

```

/*
    iiceep.h
        August 8,2004      Kenji Arai for H8/3664F
        October 10,2005

*/

/* -----< Definition > ----- */
#define DEVICE_CODE    0xa0 /* EEPROM DEVICE CODE: 1010(D3,D2,D1,D0) */
#define SLAVE_ADRS     0x00 /* SLAVE ADRS:0 000(A2,A1,A0) */
#define RW_CODE_W      0x00 /* R/W code: 0 (write) */
#define RW_CODE_R      0x01 /* R/W code: 1 (read) */
#define DEVICE_ADDRESS_WORD_W (DEVICE_CODE | SLAVE_ADRS | RW_CODE_W) /* スレーブアドレス(ライト用) */
#define DEVICE_ADDRESS_WORD_R (DEVICE_CODE | SLAVE_ADRS | RW_CODE_R) /* スレーブアドレス(リード用) */

//      Timeout
#define TIMEOUT_LIMIT_BBSY 1000 /* バスビジーチェックのタイムアウトリミット */
#define TIMEOUT_LIMIT_ACK 1000 /* ACKチェックのタイムアウトリミット */
#define TIMEOUT_LIMIT_TEND 1000 /* TENDチェックのタイムアウトリミット */
#define TIMEOUT_LIMIT_RDRF 1000 /* RDRFチェックのタイムアウトリミット */
#define TIMEOUT_LIMIT_STOP 1000 /* STOPチェックのタイムアウトリミット */

//      Error code
#define TIMEOUT_ERR_BUS_BUSY 200 /* エラーコード：バスビジータイムアウト */
#define TIMEOUT_ERR_ACK 201 /* エラーコード：アックポーリングタイムアウト */
#define TIMEOUT_ERR_TEND 202 /* エラーコード：送信完了タイムアウト */
#define TIMEOUT_ERR_RDRF 203 /* エラーコード：受信完了タイムアウト */
#define TIMEOUT_ERR_STOP 204 /* エラーコード：停止条件発行タイムアウト */
#define ERR_ACK 205 /* エラーコード：アックノリッジエラー */

//      define (others)
#define TM 125 // counter data for waiting timer

/* -----< Function Prototype iic2_eep.c> ----- */
unsigned char Read_byte_EEPROM( unsigned short );
unsigned char Write_byte_EEPROM( unsigned short , unsigned char );
unsigned char Check_bus_condition ( void );
unsigned char Master_address_set ( unsigned char Device_id, unsigned short Addr );
unsigned short Master_read_byte_random ( unsigned char Device_id, unsigned short Addr );
unsigned char Master_read_sequential ( unsigned char Device_id, unsigned short Addr, unsigned short Length, unsigned char *buff_ptr );
unsigned char Master_byte_write ( unsigned char Device_id, unsigned short Addr, unsigned char Data);
unsigned char Master_page_write ( unsigned char Device_id, unsigned short Addr, unsigned char Length, unsigned char *buff_ptr);
unsigned char Send_byte_data ( unsigned char Byte_data );
void Send_start_condition ( void );
void Set_iic_bus_mode ( unsigned char Bus_mode_MLS, unsigned char Bus_mode_WAIT, unsigned char Bus_mode_BC210);
void Set_iic_mode ( unsigned char iic_mode );
void Set_iic_rate ( unsigned char iic_rate );
unsigned char Send_stop_condition ( void );
void Set_iic2_if_enable ( unsigned char Ice );
unsigned char Set_slave_read_mode ( unsigned char Device_id );
unsigned char Set_slave_write_mode ( unsigned char Device_id );
unsigned short Receive_byte_data ( void );
void Set_receive_mode ( unsigned char Ackbt_flag, unsigned char Rcvd_flag );
unsigned char Receive_byte_data_many ( unsigned short Length, unsigned char *buff_ptr );

```

```
unsigned short Receive_byte_data_fin ( void );

void Init_eeprom ( void );
void I2c_start ( void );
void I2c_stop ( void );
void I2c_set ( unsigned char Scl , unsigned char Sda );
void I2c_bytesend ( unsigned char Tx_data );
void I2c_bitsend ( unsigned char Tx_data , unsigned char Ckbit );
void I2c_send ( unsigned char Bit_data );
unsigned char I2c_ackck ( void );
void I2c_sda_in ( void );
void I2c_sda_out ( unsigned char Data );
void Wait_timer ( unsigned short Wait_cnt );
```