

# OSCCAL

Bei einigen PICs (12F629 / 12F675 / 16F630 / 16F676) gibt es eine Besonderheit, die unter Umständen zu Problemen führen kann: **der OSCCAL**.

## Zweck von OSCCAL

Es handelt sich um einen OSCillator CALibration Wert. Da der PIC mit einem internen Oszillator betrieben werden kann (dies spart einen externen Quarz), wird dieser Wert bereits in der Chip-Fabrik an einer Speicherstelle programmiert. Dadurch können Toleranzen in der Taktfrequenz abgefangen werden. Der 16F627 besitzt den OSCCAL übrigens nicht!

## Warum wird der OSCCAL verwendet?

Die Verwendung des OSCCAL-Wertes wurde notwendig, damit alle Kombinationen von Dekoder und Zentrale zusammen laufen. Denn auch die Zentrale hat gewisse Toleranzen. Wenn nun eine langsame „Scharping“-Zentrale mit einem hektischen „Waalkes“-Dekoder zusammen funktionieren soll, so verstehen sich die beiden nicht mehr. Die Zentrale sendet einfach zu langsam.

## Wie funktioniert das?

Der OSCCAL liegt in der letzten Speicherstelle des Programm-Speichers und wurde dort von der Chip-Fabrik einprogrammiert. Beim Start des PIC-Programmes wird dieser Wert dann ausgelesen.

## Wie kommt es zu Problemen?

Beim Brennen des PIC wird der Programmspeicher mit dem eigentlichen Programm geladen. Und genau hier sieht man schon die Gefahr: dort liegt ja schon der OSCCAL rum. Um diesen nicht zu überschreiben, wird dieser von einer guten PC-Programmier-Software zuerst gelesen und dadurch gerettet. Eine schlechte PC-Programmier-Software überschreibt diese Speicherstelle einfach und der OSCCAL ist futsch. Aber auch mit der guten PC-Programmier-Software kann es vorkommen, dass der OSCCAL verloren geht. Wird der Programmier-Vorgang z.B. gestört, so ist evtl. der Speicher schon gelöscht, aber der OSCCAL-Wert noch nicht wieder zurück-programmiert worden. In diesem Fall: OSCCAL schnell notieren und später wieder einprogrammieren.

Unsere Software (PICkit 3 Programmer) zeigt den OSCCAL-Wert im Kopf an - dieser muss also beim ersten Auslesen des PICs notiert werden und zur Not später zurückgesetzt werden.

**3 PICkit 3 Programmer - BUR171726474**

File Device Family Programmer Tools View Help

Midrange/Standard Configuration

Device: PIC12F629 Configuration: 01C4

User IDs: FF FF FF FF

Checksum: 568D OSCCAL: 3434 BandGap: 0000

Reading device: Program Memory... EE... UserIDs... Config... Done.

**MICROCHIP**

VDD PICkit 3  
 On  
 /MCLR

5,0

Read Write Verify Erase Blank Check

**Program Memory**

Enabled Hex Only Source: Read from PIC12F629

000	0183	018B	018A	28EE	00A0	0E03	0183	00A1	^
008	190B	2816	30B3	0081	0825	0782	282B	2830	
010	2823	2822	282F	2822	2850	2899	30B3	0081	
018	0825	0782	2823	2822	2834	2832	2838	284C	
020	2822	2822	01A5	0805	100B	110B	0E21	0083	
028	0EA0	0E20	0009	3016	00A7	0AA5	2823	14A5	
030	0BA7	2823	0AA5	2823	302B	00A6	01A4	2847	
038	0AA5	0BA7	2823	1DA6	2822	0804	00A2	0826	
040	0084	0823	0080	06A4	0AA6	0822	0084	3009	
048	00A7	3003	00A5	2823	1003	0DA3	03A5	2823	
050	1403	0DA3	10A5	2823	0782	3401	3402	3404	
058	3408	3410	3420	3440	3480	0E31	00D9	0C59	▼

**EEPROM Data**

Enabled Hex Only

00	03	FF	FF	FF	FF	FF	1E	4A	FF	FF	FF	FF	00	03	FF	FF	^
10	C0	64	FF	06	FF	FF	FF										
20	00	00	00	00	0F	0F	0F	0F	FF								
30	FF	0C	0C	0C	0C	FF	▼										

Auto Import Hex + Write Device

Read Device + Export Hex File

**PICkit™ 3**

[PIC, OSCCAL](#)

From:  
<https://wiki.modellbahn-anlage.de/> - **Wiki der Modellbahn-Anlage.de**



Permanent link:  
<https://wiki.modellbahn-anlage.de/elektronik/osccal>

Last update: **07.05.2025 15:05**