#include <stdint.h>

#define SCL\_PIN GPIO\_Pin\_0

#define SDA\_PIN GPIO\_Pin\_1

#define SCL\_Hight GPIO\_SetBits(GPIOA,SCL\_PIN)

#define SCL\_Low GPIO\_ResetBits(GPIOA,SCL\_PIN)

#define SDA\_Hight GPIO\_SetBits(GPIOA,SDA\_PIN)

#define SDA\_Low GPIO\_ResetBits(GPIOA,SDA\_PIN)

#define AS6221\_REG\_TVAL 0x00

#define AS6221\_REG\_CONFIG 0x01

#define AS6221\_REG\_TLOW 0x02

#define AS6221\_REG\_THIGH 0x03

// Configuration constants

#define AS6221\_DEFAULT\_CONFIG 0x40A0

#define AS6221\_DEFAULT\_ADDRESS 0x48

#define AS6221\_T\_ERR 0.1

#define AS6221\_SINGLE\_SHOT 0x8000

typedef enum {

AS6221\_CONV\_RATE025 = 0x0000,

AS6221\_CONV\_RATE1 = 0x0040,

AS6221\_CONV\_RATE4 = 0x0080,

AS6221\_CONV\_RATE8 = 0x00C0,

} as6221\_conv\_rate\_t;

typedef enum {

AS6221\_STATE\_SLEEP = 0x0100,

AS6221\_STATE\_ACTIVE = 0x0000,

} as6221\_state\_t;

typedef enum {

AS6221\_CONSEC\_FAULTS1 = 0x0000,

AS6221\_CONSEC\_FAULTS2 = 0x1000,

AS6221\_CONSEC\_FAULTS3 = 0x2000,

AS6221\_CONSEC\_FAULTS4 = 0x3000,

} as6221\_consec\_faults\_t;

typedef enum {

AS6221\_ALERT\_INTERRUPT = 0x0200,

AS6221\_ALERT\_COMPARATOR = 0x0000,

} as6221\_alert\_mode\_t;

typedef enum {

AS6221\_ALERT\_ACTIVE\_LOW = 0x0000,

AS6221\_ALERT\_ACTIVE\_HIGH = 0x0400,

} as6221\_alert\_polarity\_t;

typedef struct {

as6221\_conv\_rate\_t cr;

as6221\_state\_t state;

as6221\_alert\_mode\_t alert\_mode;

as6221\_alert\_polarity\_t alert\_polarity;

as6221\_consec\_faults\_t cf;

uint8\_t singleShot;

} as6221\_config\_t;

static uint8\_t device\_address = AS6221\_DEFAULT\_ADDRESS;

static as6221\_config\_t device\_conf;

void AS6221\_GPIO\_Init(void);

void AS6221\_Init(uint8\_t addr, as6221\_config\_t conf);

double AS6221\_GetTemp(void);

void AS6221\_SetAlertLimits(uint16\_t low, uint16\_t high);

void AS6221\_SleepMode(void);

void AS6221\_CcMode(void );

void AS6221\_TriggerSingleShot(void );

void AS6221\_UpdateDevice(as6221\_config\_t conf);

#endif