Truth table:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Binary Input** | | | | |  | **Binary Output** | | | | | | | | | | **Output Hex** |
|  | **A3** | | **A2** | **A1** | **A0** | **Display** | | **idx 6** | **idx 5** | **idx 4** | | **idx 3** | **idx 2** | **idx 1** | | **idx 0** |  | |
| **Hex** | **A** | | **B** | **C** | **D** | **Pattern** | | **g** | **f** | **e** | | **d** | **c** | **b** | | **a** | **0x\_\_** | |
| 0 | 0 | | 0 | 0 | 0 | Display7Seg_0 | | 0 | 1 | 1 | | 1 | 1 | 1 | | 1 | 3F | |
| 1 | 0 | | 0 | 0 | 1 | Display7Seg_1 | | 0 | 0 | 0 | | 0 | 1 | 1 | | 0 | 06 | |
| 2 | 0 | | 0 | 1 | 0 | Display7Seg_2 | | 1 | 0 | 1 | | 1 | 0 | 1 | | 1 | 5B | |
| 3 | 0 | | 0 | 1 | 1 | Display7Seg_3 | | 1 | 0 | 0 | | 1 | 1 | 1 | | 1 | 4F | |
| 4 | 0 | | 1 | 0 | 0 | Display7Seg_4 | | 1 | 1 | 0 | | 0 | 1 | 1 | | 0 | 66 | |
| 5 | 0 | | 1 | 0 | 1 | Display7Seg_5 | | 1 | 1 | 0 | | 1 | 1 | 0 | | 1 | 6D | |
| 6 | 0 | | 1 | 1 | 0 | Display7Seg_6 | | 1 | 1 | 1 | | 1 | 1 | 0 | | 1 | 7D | |
| 7 | 0 | | 1 | 1 | 1 | Display7Seg_7 | | 0 | 0 | 0 | | 0 | 1 | 1 | | 1 | 07 | |
| 8 | 1 | | 0 | 0 | 0 | Display7Seg_8 | | 1 | 1 | 1 | | 1 | 1 | 1 | | 1 | 7F | |
| 9 | 1 | | 0 | 0 | 1 | Display7Seg_9 | | 1 | 1 | 0 | | 0 | 1 | 1 | | 1 | 67 | |
| A | 1 | | 0 | 1 | 0 | Display7Seg_A | | 1 | 1 | 1 | | 0 | 1 | 1 | | 1 | 77 | |
| B | 1 | | 0 | 1 | 1 | Display7Seg_B | | 1 | 1 | 1 | | 1 | 1 | 0 | | 0 | 7C | |
| C | 1 | | 1 | 0 | 0 | Display7Seg_C | | 0 | 1 | 1 | | 1 | 0 | 0 | | 1 | 39 | |
| D | 1 | | 1 | 0 | 1 | Display7Seg_D | | 1 | 0 | 1 | | 1 | 1 | 1 | | 0 | 5E | |
| E | 1 | | 1 | 1 | 0 | Display7Seg_E | | 1 | 1 | 1 | | 1 | 0 | 0 | | 1 | 79 | |
| F | 1 | | 1 | 1 | 1 | Display7Seg_F | | 1 | 1 | 1 | | 0 | 0 | 0 | | 1 | 71 | |
| GPIO index | | | | | GPIO Port 4 name | | | | Resistor (select 220Ω or wire) | | | | 7-segment device pin number | | | 7-segment pin name |
| 0 | | | | | 4.0 | | | | 220Ω | | | | 7 | | | a |
| 1 | | | | | 4.1 | | | | 220Ω | | | | 6 | | | b |
| 2 | | | | | 4.2 | | | | 220Ω | | | | 4 | | | c |
| 3 | | | | | 4.3 | | | | 220Ω | | | | 2 | | | d |
| 4 | | | | | 4.4 | | | | 220Ω | | | | 1 | | | e |
| 5 | | | | | 4.5 | | | | 220Ω | | | | 9 | | | f |
| 6 | | | | | 4.6 | | | | 220Ω | | | | 10 | | | g |

C code

////////////////////////////////////////////////////////

//Willard Wider

//Lab 05 - Seg7Dspl

//05/22/18

//7 Segment Display

////////////////////////////////////////////////////////

**#include** <ti/devices/msp432p4xx/driverlib/driverlib.h>

**#define** DELAY 50000

**int** **main**(**void**)

{

**volatile** uint32\_t i, j;

**unsigned** **char** seq = 0x00;

// Stop watchdog timer

WDT\_A\_hold(WDT\_A\_BASE);

//set the pins for the fewq div

**GPIO\_setAsOutputPin**(GPIO\_PORT\_P4,GPIO\_PIN0);

**GPIO\_setAsOutputPin**(GPIO\_PORT\_P4,GPIO\_PIN1);

**GPIO\_setAsOutputPin**(GPIO\_PORT\_P4,GPIO\_PIN2);

**GPIO\_setAsOutputPin**(GPIO\_PORT\_P4,GPIO\_PIN3);

**GPIO\_setAsOutputPin**(GPIO\_PORT\_P4,GPIO\_PIN4);

**GPIO\_setAsOutputPin**(GPIO\_PORT\_P4,GPIO\_PIN5);

**GPIO\_setAsOutputPin**(GPIO\_PORT\_P4,GPIO\_PIN6);

**GPIO\_setAsOutputPin**(GPIO\_PORT\_P4,GPIO\_PIN7);

//DEBUG: show at least something is working (like the LED)

//show that at least the code is being entered

**GPIO\_setAsOutputPin**(GPIO\_PORT\_P1,GPIO\_PIN0);

//DEBUG: show at least something is working

**GPIO\_setOutputHighOnPin**(GPIO\_PORT\_P1,GPIO\_PIN0);

//usefull methods

//GPIO\_setOutputHighOnPin(GPIO\_PORT\_P4, ++incriment);

//GPIO\_setOutputLowOnPin(GPIO\_PORT\_P4, ~incriment);

//hex to 7 seg values are as follows

**while**(1)

{

//set the displays back off

**GPIO\_setOutputHighOnPin**(GPIO\_PORT\_P4, 0x7F);

//set the displays on based on case

displayNumberz(seq);

//delay of pause

**for**(i = 0; i < DELAY; i++)

{

}

**if** (seq == 0x0F)

{

//set it to FF os that when it's incrimented it will go back to 0

seq = 0xFF;

}

seq++;

}

}

**void** **displayNumberz**(**char** num)

{

**switch** (num)

{

**case** 0x00:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x3F);

**break**;

**case** 0x01:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x06);

**break**;

**case** 0x02:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x5B);

**break**;

**case** 0x03:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x4F);

**break**;

**case** 0x04:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x66);

**break**;

**case** 0x05:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x6D);

**break**;

**case** 0x06:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x7D);

**break**;

**case** 0x07:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x07);

**break**;

**case** 0x08:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x7F);

**break**;

**case** 0x09:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x67);

**break**;

**case** 0x0A:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x77);

**break**;

**case** 0x0B:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x7C);

**break**;

**case** 0x0C:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x39);

**break**;

**case** 0x0D:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x5E);

**break**;

**case** 0x0E:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x79);

**break**;

**case** 0x0F:

**GPIO\_setOutputLowOnPin**(GPIO\_PORT\_P4,0x71);

**break**;

}

printf("%x\n",num);

}