**Rekenkundige operatoren**

Proton PIC Basic V3.0

Operatoren die geel zijn verwijzen naar een voorbeeld in de minicursus PIC programmeren in PIC Basic.  
Door er met de muis op te klikken kom je op die plek in de cursus.

Om er zeker van te zijn dat berekeningen in de juiste volgorde worden uitgevoerd moeten groepen berekeningen tussen haakjes worden gezet:

|  |
| --- |
| A = ((B - C) \* (D + E)) / F |

|  |  |
| --- | --- |
| **+** | Telt variabelen en/of constanten bij elkaar op. |
| **-** | Trekt variabelen en/of constanten van elkaar af. |
| [**\***](http://www.picbasic.nl/beginners1.htm#vermenigvuldig) | Vermenigvuldigd variabelen en/of constanten met elkaar. |
| **\*\*** | Levert de hoogste 16 bits van een 16-bit vermenigvuldigingsuitkomst. |
| **\*/** | Levert de middelste 16 bits van een 16-bit vermenigvuldigingsuitkomst. |
| [**/**](http://www.picbasic.nl/beginners1.htm#delen) | Deelt variabelen en/of constanten. |
| [**//**](http://www.picbasic.nl/beginners4a.htm#Modulus) | Modulus; levert de restwaarde van een delingsuitkomst. |
| [**&**](http://www.picbasic.nl/beginners3.htm#bitwise) | Bitwise AND. |
| [**|**](http://www.picbasic.nl/beginners3.htm#bitwise) | Bitwise OR. |
| [**^**](http://www.picbasic.nl/beginners3.htm#bitwise) | Bitwise XOR. |
| [**~**](http://www.picbasic.nl/beginners7.htm#NOT) | Bitwise NOT. *((complement) inverteert de bit(s) van een getal)* |
| **<<** | Schuift de bits van een waarde een opgegeven aantal keer naar links. |
| **>>** | Schuift de bits van een waarde een opgegeven aantal keer naar rechts. |
| [**ABS**](http://www.picbasic.nl/beginners5.htm#ABS) | Levert het absolute getal van een waarde. |
| **ACOS** | Levert de ARC COSINUS van een waarde in RADIALEN. |
| **ASIN** | Levert de ARC SINUS van een waarde in RADIALEN. |
| **ATAN** | Levert de ARC TANGENS van een waarde in RADIALEN. |
| **COS** | Levert de COSINUS van een waarde in RADIALEN. |
| **DCD** | 2 n -machtsverheffen van een 4-bits waarde. |
| **DIG** | Levert de opgegeven decimaal van een positief getal. |
| **EXP** | Levert de exponentionele functie van een waarde |
| **LOG** | Levert het natuurlijke logaritme van een waarde. |
| **LOG10** | Levert het logaritme van een waarde. |
| **MAX** | Levert het maximum van twee waarden. |
| **MIN** | Levert het minimum van twee waarden. |
| **NCD** | Prioriteits encoder van een 16-bit waarde. |
| **POW** | Berekent variabele naar power variabele. |
| **REV** | Keert de volgorde van de laagste bits om. |
| **SIN** | Levert de SINUS van een waarde in RADIALEN. |
| **SQR** | Levert de vierkantswortel van een waarde. |
| **TAN** | Levert de TANGENS van een waarde in RADIALEN. |
| **DIV32** | 15-bit x 31-bit deling. (Alleen voor PBP compatibiliteit) |

**Booleaanse logische operatoren**

Logische operatoren (Booleaanse functies) kunnen worden gebruikt om uitkomsten van meerdere vergelijkingen te testen op waar (TRUE) of niet waar (FALSE), het resultaat kan worden gebruikt om een beslissing te maken betreffende het programmaverloop.

|  |  |
| --- | --- |
| [**AND**](http://www.picbasic.nl/beginners2.htm#ANDORXOR) | EN |
| [**OR**](http://www.picbasic.nl/beginners2.htm#ANDORXOR) | OF |
| [**XOR**](http://www.picbasic.nl/beginners2.htm#ANDORXOR) | Exclusief OF |
| **NOT** | Inverteert de uitkomst van een conditie (verandert TRUE in FALSE, FALSE in TRUE) |

**Relationele operatoren**

Relationele operatoren worden gebruikt om twee waarden met elkaar te vergelijken, het resultaat kan worden gebruikt om een beslissing te maken betreffende het programmaverloop.

|  |  |
| --- | --- |
| **=** | Gelijk aan |
| **==** | Gelijk aan (Hetzelfde als **=**) |
| **<>** | Ongelijk aan |
| **!=** | Ongelijk aan (Hetzelfde als **<>**) |
| **<** | Kleiner dan |
| **>** | Groter dan |
| **<=** | Kleiner dan of gelijk aan (**Opmerking**: <= *is correct*, =< *is incorrect!*) |
| **>=** | Groter dan of gelijk aan (**Opmerking**: >= *is correct*, => *is incorrect!*) |

**Programma start A= A-**

**Symbol** LED1 = PORTA.1 *;Poort A.0 heeft nu de naam LED1*

**Symbol** LED2 = PORTA.0 *;Poort A.1 heeft nu de naam LED2*

**Symbol** LED3 = PORTA.2 *;Poort A.2 heeft nu de naam LED3*

**Symbol** S1 = PORTB.0 *;Poort B.0 heeft nu de naam S1*

**Symbol** S2 = PORTB.1 *;Poort B.1 heeft nu de naam S2*

**Symbol** S3 = PORTB.2 *;Poort B.2 heeft nu de naam S3*

**PORTB\_PULLUPS On** *;On-chip pull-up weerstanden actief*

**Clear** *;Wis alle RAM geheugen*

*;Hoofdprogramma*

**While** 1 = 1 *;Oneindige lus*

**If** S1 = AAN **Then** *;Als start*

**High** LED1 *;LED1 aan output 1 hoog A plus*

**EndIf** *;Einde IF...THEN...ELSE blok van S1*

**If** S2 = AAN **Then**

**Low** LED1 *; a1 signaal*

**DelayMS** 500

**High** LED2 *; amin output*

**EndIf**

**If** S3 = AAN **Then** *;Als S3=AAN dan...*

**Low** LED2 *;LED3 aanzetten*

**DelayMS** 500 *;Tijd van LED3 aan*

**Wend** *;...in dit lusje blijven wachten*

**EndIf** *;Einde IF...THEN blok van S3*

**Wend**

**End** *;Einde programma*