crc64

package standard library

Version: go1.20.1 Latest | Published: Feb 14, 2023 | License: BSD-3-Clause | Imports: 3 | Imported by: 1,093

Details Valid go.mod file Redistributable license Tagged version Stable version

Learn more

Repository cs.opensource.google/go/go

Links Report a Vulnerability

Documentation

Documentation

Overview

Package crc64 implements the 64-bit cyclic redundancy check, or CRC-64, checksum. See https://en.wikipedia.org/wiki/Cyclic_redundancy_check for information.

Index

Constants

View Source

```
const (
    // The ISO polynomial, defined in ISO 3309 and used in HDLC.
    ISO = 0xD800000000000000

    // The ECMA polynomial, defined in ECMA 182.
    ECMA = 0xC96C5795D7870F42
)
```

Predefined polynomials.

View Source

```
const Size = 8
```

The size of a CRC-64 checksum in bytes.

Variables

This section is empty.

Functions

func Checksum

```
func Checksum(data []byte, tab *Table) uint64
```

Checksum returns the CRC-64 checksum of data using the polynomial represented by the Table.

func New

```
func New(tab *Table) hash.Hash64
```

New creates a new hash.Hash64 computing the CRC-64 checksum using the polynomial represented by the Table. Its Sum method will lay the value out in big-endian byte order. The returned Hash64 also implements encoding.BinaryMarshaler and encoding.BinaryUnmarshaler to marshal and unmarshal the internal state of the hash.

func Update

```
func Update(crc uint64, tab *Table, p []byte) uint64
```

Update returns the result of adding the bytes in p to the crc.

Types

type Table

```
type Table [256]uint64
```

Table is a 256-word table representing the polynomial for efficient processing.

func MakeTable

```
func MakeTable(poly uint64) *Table
```

MakeTable returns a Table constructed from the specified polynomial. The contents of this Table must not be modified.

Source Files

View all

crc64.go

Report an Issue

Google