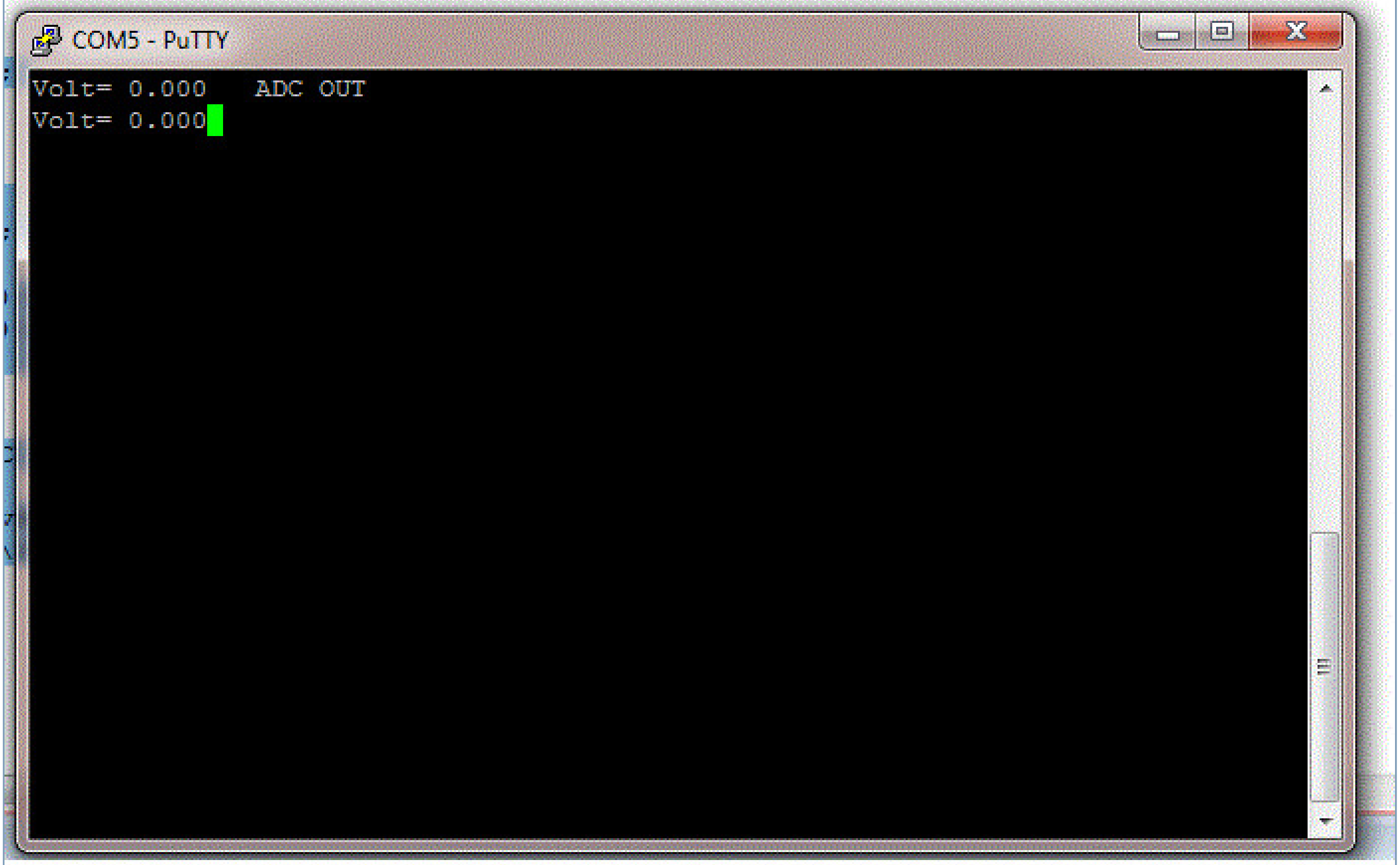
Dear All,

Some thing wrong with DMA operation please advice

```
/* Includes ------------------------------------------------------------------*/
#include "stm32f0xx.h"
#include "stm32f0xx_usart.h"
#include "stm32f0xx_adc.h"
#include "stm32f0xx_dma.h"
char name[27]="USART2 PA9TX -PA10RX dd- ";
int i,j;
__IO uint16_t  ADC1ConvertedValue = 0, ADC1ConvertedVoltage = 0;
__IO uint16_t  ADC1ConvertedDMAValue = 0, ADC1ConvertedDMAVoltage = 0;
int dem;
float volt;
char Result[32];

void RCC_Configuration(){
                RCC_AHBPeriphClockCmd(RCC_AHBPeriph_GPIOA, ENABLE);
                RCC_APB2PeriphClockCmd(RCC_APB2Periph_USART1,ENABLE);
                RCC_AHBPeriphClockCmd(RCC_AHBPeriph_GPIOC, ENABLE); /* GPIOC Periph clock enable
*/
                RCC_APB2PeriphClockCmd(RCC_APB2Periph_ADC1, ENABLE);  /* ADC1 Periph clock enable
*/
                RCC_AHBPeriphClockCmd(RCC_AHBPeriph_DMA1,ENABLE);

}
void GPIO_Configuration(){/* Configure USART1 GPIO pins:  Rx and Tx ----------------------------
*/

                GPIO_InitTypeDef GPIO_InitStructure;
                GPIO_DeInit(GPIOA);
                GPIO_PinAFConfig(GPIOA, GPIO_PinSource9, GPIO_AF_1);
                GPIO_PinAFConfig(GPIOA, GPIO_PinSource10, GPIO_AF_1);
                GPIO_InitStructure.GPIO_Pin    = GPIO_Pin_9 | GPIO_Pin_10;
                GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
                GPIO_InitStructure.GPIO_Mode    = GPIO_Mode_AF;
                GPIO_InitStructure.GPIO_OType = GPIO_OType_PP;
                GPIO_InitStructure.GPIO_PuPd    = GPIO_PuPd_UP;
                GPIO_Init(GPIOA, &GPIO_InitStructure);

                             /* Configure ADC GPIO Channel_10 as analog input */
                            GPIO_DeInit(GPIOC);
                            GPIO_InitStructure.GPIO_Pin = GPIO_Pin_0 ;
                            GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AN;
                            GPIO_InitStructure.GPIO_PuPd = GPIO_PuPd_NOPULL ;
                            GPIO_Init(GPIOC, &GPIO_InitStructure);



                GPIO_InitStructure.GPIO_Pin    = GPIO_Pin_9|GPIO_Pin_8;
                GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
                GPIO_InitStructure.GPIO_Mode    = GPIO_Mode_OUT;
                GPIO_InitStructure.GPIO_OType = GPIO_OType_PP;
                GPIO_InitStructure.GPIO_PuPd    = GPIO_PuPd_NOPULL;
                GPIO_Init(GPIOC, &GPIO_InitStructure);

    }
void delay(void){
        int time;
        for(time=0;time<1000000;time++);// tre khoang 1000000 lenh
}

void DMA_Configuration(){
                DMA_InitTypeDef DMA_InitStructure;
                __IO uint16_t ADCConvertedValue;
                /* DMA1 channel1 configuration ----------------------------------------------*/
                DMA_DeInit(DMA1_Channel1);
                DMA_InitStructure.DMA_PeripheralBaseAddr=0X40012440;//ADC_DR_DATA;
                DMA_InitStructure.DMA_MemoryBaseAddr=(uint32_t)&ADC1ConvertedDMAValue;
                DMA_InitStructure.DMA_DIR=DMA_DIR_PeripheralSRC;
                DMA_InitStructure.DMA_BufferSize=1;
                DMA_InitStructure.DMA_PeripheralInc=DMA_PeripheralInc_Disable;
                DMA_InitStructure.DMA_MemoryInc=DMA_MemoryInc_Disable;
                DMA_InitStructure.DMA_PeripheralDataSize=DMA_PeripheralDataSize_HalfWord;
                DMA_InitStructure.DMA_MemoryDataSize=DMA_MemoryDataSize_HalfWord;
                DMA_InitStructure.DMA_Mode=DMA_Mode_Circular;
                DMA_InitStructure.DMA_Priority=DMA_Priority_High;
                DMA_InitStructure.DMA_M2M=DMA_M2M_Disable;
                DMA_Init(DMA1_Channel1,&DMA_InitStructure);
                /* Enable DMA1 channel1 */
                DMA_Cmd(DMA1_Channel1,ENABLE);
}

void ADC_Configuration(void){
                ADC_InitTypeDef  ADC_InitStructure;
                ADC_DeInit(ADC1);                 /* ADCs DeInit */
                ADC_StructInit(&ADC_InitStructure);       /* Initialize ADC structure */
                ADC_InitStructure.ADC_Resolution = ADC_Resolution_12b;/* Configure the ADC1 in
continous mode withe a resolutuion equal to 12 bits  */
                ADC_InitStructure.ADC_ContinuousConvMode = ENABLE;
                ADC_InitStructure.ADC_ExternalTrigConvEdge = ADC_ExternalTrigConvEdge_None;
                ADC_InitStructure.ADC_DataAlign = ADC_DataAlign_Right;
                ADC_InitStructure.ADC_ScanDirection = ADC_ScanDirection_Upward;
                ADC_Init(ADC1, &ADC_InitStructure);
                ADC_ChannelConfig(ADC1, ADC_Channel_10 , ADC_SampleTime_239_5Cycles); /* Convert
the ADC1 Channel 10 with 239.5 Cycles as sampling time */
                ADC_GetCalibrationFactor(ADC1);
/* ADC Calibration */
                ADC_DMACmd(ADC1, ENABLE);
                ADC_Cmd(ADC1,
ENABLE);                                                                  /* Enable
ADCperipheral[PerIdx] */
                while(!ADC_GetFlagStatus(ADC1, ADC_FLAG_ADRDY));                    /* Wait the
ADRDY falg */
                ADC_StartOfConversion(ADC1);
/* ADC1 regular Software Start Conv */
}
void read_adc(){
        while(ADC_GetFlagStatus(ADC1, ADC_FLAG_EOC) == RESET);      /* Test EOC flag */
                ADC1ConvertedValue =ADC_GetConversionValue(ADC1);                        /* Get
ADC1 converted data */
            //ADC1ConvertedVoltage = (ADC1ConvertedValue *3300)/0xFFF;              /* Compute
the voltage */
}
void USART_Configuration(void){
                USART_InitTypeDef USART_InitStructure;
                USART_DeInit(USART1);
                USART_InitStructure.USART_BaudRate = 9600;
                USART_InitStructure.USART_WordLength = USART_WordLength_8b;
                USART_InitStructure.USART_StopBits = USART_StopBits_1;
                USART_InitStructure.USART_Parity = USART_Parity_No;
                USART_InitStructure.USART_HardwareFlowControl = USART_HardwareFlowControl_None;
                USART_InitStructure.USART_Mode = USART_Mode_Rx | USART_Mode_Tx;
                USART_Init(USART1, &USART_InitStructure);
                USART_Cmd(USART1,ENABLE);
}

void truyen(unsigned char uData){
  while(USART_GetFlagStatus(USART1, USART_FLAG_TXE)==RESET);
  USART_SendData(USART1,uData);
}

void Print(const char *s){
        int i,j;
        j=strlen(s);
    for(i=0;i<j;i++)
```

```
        {
            truyen(s[i]);
        }
}
int main(void){
        RCC_Configuration();
        GPIO_Configuration();
        ADC_Configuration();
        USART_Configuration();
        //DMA_Configuration();
    sprintf(&Result[0],"   ADC OUT \r\n");
    Print(Result);
while(1){
    read_adc();// doc ADC
    //DMA_Configuration();
    volt=(float)(ADC1ConvertedDMAValue*3.3)/0x0fff;
    sprintf(&Result[0],"\rVolt= %1.3f"    ,volt);
    Print(Result);
     delay();

    }
}

Result is zero
```



```
Volt= 0.000   ADC OUT
Volt= 0.000
```