

Golang Embedded

Go 语言嵌入式应用

Golang 简介

一种语言

- 编译型
- 并发型
- 垃圾回收
- 简单易学

作者们

- Robert Griesemer
- Rob Pike
- Ken Thompson

Golang 简介

设计目标

- 简单
- 速度够快
- 快速编译
- 函数式编程
- 尽量少的“坑”
- 新的集成模式

Golang 简介

Hello World

```
package main
import "fmt"

func main() {
    fmt.Println("Hello, World")
}
```

Golang 简介

Goroutines 协程

```
func Announce(message string, delay time.Duration) {  
    go func() {  
        time.Sleep(delay)  
        fmt.Println(message)  
    }() // Note the parentheses - must call the function.  
}
```

Golang 简介

Channels

```
c := make(chan int) // Allocate a channel.  
// Start the sort in a goroutine; when it completes, signal on  
the channel.  
go func() {  
    list.Sort()  
    c <- 1 // Send a signal; value does not matter.  
}()  
doSomethingForAWhile()  
<-c // Wait for sort to finish; discard sent value.
```

嵌入式Golang的优势

编译型语言

- 不需要虚拟机
- 和C/C++联合编译
- 性能足够好

开发效率

- 类似脚本语言的编码速度
- 剔除C/C++容易出错的部分
- 良好的调试接口

但是上面这些都不是最主要的！

嵌入式Golang的优势

Concurrent

并发编程语言级别的支持

- Goroutine
 - 轻量级, 高并发
- Channel
 - 用来Goroutine之间的通信
 - 用于中断/数据的传递

嵌入式Golang的优势

```
go func() {  
    data <- serial_1 // receive data from serial 1  
    sock <- data    // Send the data to socket.  
}()
```

嵌入式Golang的优势

其他优势

- 不需要交叉编译
- dwarf3 debug info, gdb 调试便利
- BSD License

嵌入式Golang的不足

- Runtime 体积略大
- 内存消耗较多
- 缺乏Thumb2和soft float支持

嵌入式Golang的现状

- Gccgo + RTEMS



Thanks!