

```

// Arduino Uno
// 61 note keyboard scan
// John Harvey 2017

int ManualSelect = 13; // Low to select manual
int RowSelect0 = 10; //
int RowSelect1 = 11; // Drive 74138 A0-2
int RowSelect2 = 12; //
int Column0 = 2;
int Column1 = 3;
int Column2 = 4;
int Column3 = 5;
int Column4 = 6;
int Column5 = 7;
int Column6 = 8;
int Column7 = 9;
byte Command = 0x91; // Note On, channel 1 of 16;
byte Velocity = 0x45; // Arbitrary value, must be > 0 for Note On but not used in Hauptwerk
int KeyboardState[61]; // Zero indexed 0-60; 0 if key on, 1 if key off
int PreviousKeyState = 1;
int CurrentKeyState = 1;

void setup() {
  pinMode (ManualSelect, OUTPUT);
  digitalWrite (ManualSelect, LOW);
  pinMode (RowSelect0, OUTPUT);
  pinMode (RowSelect1, OUTPUT);
  pinMode (RowSelect2, OUTPUT);
  digitalWrite (RowSelect0, LOW); //
  digitalWrite (RowSelect1, LOW); // Initialise to row 0
  digitalWrite (RowSelect2, LOW); //

  pinMode (Column0, INPUT_PULLUP);
  pinMode (Column1, INPUT_PULLUP);
  pinMode (Column2, INPUT_PULLUP);
  pinMode (Column3, INPUT_PULLUP);
  pinMode (Column4, INPUT_PULLUP);
  pinMode (Column5, INPUT_PULLUP);
  pinMode (Column6, INPUT_PULLUP);
  pinMode (Column7, INPUT_PULLUP);

  for (int note = 0; note <= 60; note ++) {
    KeyboardState[note] = 1; // All notes off
  }
  Serial.begin(31250); // Set MIDI baud rate
}

void loop() {

// KeyTest(0, 0, 0x24);
// NoteOn(Command, 0x24, Velocity);
// delay(500);

  for (int row = 0; row <= 7; row ++) {
    for (int col = 0; col <= 7; col ++) {
      int pitch = ((row * 8) + col) + 0x24;
      KeyTest(row, col, pitch);
    }
  }
}

// Check one key for up/down event; key debounce probably not needed, inherent in software
void KeyTest(int r, int c, int p) { // rows 0-7, columns 0-7

  digitalWrite (RowSelect0, r & B00000001);

```

```

digitalWrite (RowSelect1, r & B00000010);
digitalWrite (RowSelect2, r & B00000100);
delay(1); // 1ms settling time, may not be needed

int keyNumber = (r * 8) + c; // 0-63, only 61 will occur

PreviousKeyState=KeyboardState[keyNumber];
CurrentKeyState = digitalRead(c + 2);

if ((PreviousKeyState == 1) && (CurrentKeyState == 0)) { // Note on
  KeyboardState[keyNumber] = 0;
  NoteOn(Command, p, Velocity);
}

if ((PreviousKeyState == 0) && (CurrentKeyState == 1)) { // Note off
  KeyboardState[keyNumber] = 1;
  NoteOn(Command, p, 0x00);
}
}

// Send a MIDI note. Doesn't check to see that cmd is greater than 127, or that data values are
less than 127
void NoteOn(int cmd, int pit, int vel) {
  Serial.write(cmd);
  Serial.write(pit);
  Serial.write(vel);
}
}

```