|  |  |
| --- | --- |
| /\* SevSeg Library |  |
|  |  |
|  | Copyright 2014 Dean Reading |
|  |  |
|  | Licensed under the Apache License, Version 2.0 (the "License"); |
|  | you may not use this file except in compliance with the License. |
|  | You may obtain a copy of the License at |
|  | http://www.apache.org/licenses/LICENSE-2.0 |
|  |  |
|  | Unless required by applicable law or agreed to in writing, software |
|  | distributed under the License is distributed on an "AS IS" BASIS, |
|  | WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. |
|  | See the License for the specific language governing permissions and |
|  | limitations under the License. |
|  |  |
|  |  |
|  | This library allows an Arduino to easily display numbers in decimal format on |
|  | a 7-segment display without a separate 7-segment display controller. |
|  |  |
|  | Direct any questions or suggestions to deanreading@hotmail.com |
|  | See the included readme for instructions. |
|  | \*/ |
|  |  |
|  | // If you use current-limiting resistors on your segment pins instead of the |
|  | // digit pins, then change the '0' in the line below to a '1' |
|  | #define RESISTORS\_ON\_SEGMENTS 0 |
|  | #define MAXNUMDIGITS 8 //Increase this number to support larger displays |
|  |  |
|  |  |
|  | #ifndef SevSeg\_h |
|  | #define SevSeg\_h |
|  |  |
|  | #if defined(ARDUINO) && ARDUINO >= 100 |
|  | #include "Arduino.h" |
|  | #else |
|  | #include "WProgram.h" |
|  | #endif |
|  |  |
|  | // Use defines to link the hardware configurations to the correct numbers |
|  | #define COMMON\_CATHODE 0 |
|  | #define COMMON\_ANODE 1 |
|  | #define N\_TRANSISTORS 2 |
|  | #define P\_TRANSISTORS 3 |
|  | #define NP\_COMMON\_CATHODE 1 |
|  | #define NP\_COMMON\_ANODE 0 |
|  |  |
|  |  |
|  | class SevSeg |
|  | { |
|  | public: |
|  |  SevSeg(); |
|  |  |
|  |  void refreshDisplay(); |
|  |  void begin(byte hardwareConfig, byte numDigitsIn, byte digitPinsIn[], byte segmentPinsIn[]); |
|  |  void setBrightness(int brightnessIn); // A number from 0..100 |
|  |  |
|  |  void setNumber(long numToShow, byte decPlaces); |
|  |  void setNumber(unsigned long numToShow, byte decPlaces); |
|  |  void setNumber(int numToShow, byte decPlaces); |
|  |  void setNumber(unsigned int numToShow, byte decPlaces); |
|  |  void setNumber(char numToShow, byte decPlaces); |
|  |  void setNumber(byte numToShow, byte decPlaces); |
|  |  void setNumber(float numToShow, byte decPlaces); |
|  |  |
|  | private: |
|  |  void setNewNum(long numToShow, byte decPlaces); |
|  |  void findDigits(long numToShow, byte decPlaces, byte nums[]); |
|  |  void setDigitCodes(byte nums[], byte decPlaces); |
|  |  |
|  |  boolean digitOn,digitOff,segmentOn,segmentOff; |
|  |  byte digitPins[MAXNUMDIGITS]; |
|  |  byte segmentPins[8]; |
|  |  byte numDigits; |
|  |  byte digitCodes[MAXNUMDIGITS]; |
|  |  int ledOnTime; |
|  |  const static long powersOf10[10]; |
|  |  |
|  | }; |
|  |  |
|  | #endif //SevSeg\_h |
|  | /// END /// |