;File: adc42go.a51

;Author: Eckart Hartmann Date:05/10/2003

; Development progress: Adc842.df

;

;AdcGo==========Start conversion(s).

;C Function prototype: char AdcGo(char cChan, char cMode);

;Description of Function: Start conversion on channel cChan with cMode mode.

;User interface: Set up the ADC using AdcCfg().

; Set cChan to channel to convert.

; For single conversion set cMode = 1, for continuous conversion

; set cMode = 2 and for DMA conversion see note below.

; Call AdcGo() which starts conversion according to mode bits.

; Returns (corrected) cChan.

; Note: for DMA conversion prepare the destination RAM as

; described in the data sheet. Then set cMode to 4 and call

; AdcGo(). Then set cMode to 6 for continuous conversions or 5

; for single conversions and call AdcGo again to start conversions.

;Robustness: AdcBsy() should be used to check for completion.

; Invalid cChan will be changed to 12.

;Side effects: Overwrites a, c, P.

;

NAME ADCGO

$NOMOD51

$IC(..kei842.inc) ; Parameter passing registers for Keil .

$IC(..kei842.dat) ; SFR definition for Keil .

;

public \_AdcGo

;

?PR?\_AdcGo?ADCGO SEGMENT CODE

RSEG ?PR?\_AdcGo?ADCGO

;

\_AdcGo: mov a,cP1l ;if(cChan>12)

add a,#0f3h

jnc AdGo

mov cP1l,#12 ; cChan = 12;

AdGo: mov ADCCON2,cP1l ;Set mux.

mov a,cP2lc ;ADCCON2 = cChan+(cMode<<4);

anl a,#7

swap a

orl a,cP1l

mov ADCCON2,a ; //Start conversion

ret

;

;Function End==========================================================Function End

END