Using the TTP229/HW-136 touch keypad

I recently purchased the HW-136 keypad, it came with no information so I had a look around and found the datasheet and some code. I had a look at the various versions of code but they looked overly complicated for what should be a simple task. The next stage was to read the datasheet properly, a bit of an uphill struggle interpreting the Chinese English used, but all the info is there. The datasheet indicated that serially collecting the keystrokes should be straightforward.

As supplied the HW-136 will respond to only 8 of its 16 keys providing 8 parallel outputs plus the serial output. There are plenty of linkable options so I will just cover the default state (no links) and collecting the data serially.

With just power connected an in the default state (no links) the TTP229 SDO (Serial Data Output) is in a high state and the SCL (Serial clock) is floating. When a key is pressed the SDO will drop to zero to indicate that there is data to be collected, the clock should then be toggled and the TTP229 will deliver the first key condition to the SDO. This need to be repeated enough times to allow all the bits (8 or 16) to be collected. To achieve this the SDO will need to trigger an interrupt on the Arduino which then needs to generate the clock and recover the data.

The code was written for use on a Nano, SDO is connected to pin D3 which is external interrupt 1, SCL is pin D4 although could be any pin of your choice.

The following code is what I ended up with, not a lot of it. The result of a key press is available in Kbdata (word). Kbdata holds a map of the key pressed where key 1 will show as 0x0001, key 4 as 0x0008 and key 8 as 0x0080. This only the test code so if you want something a bit more sophisticated (16 keys, auto repeat/multiple keys pressed etc.) then that will take a little more coding and some links made.

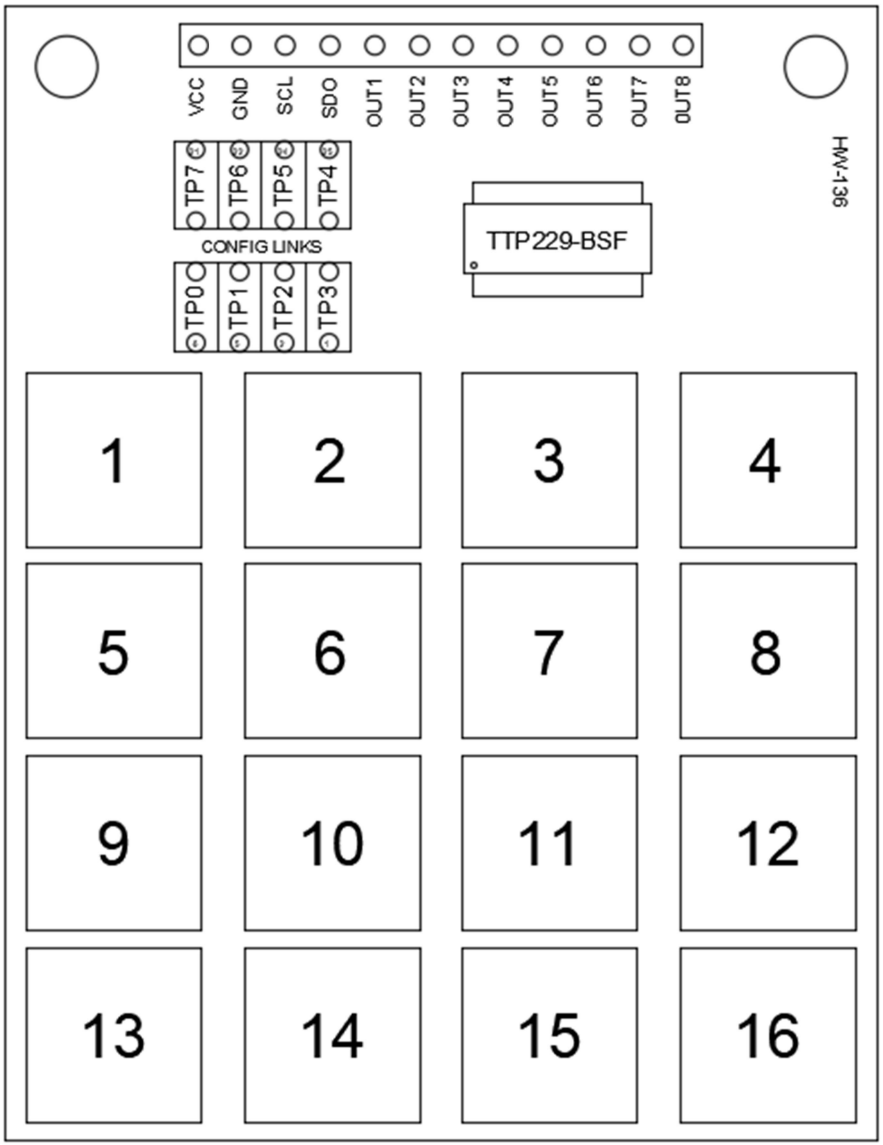
For 16 keys make the TP2 link on the HW-136 (see the picture on last page) and change the NKEYS definition to 16.

To test this code: upload the code onto an Ardino Nano. On the HW-136 connect Vcc to Nano +5V, Gnd to Nano Ground, SDO to the Nano D3 and SCL to Nano D4.

Peter

meterman.accenttechnology@gmail.com

```
#define SDO 3 // data input - triggers int1
#define SCL 4 // clock output
#define ONBOARDLED 13
#define NKEYS 8

word volatile Kbdata = 0; // holds key data

void setup() {
  Serial.begin(9600); // will send key data to the terminal
  if (not Serial)
    ;

  pinMode(SCL, OUTPUT); // clock to keyboard
  digitalWrite(SCL, 1); // clock starting condition

  pinMode(SDO, INPUT);  // pin for data input
  pinMode(ONBOARDLED, OUTPUT); // will show when a key is pressed
  digitalWrite(ONBOARDLED, 0);

  SREG |= 0x80;       // global interrupt enable
  EIMSK = 1 << INT1;  // int 1 enable
  EICRA &= 03;        // interrupt trigger on low level
}

void loop() {
  // put your main code here, to run repeatedly:
}

ISR(INT1_vect) {
  byte len = 0, i = 0;
  unsigned char Buf[25];

  EIMSK = 0;   // int 1 disable
  Kbdata = 0;  // clear the data

// read key bits from the TTP229 (8 or 16)
  for (i = 0; i < NKEYS; i++) {
//  generate clock
    digitalWrite(SCL, 0);
    digitalWrite(SCL, 1);

//  collect bit and assemble
    Kbdata |= digitalRead(SDO) << i;

// fill up the top 8 bits if not in use
  if (NKEYS == 8) {
      Kbdata |= 0xFF00;
    }
  }

// if a key is pressed then output some info
  if (Kbdata != 0xffff) {
    Kbdata ^= 0xFFFF; // invert the key data

    len = sprintf(Buf, "%4.4x\r\n", Kbdata);
    digitalWrite(ONBOARDLED, 1); // led on
    Serial.write(Buf, len);
  } else {
    digitalWrite(ONBOARDLED, 0); // led off
  }

  EIMSK = 1 << INT1;   // int 1 re-enable
}
```



Configuration Links		
TP0	Output type CMOS/OD/OC	See table 1 below
TP1	Output type SCL/SDO	
TP2	Number of Keys 8/16	No link 8 keys, link 16 keys
TP3	Key Action Single/Multiple	See table 2 below
TP4	Key Action Single/Multiple	
TP5	Sample rate in sleep mode	No link 8Hz, Link 64Hz
TP6	Sleep mode sample pulse width	No link 4mS, Link 2mS
TP7	Maximum key down time	No link - Unlimited, Link 80S
Table 1		
	TP0 No Link, TP1 No Link	8 O/P pins CMOS, Serial Active Low
	TP0 Link, TP1 No Link	8 O/P pins Open drain, Serial Active Low
	TP0 No Link, TP1 Link	8 O/P pins CMOS, Serial Active High
	TP0 Link, TP1 Link	8 O/P pins Open Collector, Serial Active High
Table 2		
	TP3 No Link, TP4 No Link	16 Keys in group, single key press
	TP3 Link, TP4 No Link	Group 1 Single Keys, Group 2 Multi Keys
	TP3 No Link, TP4 Link	Group 1 Single Keys, Group 2 Single Keys
	TP3 Link, TP4 Link	16 Keys in group, multi press