Display API

void display_send(void): Sends all of the data from the internal buffer out to the oled display.
void display_fill(unsigned char byte): Fills the display buffer with the specified value, it is useful when trying to fill the screen, or to clear it.
*void display_writebuffer(int row, int col, char *string)*: Writes to the display buffer at the specified coordinates. Its' x coordinate is row, and Its' y coordinate is col. Both coordinates are measured in characters (8 pixels).

GPIO API

bool gpio_init(void): Initializes the gpio functions, and prepares for future use. Returns false if it was unable to initialize; True if it was able to initialize.
bool gpio_config(unsigned int pin, unsigned int state): Configures the direction of a pin. The direction is whether it is input (state == 0) or output (state == 1). Unconfigured pins can't be written to, or read from. Returns false if it was unable to configure the pin; true if it was able to.
bool gpio_write(unsigned int pin, unsigned int state): Writes to the pin, configuring it to be high, (state == 1) or low (state == 0). Can only be used on pins whose directions are set for output. Returns true
int gpio_read(unsigned int pin): Reads the state of a pin. It will return 0 if the pin is low, 1 if the pin is high, and -1 if there was an error.
void gpio_close(void): Closes down the library.

I2C API

__s32 i2c_smbus_write_quick(int file, __u8 value): Transmits the byte *value.*
__s32 i2c_smbus_read_byte(int file): Reads a byte.
__s32 i2c_smbus_write_byte(int file, __u8 value): Transmits the byte *value.*
__s32 i2c_smbus_read_byte_data(int file, __u8 command): Reads the byte returned from the *command.*
__s32 i2c_smbus_write_byte_data(int file, __u8 command, __u8 value): Transmits *value* as a parameter to *command.*
__s32 i2c_smbus_read_word_data(int file, __u8 command): Reads the word returned from *command.*
__s32 i2c_smbus_write_word_data(int file, __u8 command, __u16 value): Transmits *value* as a parameter to *command.*
*__s32 i2c_smbus_read_block_data(int file, __u8 command, __u8 *values):* Reads the array of bytes returned from *command.*
*__s32 i2c_smbus_write_block_data(int file, __u8 command, __u8 length, const __u8 *values):* Transmits the array of bytes **values* as a paramteres to *command.*

DOW TEMP API

void w1temp_init(): Initializes the onewire functions.
int w1temp_getcount(): Returns the count of dow devices on the bus.
double w1temp_get_temp_celsius(int index): Returns the temperature of the device in celsius.
double w1temp_get_temp_fahrenheit(int index): Returns the temperature of the device in fahrenheit.
void w1temp_close(): Closes down the library.