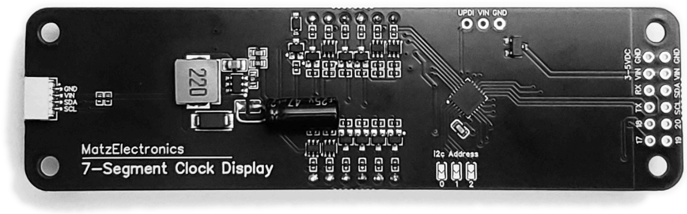
7-Segment Clock Display

By MatzElectronics

Front:

Back:



Features:

- 0.94 inch (24 mm) digit height
- Excellent reliability in bright ambient light.
- 4 7-segment digits
- 2 colon dots and 3 status dots
- Low power consumption
- Adjustable brightness
- White LED segments and a matte-black surface
- I2C/TWI interface, with breakouts for UART and GPIO (can be utilized by modifying the firmware)
- ATtiny3217 microcontroller
- User-programmable firmware, programmable using a UPDI interface

Applications:

- Home appliances and gadgets
- Instrument panels
- Digital readout displays
- Game console systems

Electrical Specifications:

- VIN 2.8 - 5 Volts DC (Absolute Max 5.5V)
- Module uses a pFET for reverse-polarity protection between GND and VIN

Current Consumption

- VIN = 3V : 200 mA
- VIN = 5V : 470 mA
    - Note: LEDs are brighter at 5V.

Component datasheets:

- Display: [Luckylight KW4-8828AWB]
- Microcontroller: [Microchip ATTINY3217]
- Boost Converter: [Aerosemi MT3608]

Pinout:

2x6 0.1” (.254 mm) unpopulated header:

17 (PC5)	18 (ADDR0, PA1)	TX (9, PB2)	RX (8, PB3)	VIN (+)	GND
19 (ADDR1, PA2)	20 (ADDR2, PA3)	SCL (11, PB0)	SDA (10, PB1)	VIN (+)	GND

1x4 JST-SH (STEMMA-QT, QWIIC) connector:

SCL (11, PB0)	SDA (10, PB1)	VIN (+)	GND

SDA and SCL pins are connected to 10KΩ pull-up resistors.

I2C/TWI Interface:

The I2C/TWI interface uses a signal (SDA) wire and a clock (SCL) wire in a bi-directional, open-drain configuration. Both the SDA and SCL lines are pulled up with 10kΩ resistors.

The default I2C address for the module is **0x2C**. There are solder jumpers marked ADDR0, ADDR1, and ADDR2. By closing one or more of these jumpers, they create a 3-bit number that is added to the base I2C address at startup, allowing up to 8 modules to share the same I2C bus.

Register Map

Register (Hex)	(Dec)	Default Value	Description
0x00	0	0	Digit 1 (leftmost)
0x01	1	0	Digit 2
0x02	2	0	Digit 3
0x03	3	0	Digit 4 (rightmost)
0x04	4	0b0011000	Dot Map (default: center colon on, others off)
0x05	5	0b0011000	Dot Blink Mask (1 = blinking) (default: center colon blinking)
0x06	6	2	Dot Blink Rate (default: 512 ms)
0x07	7	0	All Digits Numeric Value MSB (signed 16-bit int)
0x08	8	0	All Digits Numeric Value LSB (signed 16-bit int)
0x09	9	0	Digits 1 & 2 (leftmost) Numeric Value (signed 8-bit int)
0x0A	10	0	Digits 3 & 4 (rightmost) Numeric Value (signed 8-bit int)
0x0B	11	0b100	Leading Zeros on Numeric Values
0x0C	12	127	Brightness (8-bit unsigned int)

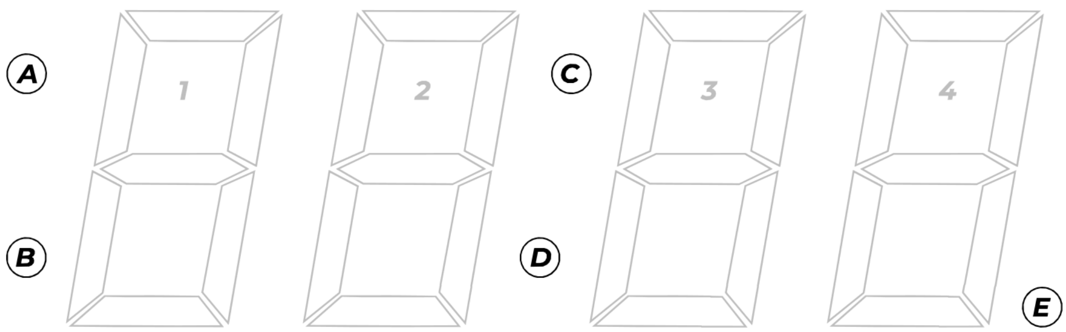
Register Descriptions

Registers 0 - 3: Digit 1 (leftmost) through Digit 4 (rightmost) value

Action: Read (truthful), Write
Type: 8-bit unsigned int/char
Behavior: Writing a byte to this register sets that corresponding digit on the display to the value of the register (via a segment map). Writing 0-9 will set the digit to that number. Writing an ASCII (32-127) character will set the digit to that character, to the best of its ability.

Register 4: Dot Map

Action: Read (truthful), Write
Type: 8-bit unsigned int
Behavior: The bits in this register represent Dots on the display:



Bits:

7 (MSB)	6	5	4	3	2	1 (LSB)
A	B	C	D	E	F (no dot)	G (no dot)

Register 5: Dot Blink Mask

Action: Read (truthful), Write
Type: 8-bit unsigned int
Behavior: The bits in this register represent which dots on the display are blinking (see diagram above). When a bit is set (1), the corresponding dot will blink if it is set.

Register 6: Dot Blink Rate

Action: Read (truthful), Write
Type: 8-bit unsigned int, Range 0-6, inclusive
Behavior: Any dots set to blink using the registers above will blink at the rate specified by this register's value:

- **0:** 4096 ms
- **1:** 2048 ms
- **2:** 1024 ms
- **3:** 512 ms (default)
- **4:** 256 ms
- **5:** 128 ms
- **6:** 64 ms

Registers 7 (MSB), 8 (LSB):

Action: Read (only truthful if set by a previous write to this register), Write
Type: 16-bit signed int, Range -999 to 9999, inclusive
Behavior: By writing a numerical value to this register, all 4 digits will display the integer value.

Registers 9 (Left 2 digits), 10 (Right 2 digits):

Action: Read (only truthful if set by a previous write to the same register), Write
Type: 8-bit signed int, Range -9 to 99, inclusive
Behavior: By writing a numerical value to this register, the 2 digits (depending on which register) will display the integer value.

Register 11:

Action: Read (truthful), Write
Type: 8-bit unsigned int
Behavior: Used by writes to registers 7-10 to determine whether leading zeros should be displayed as part of the numerical values.

Bits:

3 (MSB)	2	1 (LSB)
Leading zero for 2 digit writes (right digits) (register 10), default value is yes (1)	Leading zero for 2 digit writes (left digits) (register 9), default value is no (0)	Leading zeros for 4 digit writes (registers 7-8), default value is no (0)

Register 12:

Action: Read (truthful), Write
Type: 8-bit unsigned int
Behavior: Sets the brightness of the display. Very low values may cause visible flicker. Default is 127 (approx. 50% brightness).

UART Interface:

While the current firmware does not support writing characters to the module using UART/Serial, RX and TX pins are broken out to the right side header and the firmware can be modified to use them.

GPIO interface:

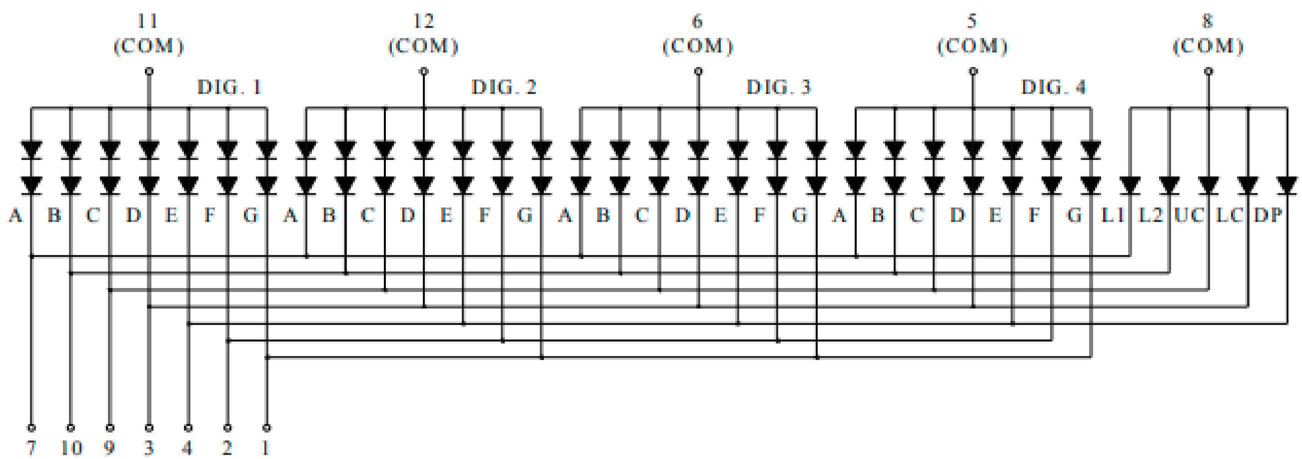
The current firmware uses 3 of the 4 available GPIO pins (18, 19, and 20) to set the I2C address, and therefore they should not be used without modifying the firmware. Pin 17 is also not currently used. These pins are broken out so that they are available if you choose to write your own custom firmware.

Firmware Customization:

The firmware for this module is written in Arduino/C++. The board type is a generic ATTINY3217 using the megaTinyCore by Spence Konde. It is flashed via UPDI (there is a UPDI header on the PCB) using a SerialUPDI adapter, which can be easily constructed with a CH340 or FTDI USB to Serial adapter and a single resistor.

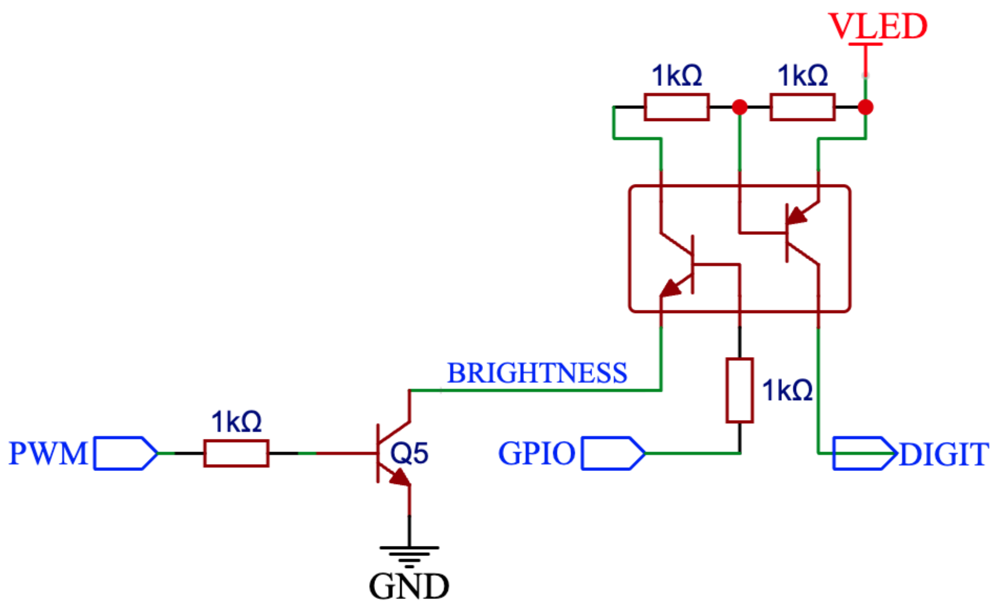
Firmware / Hardware Pin Mapping:

The 7-segment display is a common-anode LED display that uses two LEDs in series for each segment, and a single LED for each dot:

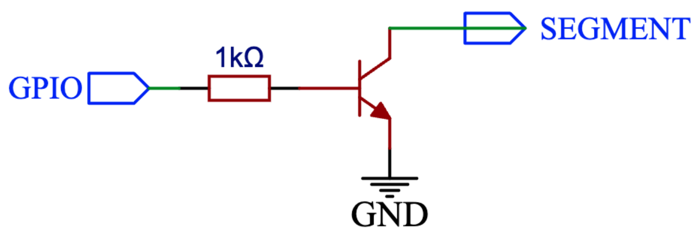


Each digit is driven using a NPN->PNP high-side driver, with the “ground” of each driver being connected through another NPN transistor which is connected to a PWM output. This allows for global, uniform brightness control (see diagram below). The Dot “digit” is connected in series with a reversed Zener diode to reduce the voltage to each dot by 3V.

The segments are powered using a boost converter outputting approximately 12VDC. This voltage is then pulse-width modulated (PWM) and further switched, reducing the average voltage to each segment to roughly 6V at full brightness.



Each segment is controlled by a single NPN low-side driver:



The ATTINY's pin are mapped to the digits and segments as shown:

PWM Pin: 0 (PA4)

Segments:

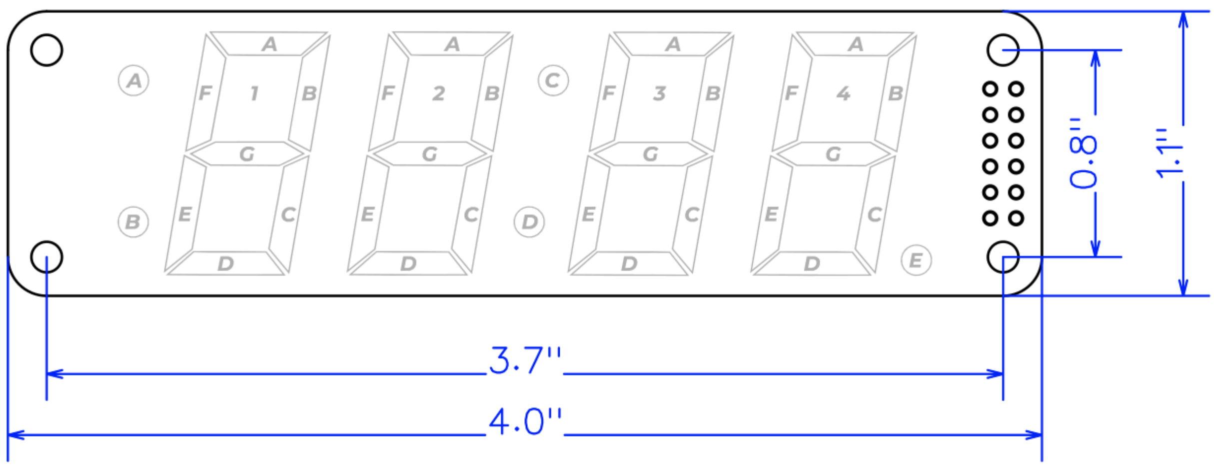
Segment A	Segment B	Segment C	Segment D	Segment E	Segment F	Segment G
7 (PB4)	5 (PB6)	6 (PB5)	2 (PA6)	1 (PA5)	3 (PA7)	4 (PB7)

Digits:

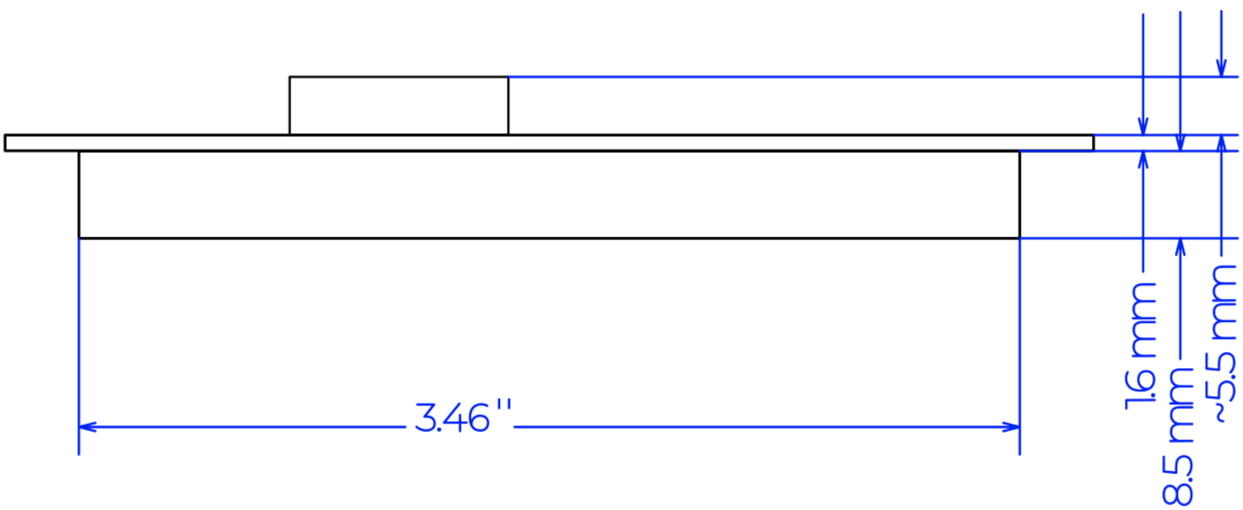
Digit 1 (leftmost)	Digit 2	Digit 3	Digit 4 (rightmost)	Digit Dots
15 (PC3)	14 (PC2)	12 (PC0)	13 (PC1)	16 (PC4)

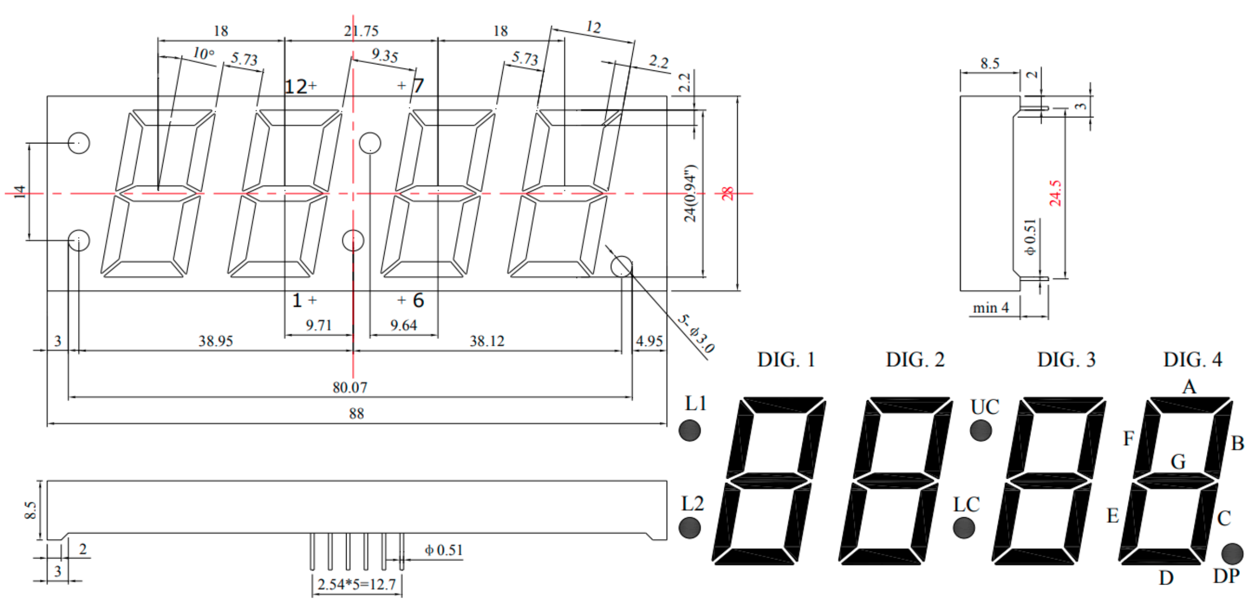
Physical Characteristics:

Dimensions



Mounting hole diameter is 0.118” (3 mm)





Design Files and Firmware:

Design files, firmware, and code examples can be found here:
https://github.com/MatzElectronics/7-Segment-Clock-Display

Disclaimer and Waiver of Liability:

By connecting and using this product, you acknowledge and agree that this product is provided "as is" without any warranties, express or implied, regarding its functionality, compatibility, or fitness for any particular purpose. MatzElectronics, Matthew Matz, HMB Creative, and/or any affiliates shall not be held liable for any damages, injuries, or losses arising from the use, assembly, or installation of this product, including but not limited to damage to connected components or devices, personal injury, or property damage.

It is recommended that you ensure proper handling and reasonable safety procedures are followed during use. By proceeding with the use of this device, you assume full responsibility for any outcomes. MatzElectronics, Matthew Matz, HMB Creative, and/or any affiliates will not be held responsible for any outcomes or damages resulting from the use or misuse of this product.