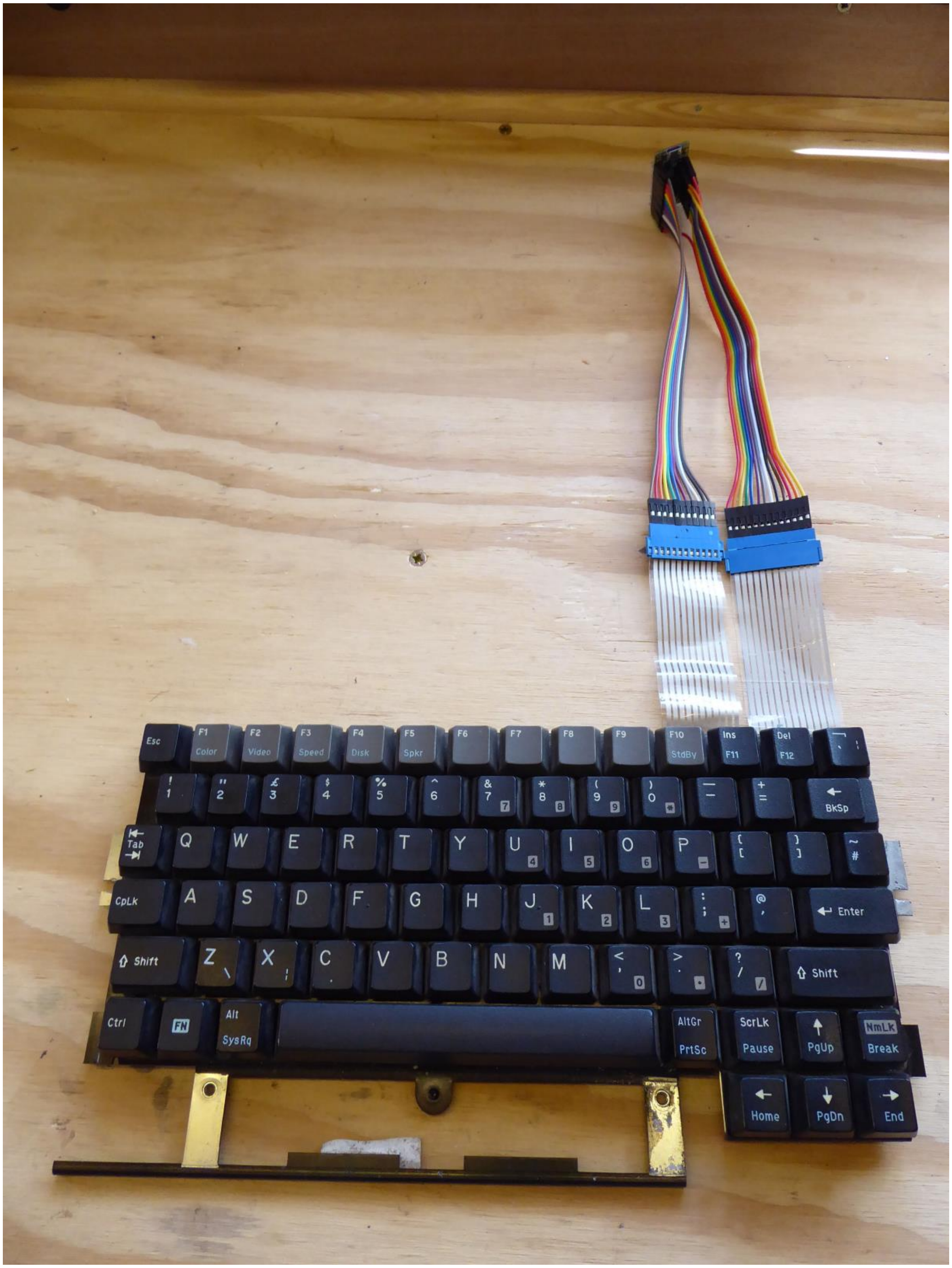
This GRiD 1550 keyboard has a UK layout as shown below. The row and column cables from the key matrix have connectors that accept push pins from a standard ribbon cable. The ribbon cables have been soldered directly to the Teensy I/O pads, making sure to avoid I/O #13 (the LED control).

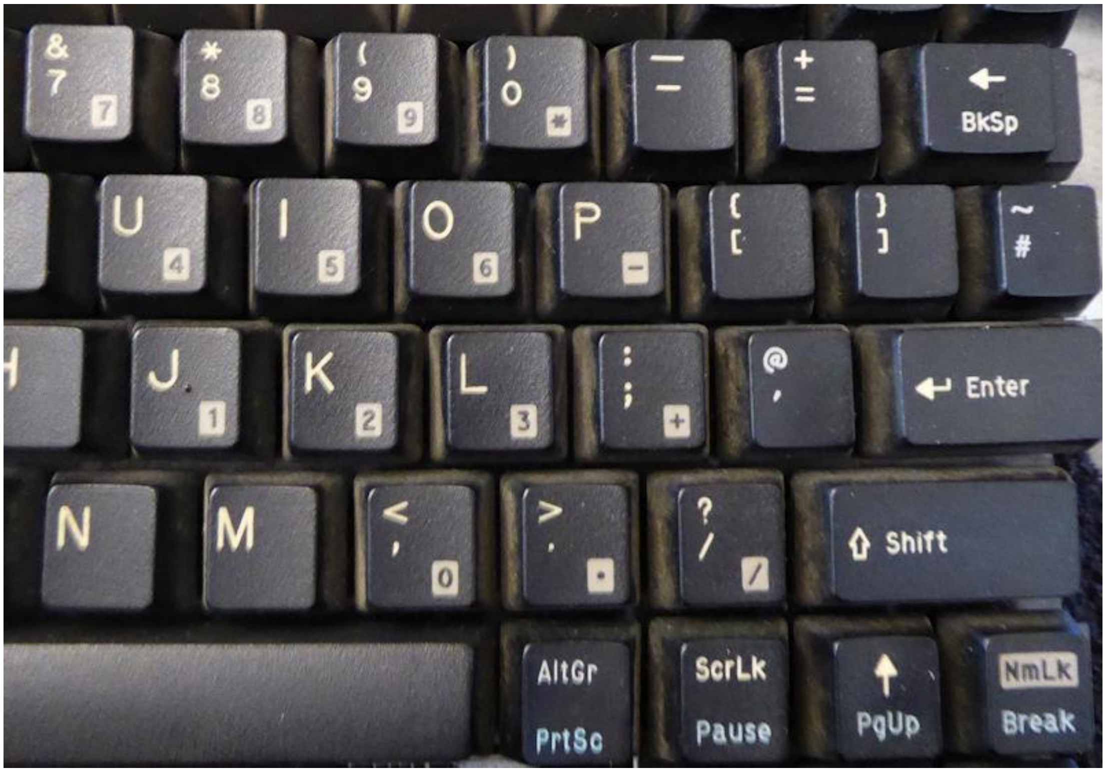


Running the matrix decoder code provided the 11 x 13 key matrix which was converted to the Teensy LC I/O pins, as shown below.

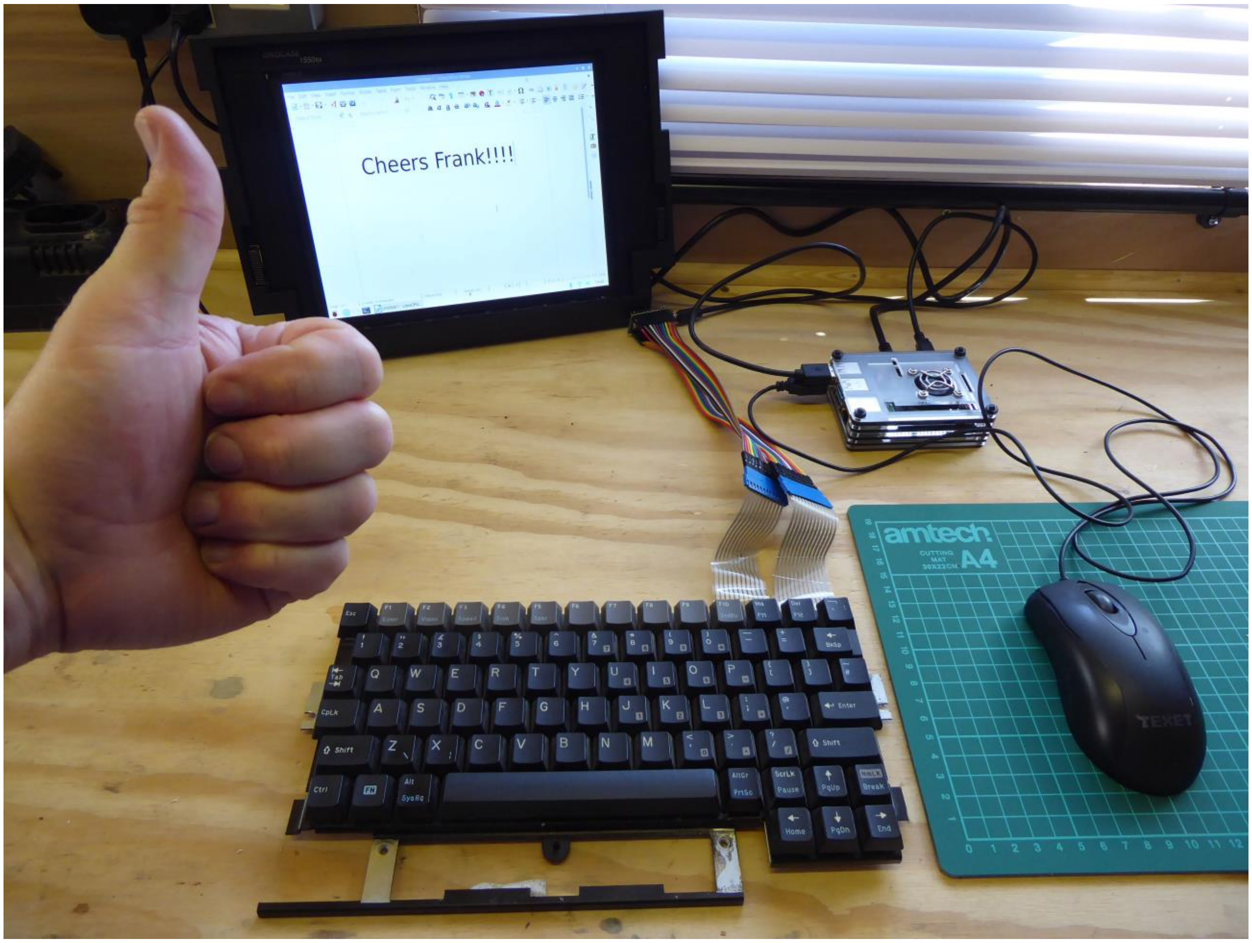
I/O#	23	22	24	21	25	20	19	18	17	16	15	14	12
1						F1	F2	F3	F4	F5		F9	
2	LSHFT	FN	RSHFT		Space								LALT
3						esc	1		F6	F7	F8	F10	
4						tab	Q	W	2	3		6	
5						CAPLK	A	S	E	4	5	7	
6				LEFT		LCNT		Z	F	D	8	9	
7				DOWN		RALT	X	C	G	R	T	0	
8				NUMLK		SCRLK	V	B	H	Y	U	-	
9				TILDE		UP	N	M	J	K	I	INS	
10				BKSLSH		RBRACE	,	BKSP	L	DEL	O	=	
11				RIGHT		ENTER	SLASH	.	“	;	LBRACE	P	

There were some anomalies with this keyboard due to its UK layout. Compiling the Arduino code with the UK keyboard selection allowed most of the keys to function properly. The exception was the “#” key and the “\” key. When a KEY_BACKSLASH is sent from the Teensy to the host, it produces a # so the backslash was placed in the matrix where the # should be. There was no easy way to produce a backslash so the code was modified to watch for a backslash key press and then send a sequence of keys that gives an Alt 92. Using Alt codes is not very pretty but it gets the job done.

The Teensy code was modified to add a numlock matrix that provides a number pad when the NumLock key is selected. The number pad keys are shown below.



Here is the GRiD keyboard & Teensy connected to a single board computer that is driving the GRiD display.



Nicely done Simon!