

Sprach-Referenz



Die Programmiersprache von Arduino kann in drei Hauptteile unterteilt werden: Struktur, Werte (Variablen und Konstanten) und Funktionen.

FUNKTIONEN

Zum Steuern des Arduino-Boards und zur Durchführung von Berechnungen.

Digital I/O digitalRead() digitalWrite() pinMode()	Mathematik abs() constrain() map() max() min() pow() sq() sqrt()	Zufallszahlen random() randomSeed() Bits und Bytes bit() bitClear() bitRead() bitSet() bitWrite() highByte() lowByte()
Analog I/O analogRead() analogReference() analogWrite()	Trigonometrie cos() sin() tan()	Externe Interrupts attachInterrupt() detachInterrupt()
Zero-, Due-, MKR-Familie analogReadResolution() analogWriteResolution()		Interrupts interrupts() noInterrupts()
Advanced I/O noTone() pulseIn() pulseInLong() shiftIn() shiftOut() tone()	Zeichen isAlpha() isAlphaNumeric() isAscii() isControl() isDigit() isGraph() isHexadecimalDigit() isLowerCase() isPrintable() isPunct() isSpace() isUpperCase() isWhitespace()	Kommunikation Serial Stream USB Keyboard Mouse

VARIABLEN

Arduino-Datentypen und -konstanten.

Konstanten Gleitkommazahlkonstanten Integer-Konstanten Konstanten Umwandlung (unsigned int) (unsigned long) byte() char() float() int() long() word()	Datentypen String() array bool boolean byte char double float int long short size_t string unsigned char unsigned int unsigned long void word
---	--

Variablenscope und Bezeichner

const
scope
static
volatile

Hilfsmittel

PROGMEM
sizeof()

STRUKTUR

Die Elemente der Arduino (C++) Programmiersprache.

Sketch loop() setup() Kontrollstruktur break continue do...while else for goto if return switch...case while Weitere Syntax #define (define) #include (include) /* */ (Kommentarblock) // (Einzeiliger Kommentar) ; (Semikolon) { } (Geschweifte Klammern)	Arithmetische Operatoren % (Modulo) * (Multiplikation) + (Addition) - (Subtraktion) / (Division) = (Zuweisung) Vergleichsoperatoren != (Ungleich) < (Kleiner) <= (Kleiner oder gleich) == (Gleich) > (Größer) >= (Größer oder gleich) Boolsche Operatoren ! (Logisches NICHT) && (Logisches UND) (Logisches ODER)	Zeiger-Operatoren & (Referenzoperator) * (Dereferenzoperator) Bitweise Operatoren & (Bitweises UND) << (Bitshift links) >> (Bitshift rechts) ^ (Bitweises XOR) (Bitweises OR) ~ (Bitweises NICHT) Verbundoperatoren %= (Zusammengesetzter Rest) &= (Zusammengesetztes Bitweises AND) *= (Zusammengesetzte Multiplikation) ++ (Inkrement) += (Zusammengesetzte Addition) -- (Dekrement) -= (Zusammengesetzte Subtraktion) /= (Zusammengesetzte Division) = (Zusammengesetztes Bitweises ODER)
---	--	---