

v2 sensor board firmware

Switch to predefined array of sensors instead of i2c scan

Creating sensor types

sensors with different data retrieval methods

id	name	Sensors
0	SHT2x	SHT20, SHT21, SHT25, HTU21d
1	Si7021	Si7021
2	HDC10xx	HDC1080
3	SHT3c	SHT30, SHT31, SHT35, SHT88
4	BME280	BME280
5	BME680	BME680
6	DHT1x	DHT12
7	DHT2x	DHT22

```
const uint8_t EMPTY = 0; /* slot is empty or sensor disabled */
const uint8_t SHT2X = 1; /* include SHT20, SHT21, SHT25, HTU21d*/
const uint8_t SI70XX = 2; /* includes Si7021 */
const uint8_t HDC10xx = 3; /* includes HDC1080 */
const uint8_t SHT3X = 4; /* include SHT30, SHT31, SHT35, SHT88*/
const uint8_t BME280 = 5; /* includes BME280 */
const uint8_t BME680 = 6; /* includes BME680 */
const uint8_t DHT1X = 7; /* includes DHT12 */
const uint8_t DHT2X = 8; /* includes DHT22 */
```

Define multiplexors

Id	I2c addr	Description	switch
0	112	Main Multiplexer with most of SHT2x, SHT3x, BMEx, DHT12 sensors	000
1	113	Second Multiplexor with Si7021	001
2	114	Third Multiplexer with HDC1080, BME680-2, SHT85, SHT3x	010
3	115	Fourth Multiplexer with DHT22 (5v sensors), additional board	100

```
uint8_t multiplexer[4]={112,113,114,115};
```

Define sensors map

```
using v2 7x8 Sensors board project#SensorsMap
```

[illegible]

3	HDC1080	HDC1080	HDC1080	HDC1080	HDC1080	HDC1080	HDC1080	HDC1080
	BME680	BME680		SHT85	SHT85	SHT85	SHT31	SHT31
4	DHT22	DHT22	DHT22	DHT22				

see [Test i2c humidity sensors#Sensori2caddressesandvoltages](#)

Sensor	address	alt address
Senserion SHT2x	0x40 (64)	0x41 (65)
Senserion SHT3x	0x44 (68)	0x45 (69)
Senserion SHT8x	0x44 (58)	
Meas⁽¹⁾ HTU21D	0x40 (64)	
Silicon Labs Si7021	0x40 (64)	
Bosch⁽²⁾ BMEx80	0x76 (118)	0x77 (119)
Bosch⁽²⁾ BME280	0x76 (118)	0x77 (119)
Ti⁽³⁾ HDC1080	0x40 (64)	
AOSONG DHT12	0x5C (92)	
AOSONG AM2320	0x5C (92)	0xB8 ?

¹ TE Connectivity Measurement Specialties

Prepare for code

	0	1	2	3	4	5	6	7
0	SHT25*	SHT21	SHT21	SHT21	SHT21	SHT20	SHT20	SHT20
	SHT35	SHT35	SHT31	SHT31	SHT31	SHT31	SHT30	SHT30
	BME680	BME680	BME280	BME280	BME280	DHT12	DHT12	DHT12
1	HTU21	Si7021	Si7021	Si7021	Si7021	Si7021	Si7021	Si7021
2	HDC1080	HDC1080	HDC1080	HDC1080	HDC1080	HDC1080	HDC1080	HDC1080
	BME680	BME680		SHT85	SHT85	SHT85	SHT31	SHT31
3	DHT22							

Array to map sensors

demo sketch to debug array with serial

```
uint8_t multiplexer[4] = {112, 113, 114, 115};
// Type of sensor
const uint8_t EMPTY = 0; /* slot is empty or sensor disabled */
const uint8_t SHT2X = 1; /* include SHT20, SHT21, SHT25, HTU21d*/
const uint8_t SI70XX = 2; /* includes Si7021 */
const uint8_t HDC10xx = 3; /* includes HDC1080 */
const uint8_t SHT3X = 4; /* include SHT30, SHT31, SHT35, SHT88*/
const uint8_t BME280 = 5; /* includes BME280 */
const uint8_t BME680 = 6; /* includes BME680 */
const uint8_t DHT1X = 7; /* includes DHT12 */
```

```

const uint8_t DHT2X = 8; /* includes DHT22 */
// indexes name in sensor arrays
const uint8_t get_type = 0; /* indexes name in sensor arrays */
const uint8_t get_collumn = 1; /* indexes name in sensor arrays */
const uint8_t get_address = 2; /* indexes name in sensor arrays */
// Sensor properties by [multiplexor][i2c_bus][number][get_type/get_collumn/get_address]
uint8_t sensor[4][8][3][3] =
{
    {
        { {SHT2X, 1, 64}, {SHT3X, 2, 68}, {BME680, 3, 119} },
        { {SHT2X, 1, 64}, {SHT3X, 2, 68}, {BME680, 3, 119} },
        { {SHT2X, 1, 64}, {SHT3X, 2, 68}, {BME280, 3, 118} },
        { {SHT2X, 1, 64}, {SHT3X, 2, 68}, {BME280, 3, 118} },
        { {SHT2X, 1, 64}, {SHT3X, 2, 68}, {BME280, 3, 118} },
        { {SHT2X, 1, 64}, {SHT3X, 2, 68}, {DHT1X, 3, 92} },
        { {SHT2X, 1, 64}, {SHT3X, 2, 68}, {DHT1X, 3, 92} },
        { {SHT2X, 1, 64}, {SHT3X, 2, 68}, {DHT1X, 3, 92} }
    },
    {
        { {SHT2X, 4, 64}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {SI70XX, 4, 64}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {SI70XX, 4, 64}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {SI70XX, 4, 64}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {SI70XX, 4, 64}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {SI70XX, 4, 64}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {SI70XX, 4, 64}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {SI70XX, 4, 64}, {EMPTY, 0, 0}, {EMPTY, 0, 0} }
    },
    {
        { {HDC10xx, 5, 64}, {BME680, 0, 118}, {EMPTY, 0, 0} },
        { {HDC10xx, 5, 64}, {BME680, 0, 118}, {EMPTY, 0, 0} },
        { {HDC10xx, 5, 64}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {HDC10xx, 5, 64}, {SHT3X, 6, 68}, {EMPTY, 0, 0} },
        { {HDC10xx, 5, 64}, {SHT3X, 6, 68}, {EMPTY, 0, 0} },
        { {HDC10xx, 5, 64}, {SHT3X, 6, 68}, {EMPTY, 0, 0} },
        { {HDC10xx, 5, 64}, {SHT3X, 6, 68}, {EMPTY, 0, 0} },
        { {HDC10xx, 5, 64}, {SHT3X, 6, 68}, {EMPTY, 0, 0} }
    },
    {
        { {DHT2X, 0, 92}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {DHT2X, 0, 92}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {DHT2X, 0, 92}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {DHT2X, 0, 92}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {DHT2X, 0, 92}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {DHT2X, 0, 92}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {DHT2X, 0, 92}, {EMPTY, 0, 0}, {EMPTY, 0, 0} },
        { {DHT2X, 0, 92}, {EMPTY, 0, 0}, {EMPTY, 0, 0} }
    }
};

void setup() {
    Serial.begin(115200);
}

void loop() {
    for (uint8_t m = 0; m < 4; m++) {
        for (uint8_t b = 0; b < 8; b++) {
            for (uint8_t n = 0; n < 3; n++) {
                for (uint8_t i = 0; i < 3; i++) {
                    Serial.print(sensor[m][b][n][i]);
                    Serial.print(", ");
                    delay (50);
                }
                Serial.print("\t");
                delay (50);
            }
            Serial.println();
            delay (50);
        }
    }
}

```

```
    delay (5000);  
}
```