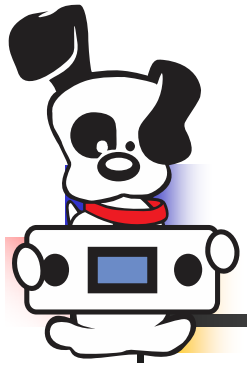
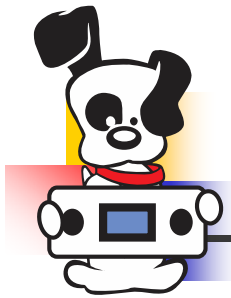


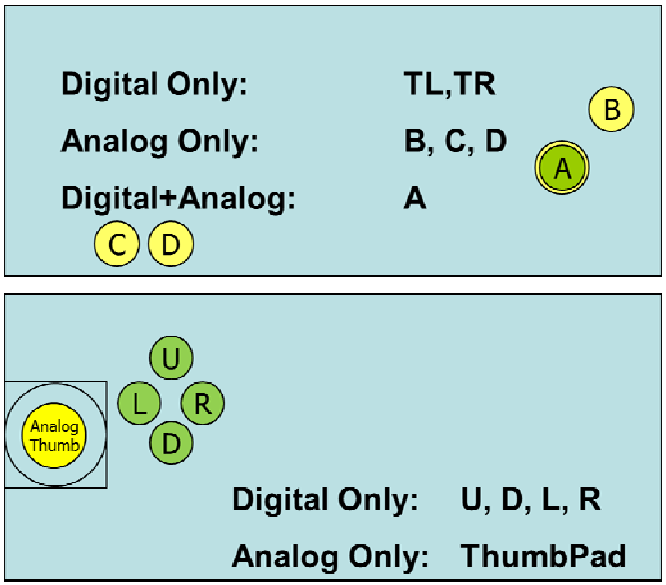
KodeKLIX for PUP

Analog Inputs



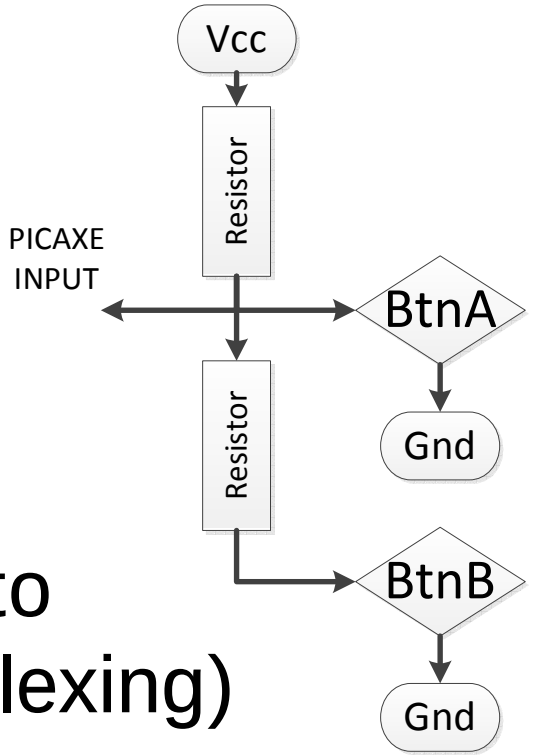
PUP Analog Inputs

- Analog sources include
  - Button inputs A, B, C, D, TR, TL
  - Directional inputs via the ThumbPad (where installed)
- Pre-defines exist:
  - _MultiBtn, _BtnADC
  - _PadX, _PadY _padADCx, _padADCy



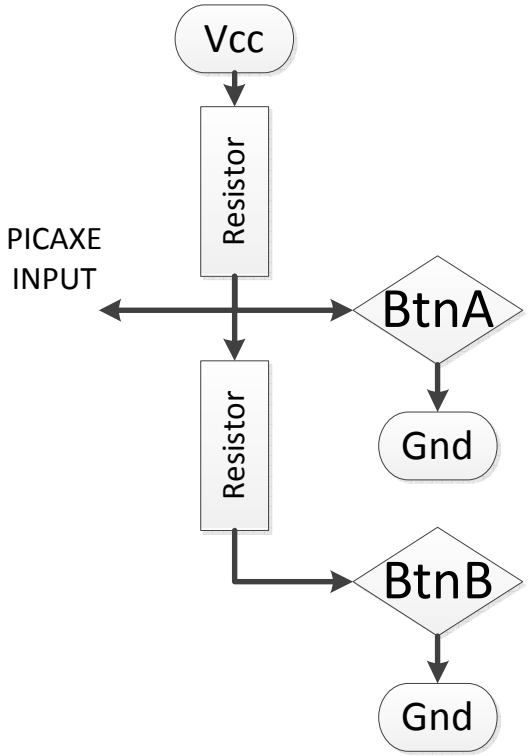
Programming Analog Inputs #1

- Analog inputs are not white or black, but shades of grey
- This means they have a value that ranges between min and max
  - Typically 0 < value < 255
- PUP uses a resistor “ladder” to connect multiple buttons to the one PICAXE input (muliplexing)
  - This saves on input pins needed



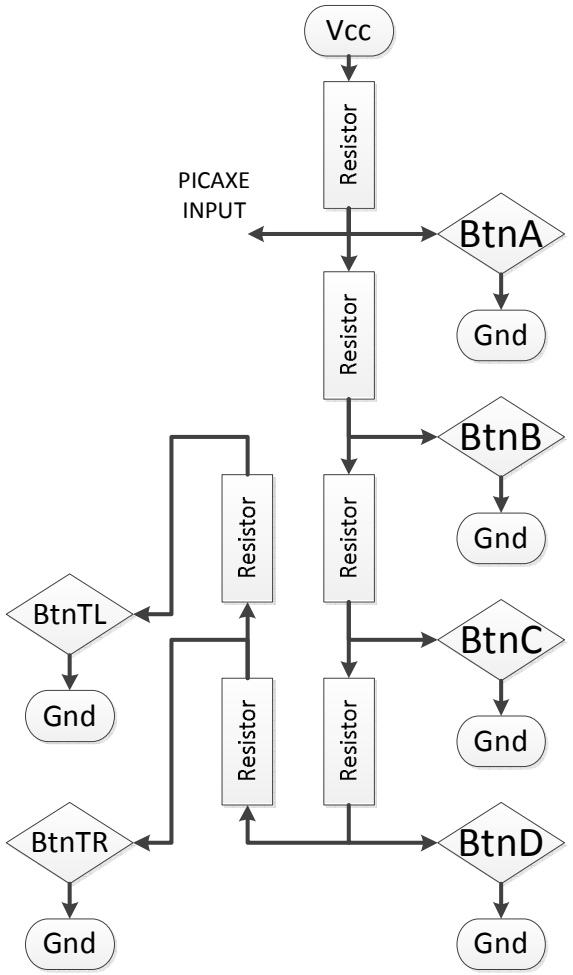
Programming Analog Inputs #2

- In the two button example
  - When no button is pressed, PICAXE INPUT = Vcc (1)
  - When BtnA is pressed, PICAXE INPUT = Gnd (0)
  - When BtnB is pressed, PICAXE INPUT is ½ way between Vcc and Gnd
    - On the
      0 < value < 255 scale
      PICAXE INPUT is ~128



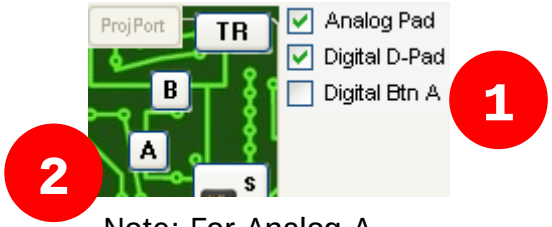
Programming Analog Inputs #3

- The actual PUP button ladder is a little more “crowded” with at least 6 buttons trying to have their say!
- There is a hierarchy:
  A <B <C <D <TR <TL
- Note: TL may not be fitted on PUPs installed in an iPhone case



Programming Analog Inputs #4

- To write code for an analog input, select the button from the coding interface

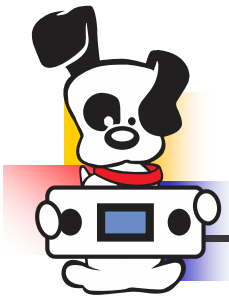


Note: For Analog-A,
1) ☐ option, then 2) press 'A'

- Copy *default* ➔ *custom* snippet
- Complete the application specific code as indicated
  - Analog inputs are read using the PICAXE's ADC hardware

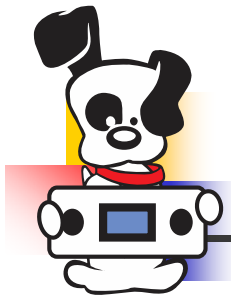
```
'=========================
' Multiplexed Button D
'=========================
{
if _btnADC > _MuxD and _btnADC < _MuxTR then
          ' end game, but keep score
' User code...


          endif
}
```



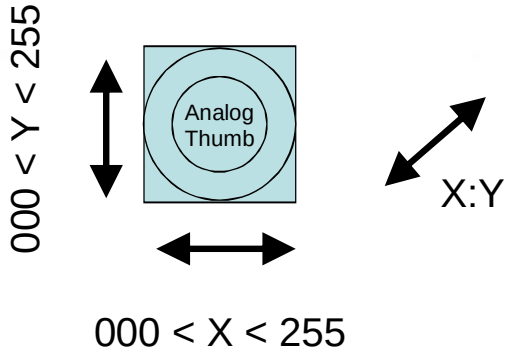
Programming Analog Inputs #5

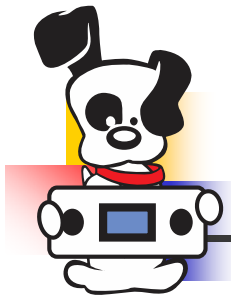
- The ADC value of a button will be in a range specified by boundaries, fortunately these are predefined for you by the API
  - *BtnA* < _MuxA
  - _MuxB < *BtnB* < _MuxC
  - _MuxC < *BtnC* < _MuxD
  - _MuxD < *BtnD* < _MuxTR
  - _MuxTR < *BtnTR* < _MuxTL
  - _MuxTL < *BtnTL* < _MuxMx



Programming Analog Inputs #6

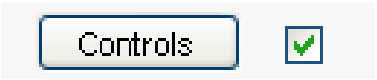
- The analog ThumbPad reports values on two axes; _padADCx and _padADCy
- In theory these return values 0-255, but in practice range is more like 40-210
- The same pins are used for D-Pad inputs Up and Down
  - Their analog value will be 0 (pressed) or 255 (unpressed)





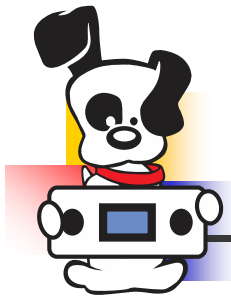
Programming Analog Inputs #7

- To use the code you need to enable the button by including its code...
  - Enable the “controls” routine



  - By default, this enables all inputs
  - Advanced users can reduce code space by deleting #macros for buttons not needed in the application
- For analog routines to work you need to include setups

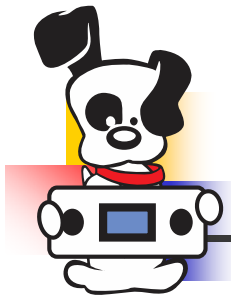
```
' ================================================
' Sequence in which control buttons are processed.
' Delete any which are not to be used by your app.
' ================================================
' Analog Thumbpad controls (if installed)
' ================================================
#padADC
#padLF
#padRT
#padUP
#padDN
' ================================================
' D-pad controls (if installed)
' ================================================
#buttonUP    ' read as analog
#buttonDN    ' read as analog
#buttonLF
```



Programming Analog Inputs #8

- #macros for buttons include two special ADC setup commands
  - #padADC for the analog ThumbPad
  - #buttonADC for the multiplexed buttons

```
' ================================================
' Sequence in which control buttons are processed.
' Delete any which are not to be used by your app.
' ================================================
' Analog Thumbpad controls (if installed)
' ================================================
#padADC
#padLF
#padRT
#padUP
#padDN
' ================================================
' D-pad controls (if installed)
' ================================================
#buttonUP   ' read as analog
#buttonDN   ' read as analog
#buttonLF
#buttonRT
' ================================================
#buttonDA   ' Digital button A
' ================================================
#buttonADC  ' required for multiplexed buttons
#buttonA    ' mux A
#buttonB    ' mux B
#buttonC    ' mux C
#buttonD    ' mux D
#buttonTL   ' mux Trigger Left (if installed)
#buttonTR   ' mux Trigger Right
' ================================================
```



Tutorial: 3.3

- Open tutorial 3.3
- Study code snippets for:
  - Control
  - Button B
  - Analog ThumbPad
- Connect PUP and Download program
- What happens when you press?
  - B _____
  - ThumbPad Left, Right _____