ENHANCING PASSENGER COMFORT WITH AUTOMATIC INTERIOR CLIMATE SYSTEMS IN VEHICLES

Dr. Emily Reed*[1] & Mason Scott[2]
[1, 2] Department of Materials Science, University of Queensland, Australia

This paper describes the design and implementation of a senior project involving a car heating and cooling system. During the summer and winter times, for cars parked outside, the interior temperature can become very high or very low causing discomfort for the driver and passengers. Therefore a device is needed to keep the vehicle interior at comfortable temperatures while standing on the parking lot or on the driveway. The goal of this project was to build a light weight compact car heater and cooler. The car heating and cooling system allows car users to maintain a reasonable temperature while sitting at home or in a parking lot. The designed system can fit in a back window. The system runs using a 12V DC power source powered through a solar panel that is mounted on the back window of the vehicle. This location was chosen because the exposure to light is greater in the rear and allows the solar panel to recharge faster and more efficiently. All components used in this system are rated 12VDC, with the exception of the microcontroller; an additional 5V DC regulator is used to power the Arduino microcontroller. Using a thermistor, the microcontroller tracks the temperature inside a vehicle. When a high temperature (i.e. temp > 70°F) condition is detected by the thermistor, the microcontroller l enable s a power relay to apply power to both the fan and the cooling element, based on a peltier and heat sink. This process continues until the temperature is within the specified range (i.e. 66°F ~ 70°F) or until user disables the system. In the same manner, when a low temperature (i.e. temp < 66°F) is detected by the microcontroller through the thermistor, a power relay is enabled to apply power to both the fan and the heating element, also based on a peltier and heat sink. This process continues until the temperature is within the specified range (i.e. 66°F ~ 70°F) or until user disables the system. As an additional feature, an air freshener pump is also included in the system that can be activated at 30 minute intervals and will continue until user disables the system. All features and temperature measurements are tracked and displayed on the LCD which is connected to the Arduino microcontroller. The paper covers the details of the design of hardware and software components of the system,

1. INTRODUCTION

The objective of this project is to design and implement an effective and inexpensive system to safely heat up or cool down the interior of a compact car when it is parked in parking lot or on a driveway. The designed system will be compact and easy to install with a self-sufficient power supply from a solar panel. During the summer and winter seasons, for cars parked outside the cabin temperature can become be too high or too low thus making it difficult for drivers to sit comfortably in the cars to operate them. This paper presents a solution to this problem. The designed project keeps the interior of a car at a comfortable temperature while the car is parked in parking lot or on the driveway

2. LITERATURE SURVEY

Similar devices are available in the market but have following limitations:

• Solar cooler powered fans and battery powered heaters have drawbacks.

Small and inefficient, can't cool/heat entire car

Do not have temperature sensors

• Devices that use the vehicle's battery are not recommended as the battery can die if the device is left on for a long time.

Devices that use batteries are expensive to maintain.

Fans will only blow around the hot air inside the vehicle

3. SYSTEM OVERVIEW

Block diagram

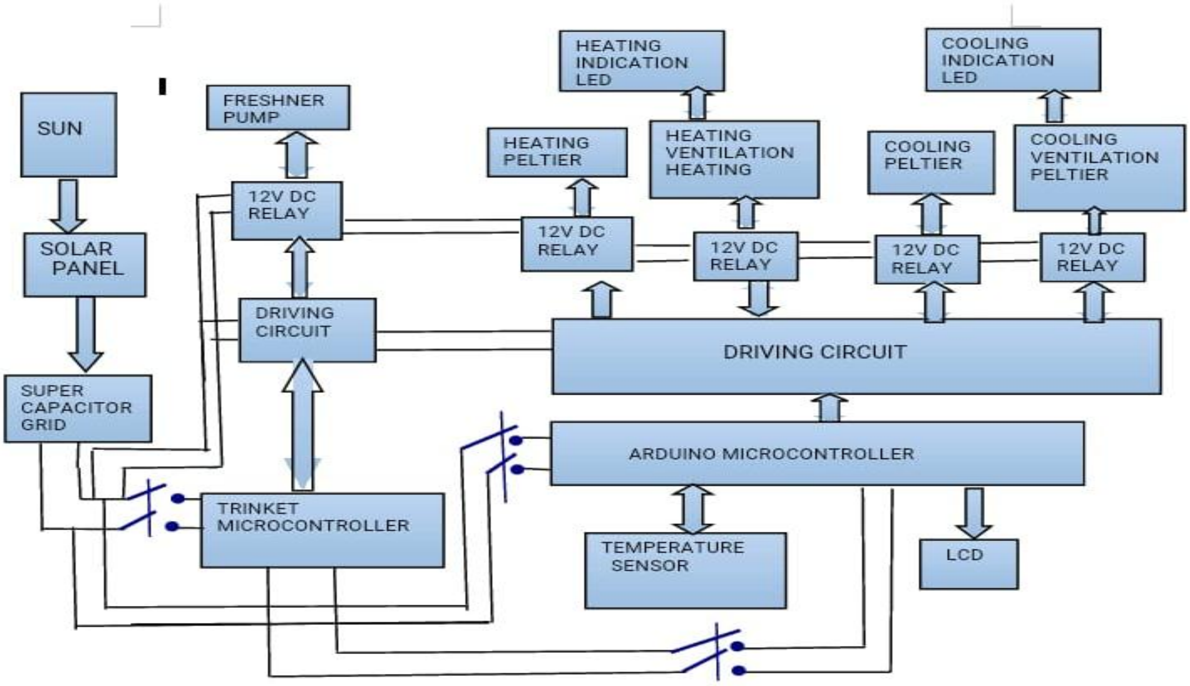


Figure : block diagram

Figure 1 illustrates the block diagram of the system. The designed system can maintain a constant temperature inside a vehicle regardless of what the outside temperature is. The entire system runs using 12VDC that is provided through a solar panel mounted on the back window of the vehicle. This location was chosen because the exposure to light is greater in the rear and will allow the solar panel to recharge faster and more efficiently. All components used in this system are rated 12VDC. Using a thermistor, a microcontroller tracks the temperature inside a vehicle. When a high temperature (temp > 70°F) condition is detected by the thermistor, the microcontroller enables RY1 and RY2 power relays to apply power to both the fan and the cooling element (peltier). This process continues until the temperature reaches within the specified range of 66°F ~ 70°F or until user disables the system. In the same manner, when a low temperature (temp < 66°F) is detected by the microcontroller through the thermistor, RY3 and RY4 power relays are enabled to apply power to both the fan and the heating element (peltier). This process will continue until the temperature is within the specified range of 66°F ~ 70°F or until user disables the system. In order to set desirable the lowest and highest temperature, there are three buttons, select, increase and decrease, as a human interface option. By pressing select button, users can set the highest temperature by using the increasing and decreasing buttons, then the lowest temperature in the same way. As an optional feature, an air freshener pump could be added to the system that will be activated through RY5 at 1 hour intervals and will continue until user disables the system. All features and temperature readings are tracked and displayed on the LCD display of the systems

H I G H S E T P O I N T : 8 6 °
F L O W S E T P O I N T : 6 5 ° F
T E M P E R A T U R E : 7 6 . 0 0 ° F

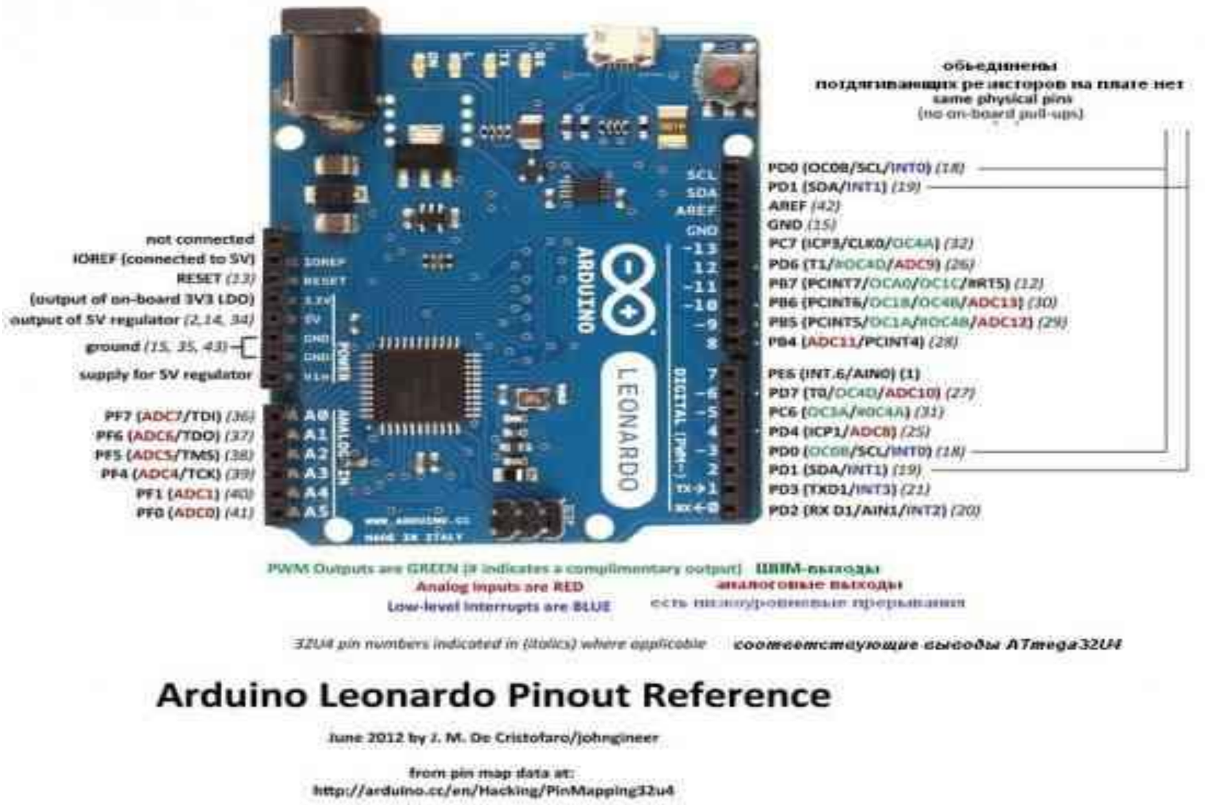
H I G H S E T P O I N T : 8 6 °
F L O W S E T P O I N T : 6 5 ° F
T E M P E R A T U R E : 7 6 . 0 0 ° F
H E A T I N G

H I G H S E T P O I N T : 8 6 °
F L O W S E T P O I N T : 6 5 ° F
T E M P E R A T U R E : 7 6 . 0 0 ° F

C O O L I N G

Arduino Leonardo Microcontroller controls and signals the appropriate elements to turn on turns on, it also sends information to be displayed on the LCD display, based on the measurements it receives from the temperature sensor.

Pin diagram



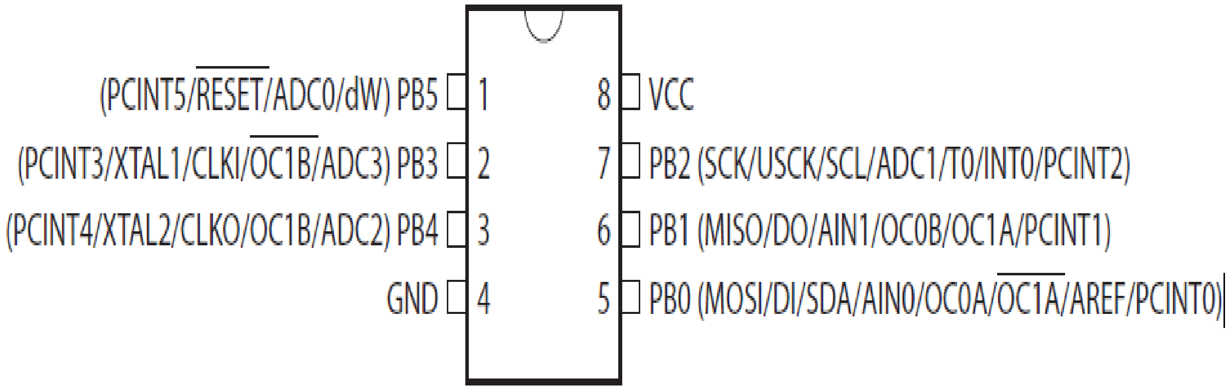
Specifications and features

- **Input and Output:** Each of the 20 digital i/o pins on the Arduino Leonardo can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms.
  In addition, some pins have specialized functions:.
- **Serial:** 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data using the ATmega32U4 hardware serial capability. Note that on the Leonardo, the Serial class refers to USB (CDC) communication; for TTL serial on pins 0 and 1, use the Serial1 class.
- **TWI:** 2 (SDA) and 3 (SCL). Support TWI communication using the Wire library. External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
- **PWM:** 3, 5, 6, 9, 10, 11, and 13. Provide 8-bit PWM output with the analogWrite() function.
- **SPI:** on the ICSP header. These pins support SPI communication using the SPI library. Note that the SPI pins are not connected to any of the digital I/O pins as they are on the Uno, They are only available on the ICSP connector. This means that if you have a shield that uses SPI, but does NOT have a 6-pin ICSP connector that connects to the Leonardo's 6-pin ICSP header, the shield will not work.
- **LED:** 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- **Analog Inputs:** A0-A5, A6 – A11 (on digital pins 4, 6, 8, 9, 10, and 12). The Leonardo has 12 analog inputs, labeled A0 through A11, all of which can also be used as digital i/o. Pins A0-A5 appear in the same locations as on the Uno; inputs A6-A11 are on digital i/o pins 4, 6, 8, 9, 10, and 12 respectively. Each analog input provide 10 bits of resolution (i.e. 1024 different values). By default the analog inputs measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function.

There are a couple of other pins on the board:

1. **AREF.** Reference voltage for the analog inputs. Used with analogReference().
2. **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.
3.

TRINKET MICROCONTROLLER
PIN Diagram



Specifications
Power pins
BAT+ is the Battery + Input pin. To power the trinket from a battery or power adapter or solar panel or any other kind of power source, connect the + (positive) pin to BAT+ We can connect up to 16V DC. This input is reverse-polarity protected.

USB+ is the USB + Output .USB 5V power for charging a battery, or if we need more than 150mA of current (this pin can supply 500mA+ from USB ports) or to detect when the Trinket is plugged into USB, this pin will have 5V power on it if and only if its plugged into something via the mini B connector

GND is the common ground pin, used for logic and power. It is connected to the USB ground and the power regulator, etc. This is the pin you'll want to use for any and all ground connections

GPIO Pins
There are 5 GPIO pins.All the GPIO pins can be used as digital inputs, digital outputs, for LEDs, buttons and switches etc. They can provide up to 20mA of current. Don't connect a motor or other high-power component directly to the pins Instead, use a transistor to power the DC motor on/off

On a 3V Trinket, the GPIO are 3.3V output level, and should not be used with 5V inputs. On a 5V Trinket, the GPIO are 5V output level, and can be used with 3V inputs but may damage electronic devices that are 3V input only.

The first 3 pins are completely 'free' pins, they are not used by the USB connection so you never have to worry about the USB interface interfering with them when programming
GPIO #0 - this is connected to PB0 on the ATtiny85. This pin can be used as a PWM output, and is also used for I2C data, and SPI data input.

GPIO #1 - this is connected to PB1 on the ATtiny85. This pin can be used as a PWM output, and is also used for SPI data output. This pin is also connected to the onboard LED (like pin 13 on a regular Arduino).

GPIO #2 - this is connected to PB2 on the ATtiny85. This pin can be used as an analog input (known as Analog A1), and is also used for I2C clock and SPI clock.

USB Pins

The next 2 pins are also used for USB programming. That means that when the Trinket is connected to a computer and in bootloader mode or in the middle of uploading a new program, they are used for sending data to/from the computer! It's possible to share these pins. if you are careful. These pins can be used as outputs to things like LEDs.

GPIO #3 - this is connected to PB3 on the Attiny85. This pin is used for USB programming, but its also an analog input known as Analog A3. This pin has a 1.5K pullup to 3.3V built into the Trinket, for USB comm so it may be difficult to use for analog or digital input.

GPIO #4 - this is connected to PB4 on the Attiny85. this pin is used for USB programming, but it can also be used as a PWM analog output and an analog input known as Analog A2

Reset and Regulator Output
The final two pins are at the bottom of the board

First is the Rst reset pin. This is connected directly to the ATtiny85's reset pin and also the reset button which is right next to it. The reset pin is used to enter the bootloader and to reset the board in case you want to restart it. It's also possible to use this pin to re-program in the bootloader or completely remove the bootloader if we have an AVR programmer such as an AVR Dragon, MKii or USBtinyISP. We can re-program the board when its in an enclosure or box or otherwise hard to reach, wire a simple button from the RST pin to ground and press it to enter the bootloader for 10 seconds. The reset button cannot be used as a GPIO, but we think its a lot more useful as a proper reset button!Lastly we have the regulator output pin. There is an onboard mini power regulator that will take up to 16V DC from the BAT+ or USB connection and regulate it down to a steady 3.3V or 5.0V DC so its safe to use with your sensors and LEDs. On a 3V Trinket, this output will be about 3.3V. On a 5V Trinket, this output will be 5V so be aware in case you want to swap one with the other. You can draw up to 150mA output from this pin.

4. ADVANTAGES AND APPLICATIONS

Equipment will be able to heat/cool an entire compact size car
• The size of the product will fit on to the back window
• Automatic control
• Available in different colors
• Option for air freshener
• Small and powerful device
• Affordable for every income level
• Self-sufficient power supply, using the sun’s energy.
• Environmentally friendly device

5. CONCLUSION

This paper described the design and successful implementation of a senior project “Heating and Cooling System for car’s interior.” The designed device can control the temperature of a car’s interior when the car is not on or running its heater or AC. The device runs on energy generated from a solar panel mounted on the vehicle, making it self-sufficient and environmentally friendly. The application of the system is to maintain temperatures within a predetermined range Page 24.102.11 inside the car for drivers to get inside a comfortable car. Moreover, the project allowed for an educational experience in which the team members incorporated concepts learned in their corresponding programs of EET and CET. The team members integrated concepts learned in electronic and programming courses to program a microcontroller and assemble the circuitry of the device.

REFERENCES

[1] Rosanne Spector , Parked Car Gets Dangerously Hot, Even On Cool Days, Studies Finds.,. Internet : http: /med. stanford. edu/news_releases/2005/july/hot-cars. htm, May 7, 2005 [May 6, 2014].
[2] Ibrahim Almanjahie, Temperature Variation in a Parked Car, Masters in Math. Stat. Sc, University of Western Australia, 2008).
[3] R. Saidur, H. H. Masjuki and M. Hasanuzzaman, Performance of an Improved Solar Car Ventilator, International Journal of Mechanical and Materials Engineering (IJMME), Vol. 4 (2009), No. 1, 24 -34.
[4] Antonio Martin Galvez-Ramos , Climate control system for parked automobiles, Patent Application Number-20090130965, 2009-05-21.
[5] Michael J. Lesle, Paul J. Kolokowski, Through Glass Ventilation, US Patent 12/645, 935, 2009-12-23.

[6] Hussain H. Al-Kayiem, M. Firdaus Bin M. Sidik and Yuganthira R.A. L Munusammy, Study on the Thermal Accumulation and Distribution Inside a Parked Car Cabin,. American Journal of Applied Sciences 7. vol. 6, pp.784-789, (2010)