

```

/*****
/*                               A/Dコンバータの制御                               */
*****/

#include "typedefine.h"
#include "iodefine.h"

#pragma section
/*****

static WORD wADdata[8];           /* 平均後のデータ */
static WORD wADaccu[8];          /* 積算値 */
static WORD wADcntr[8];          /* 積算カウンタ */

#pragma abs8(AD_Channel, AD_Enable)
static BYTE AD_Channel;          /* A/D変換中のチャンネル番号 (0~7ch) */
static BYTE AD_Enable;

/* A/D 変換初期化 *****/

void Init_AD(void)
{
    AD.ADCR.BYTE = 0x7E;          /* 外部トリガ無効 */
    AD.ADCSR.BYTE = 0x40;        /* ADIE=1:A/D割り込み有効、ADST=0:A/Dスタートせず、*/
                                /* SCAN=0:単一モード指定、CKS=0:134ステート、CH=0 */
    AD_Channel = 0;              /* A/D変換中のチャンネル番号を示す (初期=0ch) */
    AD_Enable = 1;
}

/* A/D変換開始 *****/
/* ※100mS周期割り込み処理から呼ばれる。 */

void Start_AD(void)
{
    if(AD_Enable == 0) return;
    AD.ADCSR.BIT.CH = AD_Channel;
    AD.ADCSR.BIT.ADST = 1;      /* A/D変換開始 */
}

/* A/D 割り込み処理 *****/

#pragma interrupt (Int_AD)
void Int_AD(void)
{
    WORD* pData;
    WORD* pAccu;
    WORD* pCntr;
    WORD wData;

    if(AD.ADCSR.BIT.ADF != 1)
        return;
    AD.ADCSR.BIT.ADF = 0;
    /* A/Dデータ読み込み */
    switch(AD_Channel & 0x03) {
        default:
        case 0:
            wData = AD.ADDRA;
            break;
        case 1:
            wData = AD.ADDRB;
            break;
        case 2:
            wData = AD.ADDRC;
            break;
        case 3:
            wData = AD.ADDRD;
            break;
    }
    /* 格納場所準備 */
    pData = &wADdata[AD_Channel];
    pAccu = &wADaccu[AD_Channel];
    pCntr = &wADcntr[AD_Channel];
    /* データの積算 */
    *pAccu += (WORD)(wData >> 6);
    /* データの平均化 (16回平均) */
    *pCntr++;
    if(*pCntr >= 16) {
        *pData = (WORD)((*pAccu + 8) / 16); /* 7捨8入して1/16 */
        *pAccu = 0;
        *pCntr = 0;
    }
    /* チャンネル番号更新 */
    AD_Channel++;
    if(8 <= AD_Channel) {
        AD_Channel = 0;
    }
}

```

/\* End of File \*/