

Vbuddy functions for Testbech

```
// ---- Vbuddy user functions

// Open Vbuddy device, port path specified in vbuddy.cfg
//      ... return 1 if successful, else return <0 if fail
int vbdOpen();

// Close an opened Vbuddy device
void vbdClose();

// Clear the TFT screen to black
void vbdClear();

// Display 4-bit binary value in v on a 7 segment display
//      ... digit is from 1 (right-most) to 5 (left most)
void vbdHex(int digit, int v);

// Plot y value scaled between min and max on TFT screen on next x coord.
//      ... When x reaches 240, screen is cleared and x starts from 0 again.
void vbdPlot(int y, int min, int max);

// Write header at top of TFT, centre justified
void vbdHeader(const char* header);

// Report the cycle count on bottom right of TFT screen
void vbdCycle(int cycle);

// Return current Flag value
bool vbdFlag();

// Set Flag mode: 0 - toggle, 1 - one-shot
void vbdSetMode (int mode);

// Return parameter value on Vbuddy set by the rotary encoder
int vbdValue();

// Initialise DAC output buffer with N samples
void vbdInitAnalogOut(int Nsamp);

// Output a sample to Vbuddy DAC buffer
void vbdOutputSample(int sample);

// Turn analog output ON
void vbdAoutON();

// Turn analog output OFF
void vbdAoutOFF();

// Initialise microphone buffer to capture N samples
void vbdInitMicIn(int Nsamp);
```

```
// get next sample from microphone buffer
int vbdMicValue();

// Return 0 if no key is pressed, otherwise return ASCII code of key
// ... this function is non-blocking and does not actually use Vbuddy
char vbdGetkey();

// Initialise an internal stop watch in msec on Vbuddy
void vbdInitWatch();

// Returns elapsed time in msec since last vbdInitWatch() call
int vbdElapsed();

// ---- End of Vbuddy User Function declarations
```