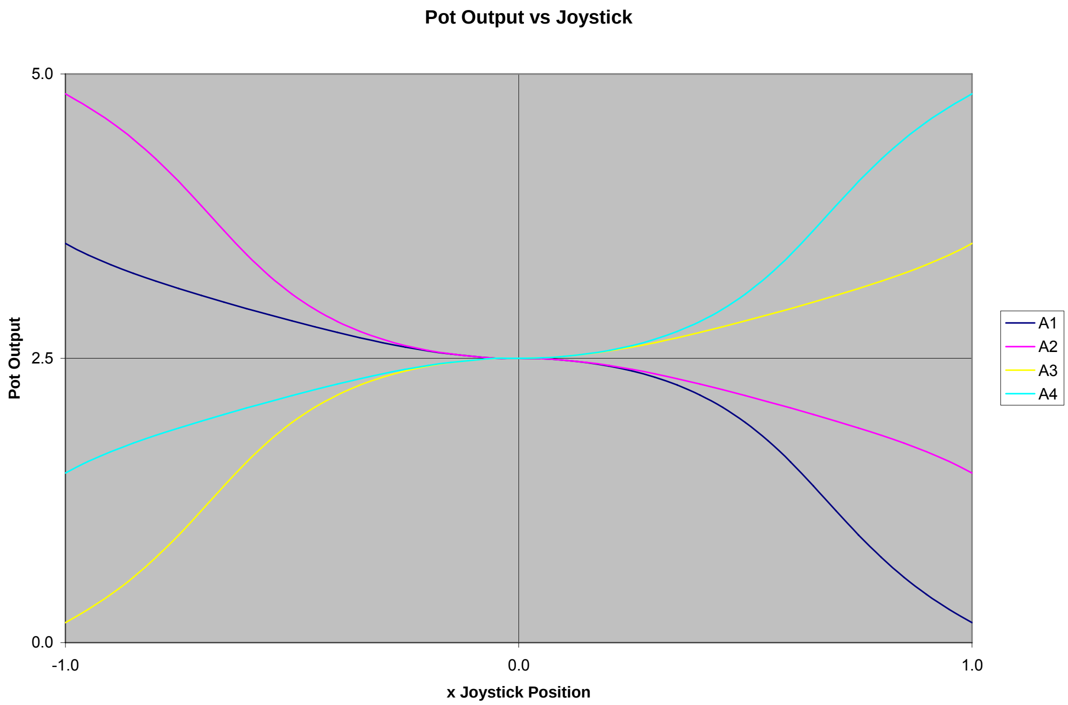
Potentiometers are limited to a 270° arc of control. This places some important constraints on our approach. Snake, Automobile and Tank modes can be satisfactorily controlled with Potentiometers (but there's a hitch with biaxial drive, which will be discussed later). Crab mode will be limited with Potentiometers.

<u>Mapping to the Potentiometers</u>

Potentiometer range is 0-5 volts (alternatively 0-1024). Potentiometers should be positioned and calibrated so that the output is mid-range (2.5 volts or 512) when pivots are aligned with the robot long axis (x-bias). This represents a pivot angle (α) of zero.

Set Potentiometer output to increase with clockwise pivot. Since right-hand rule is used for measuring angles, angles and potentiometer output run in opposite directions. Potentiometer range is nominally 270° ($3\pi/2$ radians – all angles are calculated in radians), so the potentiometer output = $2.50 - 1.06\ \alpha$ (radians) [or = $512 - 217\ \alpha$].

Mapping Potentiometer Output to Joystick x-position for the Snake-mode x-bias:



Note that the (4) Pivot curves reversed and/or inverted versions of one curve.

Mapping in y-bias

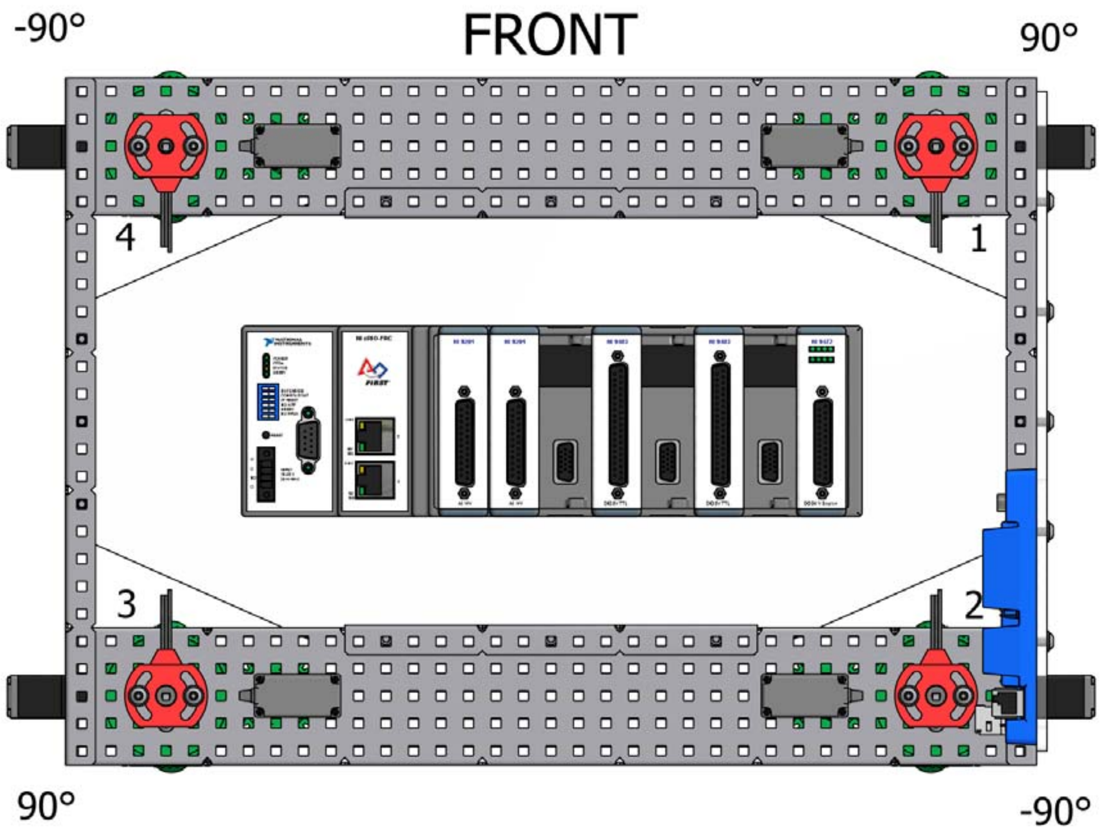
In order to stay within the 270° Potentiometer limits, the pivots need to rotate so that all Drive Motors face outwards in Snake or Automobile Modes y-bias, as shown below.

Twitch used the opposite rotation.

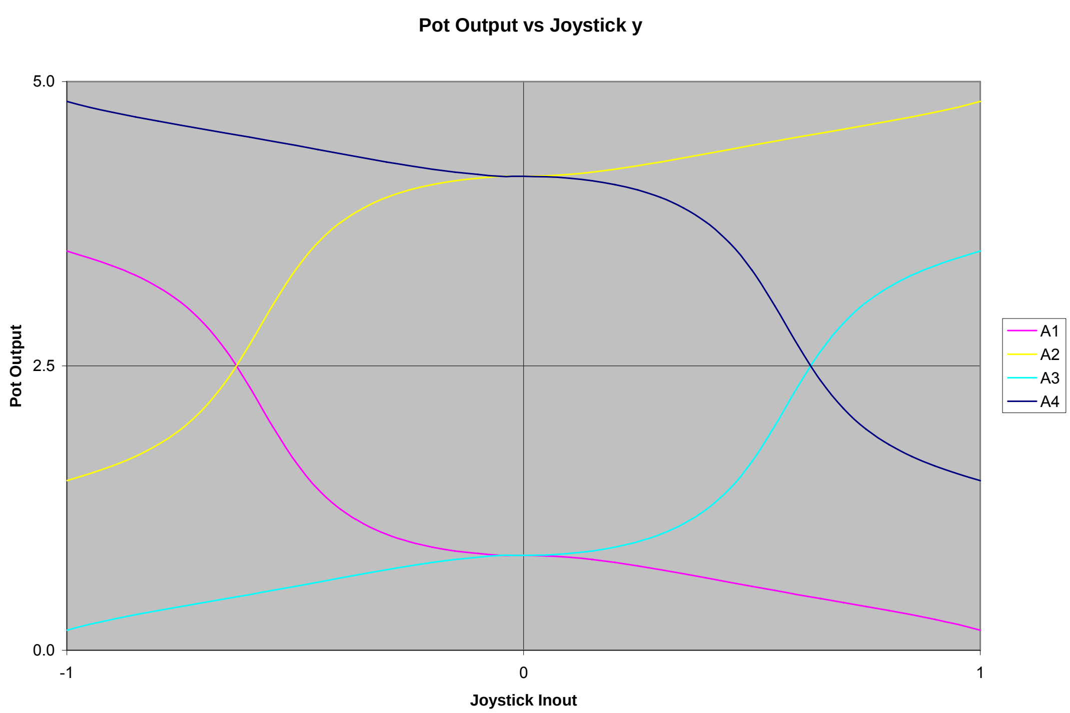
This is okay for this prototype, but is not really viable for an FRC robot.

Encoders would allow the motors to pivot inboard.

For Tank mode, the motors can pivot inboard.



Mapping Potentiometer Output to Joystick x-position for the Snake-mode y-bias:



Snake and Automobile Modes

The calculations used to determine pivot angle and drive speed reductions for Snake and Automobile modes are complex and unsuitable for real-time calculation while driving. Fortunately, the calculated values are dependent on joystick x-position and wheelbase dimensions (*l* & *w*) only. As a result, arrays of Potentiometer targets (setpoints) for a variety of Joystick x-positions may be pre-defined. Fours arrays would be naturally defined for (Snake & Auto) x (x-bias & y-bias).

These arrays could be used in one or two ways. The first is to use these as discrete points and to interpolate when a joystick value falls between two points defined in the array. This method is more accurate and provides a smoother transition of setpoints, but imposes a higher processing burden.

The second, which I use in this document, breaks the x-joystick domain into n + 1 ranges (21 equal-sized ranges in the case developed) and assigns steering setpoints and drive-speed factors to each of these. Real-time calculation is therefore minimized, but the transition from range-to-range could be jerky.

An array can be defined (D_i) where the elements are the potentiometer #1 deviation from center at each of the selected joystick ranges. These values are different for x and y orientations. There are n+1 array elements for each bias. Values below are for potentiometer voltage.

x Joystick Input		*i*	D_i	
		i = 0 to n	Snake	
>=	<	*n = 20*	x	y
-1.00	-0.95	0	1.01	0.66
-0.95	-0.85	1	0.82	0.56
-0.85	-0.75	2	0.68	0.48
-0.75	-0.65	3	0.55	0.41
-0.65	-0.55	4	0.44	0.34
-0.55	-0.45	5	0.33	0.27
-0.45	-0.35	6	0.23	0.20
-0.35	-0.25	7	0.14	0.13
-0.25	-0.15	8	0.06	0.06
-0.15	-0.05	9	0.02	0.02
-0.05	0.05	10	0.00	0.00
0.05	0.15	11	-0.02	-0.02
0.15	0.25	12	-0.07	-0.07
0.25	0.35	13	-0.17	-0.19
0.35	0.45	14	-0.32	-0.40
0.45	0.55	15	-0.56	-0.83
0.55	0.65	16	-0.91	-1.48
0.65	0.75	17	-1.34	-2.04
0.75	0.85	18	-1.75	-2.36
0.85	0.95	19	-2.08	-2.55
0.95	1.00	20	-2.32	-2.68

These can be converted to pivot position setpoints for each of the four wheels using simple addition and subtraction.

Physical Motors can be identified with k (k = 1, 2, 3, 4). Each Physical Motor has a Center Position (CP_k), which changes between x-bias and y-bias drives.

	x-bias	y-bias
CP_1	2.50	0.83
CP_2	2.50	4.17
CP_3	2.50	0.83
CP_4	2.50	4.17

Relative Motor positions can be identified with j (j = 1, 2, 3, 4).

- j = 1 – Rear Right
- j = 2 – Rear Left
- j = 3 – Front Left
- j = 4 – Front Right

Setpoints are calculated for each pivot based on their relative (to drive direction) positions.

$$SP_{Rear\text{-}Rt} = CP_{Rear\text{-}Rt} + D_i \qquad \text{eq. 1}$$
$$SP_{Rear\text{-}Lt} = CP_{Rear\text{-}Lt} - D_{n-i} \qquad \text{eq. 2}$$
$$SP_{Fore\text{-}Lt} = CP_{Fore\text{-}Lt} + D_{n-i} \qquad \text{eq. 3}$$
$$SP_{Fore\text{-}Rt} = CP_{Fore\text{-}Rt} - D_i \qquad \text{eq. 4}$$

A worksheet was written to test the calculations and logic.

Pivot Setpoints calculated from D_i - Snake Mode

l = 14.0 in
w = 10.0 in

Orient: 1

x Joystick Input		*i*	D_i		*k*	1 (*Rear-Rt*)	2 (*Rear-Lft*)	3 (*Fore-Lft*)	4 (*Fore-Rt*)
		i = 0 to n	Snake		CP_k	2.50	2.50	2.50	2.50
>=	<	*n = 20*	x	y	*j*	1	2	3	4
-1.00	-0.95	0	1.01	0.66	1.01	3.51	4.82	0.18	1.49
-0.95	-0.85	1	0.82	0.56	0.82	3.32	4.58	0.42	1.68
-0.85	-0.75	2	0.68	0.48	0.68	3.18	4.25	0.75	1.82
-0.75	-0.65	3	0.55	0.41	0.55	3.05	3.84	1.16	1.95
-0.65	-0.55	4	0.44	0.34	0.44	2.94	3.41	1.59	2.06
-0.55	-0.45	5	0.33	0.27	0.33	2.83	3.06	1.94	2.17
-0.45	-0.35	6	0.23	0.20	0.23	2.73	2.82	2.18	2.27
-0.35	-0.25	7	0.14	0.13	0.14	2.64	2.67	2.33	2.36
-0.25	-0.15	8	0.06	0.06	0.06	2.56	2.57	2.43	2.44
-0.15	-0.05	9	0.02	0.02	0.02	2.52	2.52	2.48	2.48
-0.05	0.05	10	0.00	0.00	0.00	2.50	2.50	2.50	2.50
0.05	0.15	11	-0.02	-0.02	-0.02	2.48	2.48	2.52	2.52
0.15	0.25	12	-0.07	-0.07	-0.07	2.43	2.44	2.56	2.57
0.25	0.35	13	-0.17	-0.19	-0.17	2.33	2.36	2.64	2.67
0.35	0.45	14	-0.32	-0.40	-0.32	2.18	2.27	2.73	2.82
0.45	0.55	15	-0.56	-0.83	-0.56	1.94	2.17	2.83	3.06
0.55	0.65	16	-0.91	-1.48	-0.91	1.59	2.06	2.94	3.41
0.65	0.75	17	-1.34	-2.04	-1.34	1.16	1.95	3.05	3.84
0.75	0.85	18	-1.75	-2.36	-1.75	0.75	1.82	3.18	4.25
0.85	0.95	19	-2.08	-2.55	-2.08	0.42	1.68	3.32	4.58
0.95	1.00	20	-2.32	-2.68	-2.32	0.18	1.49	3.51	4.82

The variable *Orientation* (which may be set to1, 2, 3 or 4) determines which side of the robot is the front (NESW, respectively).

The variable *LFT* sets universal motor direction (may be set to 1 or -1).

Pivots will need to be actively driven to the set positions and maintained there until the setpoint is changed. PID (Proportional – Integral – Derivative) control should be suitable.

Likewise, motor speed factors, accounting for both turn radius and motor rotation direction, are calculated from another array, V_i. There are n/2+1 elements in the V_i array for each bias (because it mirrors). Logic checks whether wheel is on inside-of-turn and applies speed reduction if true. Motor direction (+/-) is based relative Left/Right side and x versus y bias.

Motor Speed Factors from V_i - Snake Mode

l = 14.0 in Orient: 1

w = 10.0 in

x Joystick Input		*i*	V_i		*k*	Motor Speed 1	2	3	4
		i = 0 to n	Snake		CP_k	*Rear-Rt*	*Rear-Lft*	*Fore-Lft*	*Fore-Rt*
>=	<	*n = 20*	x	y	*j*	1	2	3	4
-1.00	-0.95	0	1.00	1.00	1.00	1.00	-1.00	-1.00	1.00
-0.95	-0.85	1	0.76	0.75	0.76	1.00	-0.76	-0.76	1.00
-0.85	-0.75	2	0.60	0.56	0.60	1.00	-0.60	-0.60	1.00
-0.75	-0.65	3	0.52	0.41	0.52	1.00	-0.52	-0.52	1.00
-0.65	-0.55	4	0.53	0.32	0.53	1.00	-0.53	-0.53	1.00
-0.55	-0.45	5	0.60	0.36	0.60	1.00	-0.60	-0.60	1.00
-0.45	-0.35	6	0.71	0.50	0.71	1.00	-0.71	-0.71	1.00
-0.35	-0.25	7	0.82	0.67	0.82	1.00	-0.82	-0.82	1.00
-0.25	-0.15	8	0.91	0.84	0.91	1.00	-0.91	-0.91	1.00
-0.15	-0.05	9	0.98	0.96	0.98	1.00	-0.98	-0.98	1.00
-0.05	0.05	10	1.00	1.00	1.00	1.00	-1.00	-1.00	1.00
0.05	0.15	11			0.98	0.98	-1.00	-1.00	0.98
0.15	0.25	12			0.91	0.91	-1.00	-1.00	0.91
0.25	0.35	13			0.82	0.82	-1.00	-1.00	0.82
0.35	0.45	14			0.71	0.71	-1.00	-1.00	0.71
0.45	0.55	15			0.60	0.60	-1.00	-1.00	0.60
0.55	0.65	16			0.53	0.53	-1.00	-1.00	0.53
0.65	0.75	17			0.52	0.52	-1.00	-1.00	0.52
0.75	0.85	18			0.60	0.60	-1.00	-1.00	0.60
0.85	0.95	19			0.76	0.76	-1.00	-1.00	0.76
0.95	1.00	20			1.00	1.00	-1.00	-1.00	1.00

Data Array
Selected data
User Settable

Two worksheets are posted with this document, providing calculations and logic for Snake, Automobile & Crab Modes. One is based on potentiometer output measured in volts (0-5), the other in counts (0-1024). Other than the potentiometer convention, the two worksheets are equivalent.

Automobile control is similar to Snake. k & j assignments are the same and values of CP_k for the various Orientations are identical to Snake's. The D_i array values are different for Automobile, but equations 3 & 4 are still used to calculate the front pivot setpoints. Automobile rear pivot setpoints are fixed at the CP_k positions and do not change with joystick position.

Drive motor control logic is somewhat more complex with Automobile Mode since in a turn, all motors except for the outside-front motor need to be reduced in speed and by different amounts. Same basic approach as Snake's, just more specific than answering the simple *"inside-of-turn?"* question. The V_i array for Automobile is three times larger than Snake's. The motor direction logic is identical to Snake's.

For data arrays and details on the logic, see the **Pivot Calcs Automobile** page of either Pivot Practical (0-5vdc).xls or Pivot Practical (0-1024).xls.

Crab Mode

Crab Mode is completely different. It provides true 2-d maneuverability. In many ways, it is simpler than Snake & Automobile. It also has its own challenges.

Our Prototype can only approximate a real Crab drive. There are two reasons for this: 1) the drive motors are on the pivots and rotate with them and their wires would fail if the pivot turns too far; and 2) the potentiometers can only monitor pivot orientation over an arc of 270°. These two constraints are not additive (they are more complimentary).

On the other hand, the purpose of this prototype is to gain experience into the performance and control of a Pivot-Wheel robot; to gain the most knowledge and insight for the least cost and time. These limitations are therefore reasonable and acceptable.

From a calculation standpoint, Pivot Angle setpoint is based on the arctangent of the x & y joystick positions. The joystick space was mapped into quadrants (to determine whether arctan x/y or arctan y/x is used).

For the front quadrant ($y \geq 0$ and $|y| > |x|$), $\alpha = -\tan^{-1}(x/y)$.

For the left quadrant ($x < 0$ and $|x| \geq |y|$), $\alpha = \tan^{-1}(y/x) + \pi/2$.

For the right quadrant ($x > 0$ and $|x| \geq |y|$), $\alpha = \tan^{-1}(y/x) - \pi/2$.

The rear quadrant ($y < 0$ and $|y| > |x|$) is calculable, but not useful.

The mapping of joystick position to potentiometer output is provided below.

Pivot Setpoints - Crab Mode

Angle - Voltage Conversion		
α	v	
-2.36	5.00	v = 2.5 + -1.06 α
2.36	0.00	

Pot SP volts	Joystick x																				
Joystick y	-1.00	-0.90	-0.80	-0.70	-0.60	-0.50	-0.40	-0.30	-0.20	-0.10	0.00	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
1.00	1.67	1.72	1.78	1.85	1.93	2.01	2.10	2.19	2.29	2.39	2.50	2.61	2.71	2.81	2.90	2.99	3.07	3.15	3.22	3.28	3.33
0.90	1.61	1.67	1.73	1.80	1.88	1.96	2.06	2.16	2.27	2.38	2.50	2.62	2.73	2.84	2.94	3.04	3.12	3.20	3.27	3.33	3.39
0.80	1.55	1.60	1.67	1.74	1.82	1.91	2.01	2.12	2.24	2.37	2.50	2.63	2.76	2.88	2.99	3.09	3.18	3.26	3.33	3.40	3.45
0.70	1.48	1.53	1.60	1.67	1.75	1.84	1.95	2.07	2.20	2.35	2.50	2.65	2.80	2.93	3.05	3.16	3.25	3.33	3.40	3.47	3.52
0.60	1.41	1.46	1.52	1.59	1.67	1.76	1.88	2.01	2.16	2.32	2.50	2.68	2.84	2.99	3.12	3.24	3.33	3.41	3.48	3.54	3.59
0.50	1.33	1.37	1.43	1.49	1.57	1.67	1.78	1.93	2.10	2.29	2.50	2.71	2.90	3.07	3.22	3.33	3.43	3.51	3.57	3.63	3.67
0.40	1.24	1.28	1.33	1.38	1.46	1.55	1.67	1.82	2.01	2.24	2.50	2.76	2.99	3.18	3.33	3.45	3.54	3.62	3.67	3.72	3.76
0.30	1.14	1.17	1.21	1.26	1.33	1.41	1.52	1.67	1.88	2.16	2.50	2.84	3.12	3.33	3.48	3.59	3.67	3.74	3.79	3.83	3.86
0.20	1.04	1.07	1.09	1.13	1.17	1.24	1.33	1.46	1.67	2.01	2.50	2.99	3.33	3.54	3.67	3.76	3.83	3.87	3.91	3.93	3.96
0.10	0.94	0.95	0.97	0.98	1.01	1.04	1.09	1.17	1.33	1.67	2.50	3.33	3.67	3.83	3.91	3.96	3.99	4.02	4.03	4.05	4.06
0.00	0.83	0.83	0.83	0.83	0.83	0.83	0.83	0.83	0.83	0.83	2.50	4.17	4.17	4.17	4.17	4.17	4.17	4.17	4.17	4.17	4.17
-0.10	0.73	0.72	0.70	0.68	0.66	0.62	0.57	0.49	0.34	0.00		5.00	4.66	4.51	4.43	4.38	4.34	4.32	4.30	4.28	4.27
-0.20	0.62	0.60	0.57	0.54	0.49	0.43	0.34	0.21	0.00				5.00	4.79	4.66	4.57	4.51	4.46	4.43	4.40	4.38
-0.30	0.52	0.49	0.45	0.40	0.34	0.26	0.15	0.00						5.00	4.85	4.74	4.66	4.60	4.55	4.51	4.48
-0.40	0.43	0.39	0.34	0.28	0.21	0.12	0.00								5.00	4.88	4.79	4.72	4.66	4.61	4.57
-0.50	0.34	0.30	0.24	0.18	0.10	0.00										5.00	4.90	4.82	4.76	4.70	4.66
-0.60	0.26	0.21	0.15	0.08	0.00												5.00	4.92	4.85	4.79	4.74
-0.70	0.19	0.13	0.07	0.00														5.00	4.93	4.87	4.81
-0.80	0.12	0.06	0.00																5.00	4.94	4.88
-0.90	0.06	0.00																		5.00	4.94
-1.00	0.00																				5.00

Naturally, pivot setpoints for all (4) wheels are the same.

Drive motor speed; simplest approach is to use max(|x|,|y|) for the drive speed setpoint. All (4) drive motor speeds are the same. The calculated worksheet:

Drive fractional	Joystick x																				
Joystick y	-1.00	-0.90	-0.80	-0.70	-0.60	-0.50	-0.40	-0.30	-0.20	-0.10	0.00	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
0.90	1.00	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	1.00
0.80	1.00	0.90	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.90	1.00
0.70	1.00	0.90	0.80	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.80	0.90	1.00
0.60	1.00	0.90	0.80	0.70	0.60	0.60	0.60	0.60	0.60	0.60	0.60	0.60	0.60	0.60	0.60	0.60	0.60	0.70	0.80	0.90	1.00
0.50	1.00	0.90	0.80	0.70	0.60	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.60	0.70	0.80	0.90	1.00
0.40	1.00	0.90	0.80	0.70	0.60	0.50	0.40	0.40	0.40	0.40	0.40	0.40	0.40	0.40	0.40	0.50	0.60	0.70	0.80	0.90	1.00
0.30	1.00	0.90	0.80	0.70	0.60	0.50	0.40	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
0.20	1.00	0.90	0.80	0.70	0.60	0.50	0.40	0.30	0.20	0.20	0.20	0.20	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
0.10	1.00	0.90	0.80	0.70	0.60	0.50	0.40	0.30	0.20	0.10	0.10	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
0.00	1.00	0.90	0.80	0.70	0.60	0.50	0.40	0.30	0.20	0.10	0.00	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
-0.10	1.00	0.90	0.80	0.70	0.60	0.50	0.40	0.30	0.20	0.10		0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
-0.20	1.00	0.90	0.80	0.70	0.60	0.50	0.40	0.30	0.20				0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
-0.30	1.00	0.90	0.80	0.70	0.60	0.50	0.40	0.30						0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
-0.40	1.00	0.90	0.80	0.70	0.60	0.50	0.40								0.40	0.50	0.60	0.70	0.80	0.90	1.00
-0.50	1.00	0.90	0.80	0.70	0.60	0.50										0.50	0.60	0.70	0.80	0.90	1.00
-0.60	1.00	0.90	0.80	0.70	0.60												0.60	0.70	0.80	0.90	1.00
-0.70	1.00	0.90	0.80	0.70														0.70	0.80	0.90	1.00
-0.80	1.00	0.90	0.80																0.80	0.90	1.00
-0.90	1.00	0.90																		0.90	1.00
-1.00	1.00																				1.00