CAN 控制器的寄存器简介

## 一：整体配置寄存器

**CNTL**：

该寄存器主要控制着整个控制器的初始化工作的设定，各种中断使能和关闭，模式设定。

**STAT：**

反应CAN的各种状态属性

**INT:**

反应各种中断

**EC：**

错误计数寄存器，反应传输出错的次数

**BT：**

Bit timing ，负责设置CAN 的波特率的

**TEST**：

负责设置测试模式下的几种不同模式

**BRPE：**

对CAN 时钟进行预分频。

CAN\_CLK：

CAN\_CLK = PCLK/(CLKDIVVAL + 1)

PCLK 是 peripheral clock。

## 二：Interface Registers

**IF1/2\_CMDREQ:**

该寄存器负责由CPU写，每次写完该寄存器，CAN 就会开始一次发送数据。发送完毕，寄存器的 BUSY 位清零。可以开始下一轮发送。

**IF1/2\_CMDMSK\_W/R:**

该寄存器负责在Interface Registers 和 Message Objects 之间传输数据时，选择目标位置屏蔽哪些位，从而不受到源对应位置的影响。比如是否向目的地址（比如interface registers 向 Message Object 传输。），传输 MSK，ARB等等对应位。

**IF1/2\_MSK1/2:**

负责用来设置是否屏蔽 CAN\_ID 的相应为。

**IF1/2\_ARB1/2:**

用来设置 CAN Message 的 arbitration 场。

**IF1/2\_MCTRL:**

用来设置 Message RAM 中某些控制。

**IF1/2\_DA1/2 和 IF1/2\_DB1/2:**

负责存放需要传送的数据或者接受的数据，一共八个字节。

## 三：Message ROM 中相关寄存器

**TXRQST1/2：**

包含了 Message Object 的 TXRQST 位 0-31

**ND1/2 :**

包含了Message Object 的 NEWDAT 位 ：0-31

**IR1/2：**

Message Object 是否为中断源，为0的位会被 Message Handler 忽略，不时中断源，而非零的会被处理。

**MSGVAL1/2：**

为 0 的位暗示该 Message Object 无效，为1为有效。

## 四：CAN控制器初始化流程：

