Undocumented Features

This page is still incomplete. Please contribute by clicking on the `Edit this page on GitHub` link at the bottom of this page.

Table of Contents

1. Introduction
2. 16 MHz System Clock (All MCUs)
3. Resistance to Frequency Converter (PFS154 and PFS173)
4. 11 Bit ADC (PFS173)

Introduction

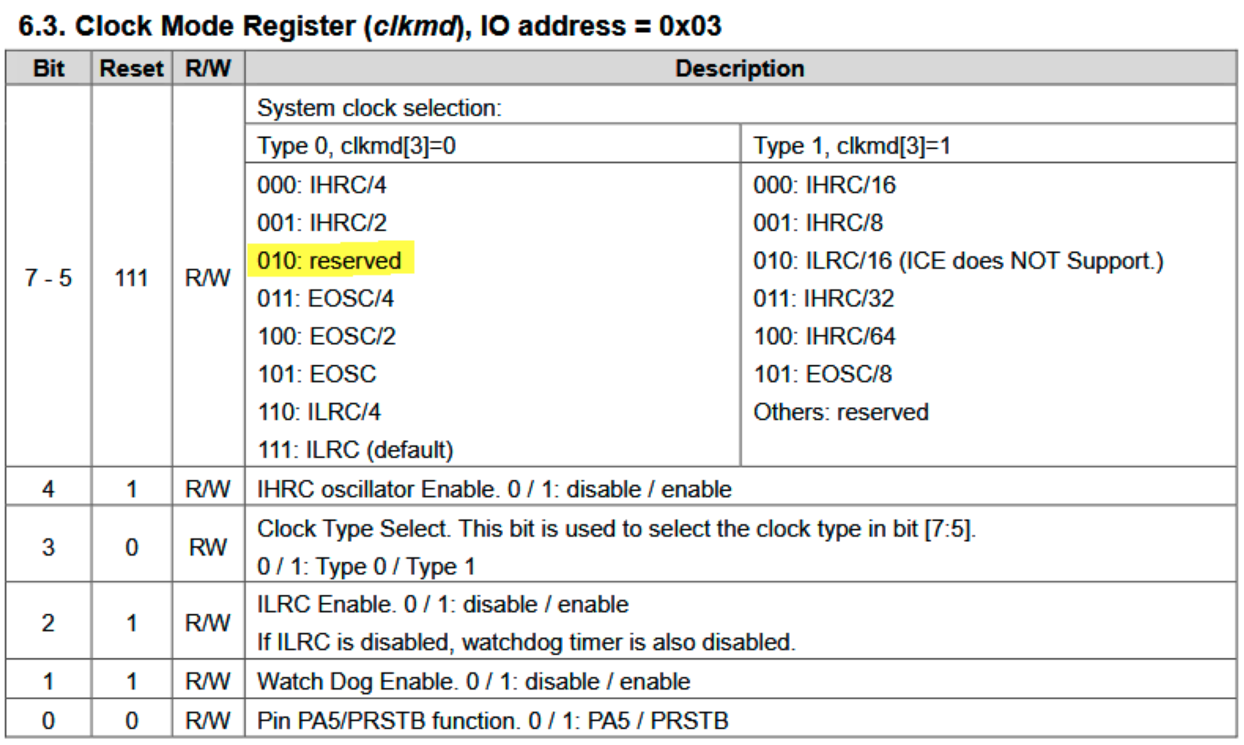
This page lists features that were intentionally removed from the documentation by Padauk, but are still available in several microcontrollers. In some cases the reason for this seems to be marginal performance, in other cases the reason is not entirely clear.

Please keep in mind that using these features in your design comes with some risks:

- They may cease to work under certain circumstances, for example increased temperature.
- It is likely that Padauk is not testing these features in their production testing. Therefore you may encounter devices where these features are defective.
- Padauk may remove these features from their products without notice.

16 MHz System Clock (All MCUs)

The CPU system clock is configured in the *clkmd* register that is present on all MCUs. The highest clock speed setting is 8 MHz, derived from dividing the clock speed of the internal high-speed RC oscillator (IHRC) by two. A setting for a clock divider of one is absent from all devices. Notably there is a bit combination marked as *reserved*.



6.3. Clock Mode Register (*clkmd*), IO address = 0x03

Bit	Reset	R/W	Description	
7 - 5	111	R/W	System clock selection:	
			Type 0, clkmd[3]=0	Type 1, clkmd[3]=1
			000: IHRC/4 001: IHRC/2 010: reserved 011: EOSC/4 100: EOSC/2 101: EOSC 110: ILRC/4 111: ILRC (default)	000: IHRC/16 001: IHRC/8 010: ILRC/16 (ICE does NOT Support.) 011: IHRC/32 100: IHRC/64 101: EOSC/8 Others: reserved
4	1	R/W	IHRC oscillator Enable. 0 / 1: disable / enable	
3	0	RW	Clock Type Select. This bit is used to select the clock type in bit [7:5]. 0 / 1: Type 0 / Type 1	
2	1	R/W	ILRC Enable. 0 / 1: disable / enable If ILRC is disabled, watchdog timer is also disabled.	
1	1	R/W	Watch Dog Enable. 0 / 1: disable / enable	
0	0	R/W	Pin PA5/PRSTB function. 0 / 1: PA5 / PRSTB	

Selecting this bit combination will set the clock divider to one and will therefore allow setting the system clock to 16 MHz. One issue is that the MCU is only stable at this clock at a relatively high supply voltage (VDD) close to 5V. For example, the minimum voltage where the PFS173 can be clocked at 16 MHz is 4.0V. Most likely even higher voltages are required at temperatures higher than room temperature. It is possible that 16 MHz core clock will cease to work if a certain environmental temperature is exceed.

Therefore, it is necessary to ensure that the supply voltage has reached a sufficiently high level before activating the 16 MHz clock setting to prevent crashing of the MCU. This can be achieved by activating the low-voltage-reset feature (LVR), see tutorial page. Discussion here and following posts.

Example code for PFS154 and PFS173:

```
unsigned char _sdcc_external_startup(void)
{
        MISCLVR = MISCLVR_4V;
        CLKMD = CLKMD_IHRC_DIV | CLKMD_ENABLE_IHRC;  // 16/1=16 Mhz main c
        // Optional: add clock calibration here
        return 0; // perform normal initialization
}
```

Alternatively you can use the `EASY_PDK_INIT_SYSCLOCK_16MHZ()` define provided with the Free-PDK environment.

Resistance to Frequency Converter

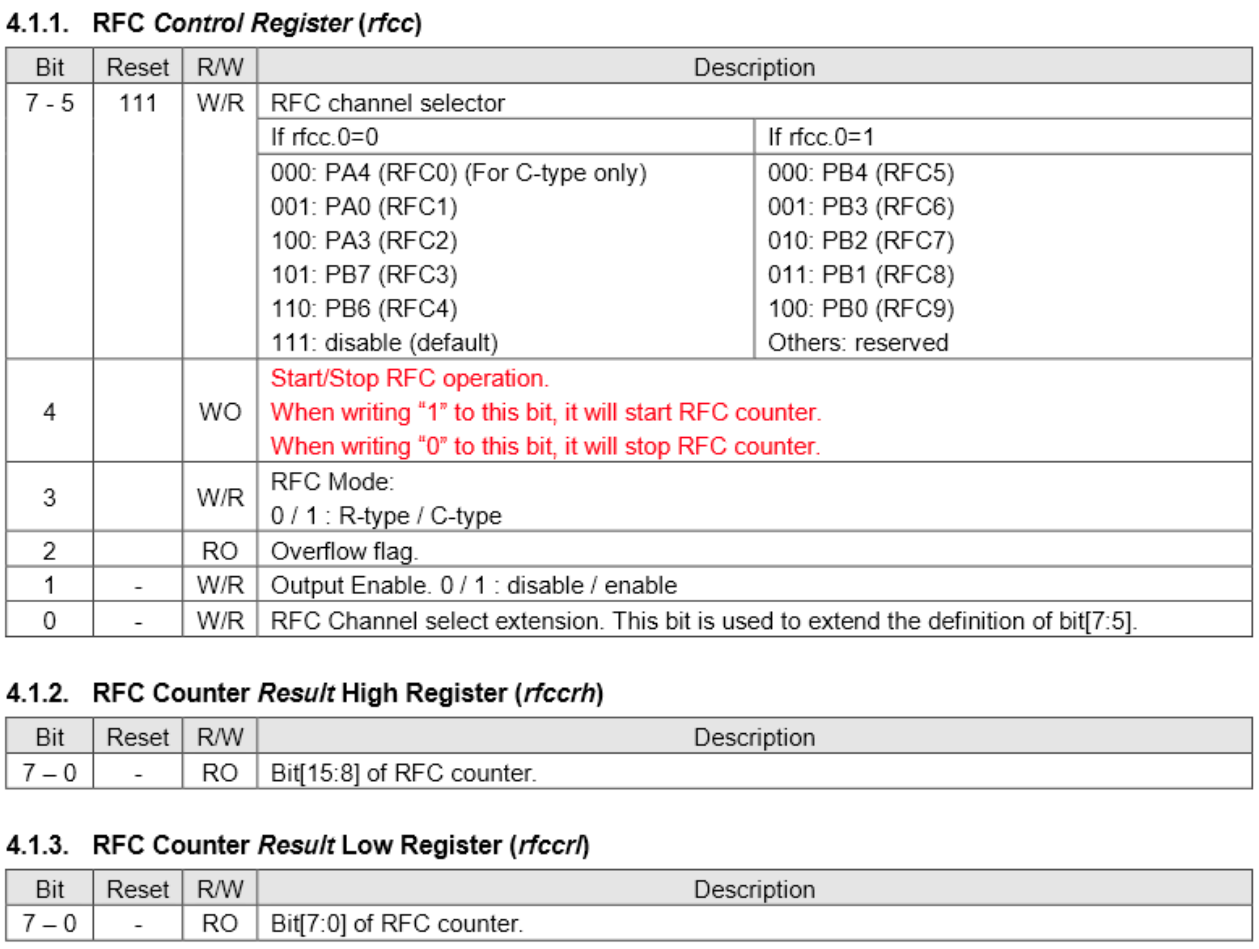
(PFS154 and PFS173)

The resistance to frequency converter (RFC) is mentioned as a footnote in Paudaks product line card, but no devices are listed with this feature. A description of the peripheral was found in a data-sheet for a discontinued MCU at a distributor. The document in question was saved here and describes the operation in detail.

This peripheral measures a time constant defined by an externally connected resistor-capacitor pair. It can be used for various sensing applications where a change in resistance or capacitance needs to be measured.

It was noted that the RFC is present in at least the current revisions of PFS154 and PFS173, despite not being mentioned in the datasheet. More details can be found here. The RFC is located at I/O 0x36-0x38 in the PFS154 and 0x2d-0x2f in the PFS173.

The RFC consists of a relaxation oscillator and a 16-bit counter. The frequency of the oscillator is defined by an RC pair connected to a selectable I/O pin. The counter will count the number of pulses after being started by setting a bit in the I/O register. The counter value after a defined time is representative of oscillator-frequency and hence of the RC time constant.



4.1.1. RFC *Control Register* (*rfcc*)

Bit	Reset	R/W	Description	
7 - 5	111	W/R	RFC channel selector	
			If rfcc.0=0	If rfcc.0=1
			000: PA4 (RFC0) (For C-type only)	000: PB4 (RFC5)
			001: PA0 (RFC1)	001: PB3 (RFC6)
			100: PA3 (RFC2)	010: PB2 (RFC7)
			101: PB7 (RFC3)	011: PB1 (RFC8)
			110: PB6 (RFC4)	100: PB0 (RFC9)
			111: disable (default)	Others: reserved
4		WO	Start/Stop RFC operation. When writing "1" to this bit, it will start RFC counter. When writing "0" to this bit, it will stop RFC counter.	
3		W/R	RFC Mode: 0 / 1 : R-type / C-type	
2		RO	Overflow flag.	
1	-	W/R	Output Enable. 0 / 1 : disable / enable	
0	-	W/R	RFC Channel select extension. This bit is used to extend the definition of bit[7:5].	

4.1.2. RFC Counter *Result* High Register (*rfccrh*)

Bit	Reset	R/W	Description
7 – 0	-	RO	Bit[15:8] of RFC counter.

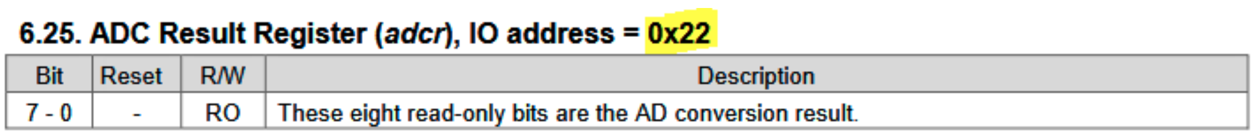
4.1.3. RFC Counter *Result* Low Register (*rfccrl*)

Bit	Reset	R/W	Description
7 – 0	-	RO	Bit[7:0] of RFC counter.

You can find some electrical characterization results and usage examples in this HaD.io project.

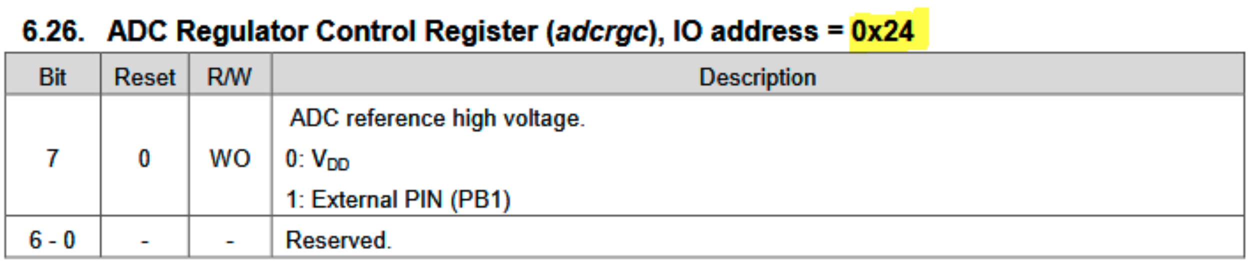
11 Bit ADC (PFS173)

Register 0x23, which is part of the ADC peripheral I/O block in the PFS173, is notably absent from the documentation of the I/O space.



6.25. ADC Result Register (*adcr*), IO address = 0x22

Bit	Reset	R/W	Description
7 - 0	-	RO	These eight read-only bits are the AD conversion result.



6.26. ADC Regulator Control Register (*adcrgc*), IO address = 0x24

Bit	Reset	R/W	Description
7	0	WO	ADC reference high voltage. 0: V_{DD} 1: External PIN (PB1)
6 - 0	-	-	Reserved.

Probing this register shows that bits 7-5 are readable and seem to represent three additional least significant bits of the ADC result. It is not clear why these bits were left undocumented. Possibly they were deemed too noisy, or malfunction under certain settings. More investigation is needed here.

The undocumented register `ADCRL` can be accessed like this:

```
        __sfr __at(0x23)          _adcrl;
        #define ADCRL             _adcrl

        uint8_t adcout  =ADCR;                    // Read most significan
        uint8_t adcoutl =ADCRL>>5;                // Read three hidden bi
        uint16_t adc=(adcout<<3)|adcoutl;         // adc contains 11 bit
```

Edit this page on GitHub

Checkout /doc-style for more information on some of the special Markdown formatting features we use.

Free PDK

Free PDK	free-pdk	Free PDK is an effort to create an open source alternative to the proprietary Padauk µC programmer, as well as adding support to SDCC for Padauk µCs.