lowRISC多核开源平台和RISC-V在中国的发展

中国科学院信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING,CAS

宋威

2018年4月25日

自我介绍

宋威

- 曼彻斯特大学计算机博士，剑桥大学博士后，中国科学院信息工程研究所副研究员。
- lowRISC项目的初创成员，硬件平台主要开发人员。
  - lowRISC是剑桥大学计算机学院的一个非盈利组织。
  - RISC-V基金会的初创成员之一。
  - 是在美国之外唯一提供Rocket片上系统的开源平台。
  - 第一个实现纯RISC-V开发平台的组织。
- 信息工程研究所安全处理器架构小组成员。
  - 2017年11月回国。
  - 改造现有处理器流水线、缓存和片上总线。
  - 防止控制流劫持、信息泄露等等攻击。
- CNRV的主要成员
  - CNRV通过微信群、网站和公众号，与大部分国内RISC-V的使用者、参与者和爱好者建立了联系，包括软件/IC/硬件开发者、爱好者、相关公司高管、高校研究人员以及投资人等。
  - 每两周通过网站和公众号发布《CNRV双周简报》，向中文读者介绍和总结RISC-V相关领域的最新进展，可理解为RISC-V大参考～

内容简介

- 什么是RISC-V
  RISC指令集、计算机体系结构与微结构、指令集分类、扩展等等。
- RISC-V的组织构成
  RISC-V基金会、RISC-V的商业模式、RISC-V商标和兼容性测试。
- lowRISC的Rocket片上系统平台
  lowRISC的历史，lowRISC的商业模式，现有平台特性、成果等等。
- RISC-V在中国的发展现状(CNRV的角度)
- 总结

什么是RISC-V

- RISC-V是加州伯克利大学研发的第五代精简指令集（RISC）。
  - Krste Asanovic，加州伯克利教授，RISC-V基金会主席。
  - David Patterson，量化分析一书的作者，RISC概念的提出者，图灵奖获得者。
  - Andrew Waterman，SiFive首席工程师，Rocket处理器的作者。
  - Yunsup Lee，SiFive首席技术官，Rocket处理器的作者。

- RISC-V是一个开放指令集。
  - 体系结构≠处理器设计。
  - RISC-V只定义了软件和硬件的接口，而并不定义具体处理器实现。
  - RISC-V是开放的，免费的，但是遵循RISC-V标准设计的CPU不一定开放并免费。

- RISC-V是一个全新的指令集
  - RISC-V在加州伯克利内部使用了几年，然后于2014年公开。
  - RISC-V指令集的设计吸取了过去几十年来各类指令集（Alpha，ARM, MIPS, Power，x86等）成功和失败的经验教训
  - RISC-V是一个模块化可扩展的指令集。能够支撑面向从IoT到高性能计算的所有应用。

RISC-V的设计思路

- 简单的硬件支持最核心的指令功能。
  - 基本指令集的编码设计支持高效的指令解码硬件。
  - 所有基本指令都是单周期指令（不支持多周期指令）。
  - 不使用条件执行指令，简化处理器控制逻辑。
  - 不支持缓存直接控制指令，简化一致性处理。
  - 支持安全的新特性（SMAP/SMEP/KPTI）。
  - 简单的通过16/32-bit混和指令长度就可以达到和X86、ARMv8同样的指令密度

- 可扩展的指令集设计
  - 两个基本指令集（RV32和RV64）支持所有的基本功能。
  - C：压缩指令集,16比特比特指令，面向超小系统和高性能系统。
  - A：原子操作指令集；M：整数乘除指令集。
  - F：单精度浮点计算；D：64位双精度浮点计算；Q：128位浮点计算指令集。
  - B：比特处理指令集；P：Packed SIMD指令集；V：vector指令集。
  - J：动态翻译（Dynamic Binary Translation）指令集。

RISC-V的安全性

- 支持可写不可执行。
  - 需要操作系统支持，防止直接代码注入。
- 系统态不读写用户数据
  - 需要操作系统支持，防止使用用户软件攻击系统软件。
- 支持系统和用户页表分离
  - 防止用户态程序访问系统数据空间（Meltdown）。
- 所有缓存操作指令都只是预操作而非命令
  - 防止由缓存操作指令产生缓存侧信道（flush和prefetch）。

RISC-V的组织形式（非官方解释）

- Krste Asanovic 是RISC-V商标的所有者。
  - Krste授权RISC-V基金会使用RISC-V商标。
  - 防止RISC-V基金会被大公司或者政府控制，最后的杀手锏。
- RISC-V基金会是RISC-V标准的管理机构
  - 在美国成立的非盈利组织，采取会员制。
  - RISC-V董事会成员在RISC-V的白金会中差额选举产生。
  - RISC-V董事会审核由工作小组提交的指令集扩展草案并组织投票。
  - RISC-V下设技术委员会和市场委员会。
- RISC-V技术委员会和市场委员会
  - 由RISC-V的可投票会员选举产生，负责管理RISC-V工作小组和市场营销。
- RISC-V工作小组
  - 由RISC-V会员组成，讨论并起草RISC-V指令集扩展草案。

RISC-V的会员（非官方解释）

- RISC-V独立会员（Individual）
  - 可以参加工作小组讨论。
  - 99美元每年。
- RISC-V旁观会员（Audit）
  - 可以参加工作小组和委员会，但不可以投票。
  - 2,500美元每年。
- RISC-V白银会员（Silver）
  - 可以参与投票。
  - 5,000美元每年。
- RISC-V黄金会员（Gold）
  - 可以成为工作小组主席。
  - 10,000美元每年。
- RISC-V白金会员（Platinum）
  - 可以入选为RISC-V董事会成员。
  - 25,000美元每年。

RISC-V的会员（非官方解释）

- 所有的RISC-V会员都必须签署RISC-V的会员协议（by-law）。
  - 会员协议规定了会员的年费、投票权、参与权等等。
  - 会员协议规定了会员不可以保护其对RISC-V标准的贡献。即会员不可以专利化其对RISC-V标准的贡献，从而在将来索取专利费。
  - 会员之间不可以互诉对方侵犯其关于RISC-V的专利。

RISC-V扩展指令集的产生（非官方解释）

- RISC-V技术委员会或市场委员会成立工作小组。
- 工作小组选取自己的主席（必须是黄金或白金会员）和副主席，制定标准扩展草案的轮廓。
- 当工作小组完成标准草案，草案向公众（非基金会成员）公布45天，搜集意见。
- 45天后，工作小组根据搜集的意见，选择做出改动然后递交基金会审阅，或重新编写草案重新公布。
- 如果提交基金会，工作小组必须对公示期内的所有意见提供反馈，并且对组内的反对意见给出解释。
- 如果草案被提交基金会，基金会董事会将根据草案和意见反馈决定是否展开投票。（董事会每月会举行一次）
- 如果投票通过，草案成为标准的一部分。

RISC-V的商业模式（非官方解释）

- RISC-V是一个非盈利组织。
  - RISC-V基金会管理RISC-V指令集标准的制定，本身不从标准获利。
  - RISC-V基金会保证RISC-V指令集标准是开放并免费获取的。
  - RISC-V指令集的使用是免费的（有条件，见下）。
  - 声称兼容RISC-V指令集的处理器需要通过RISC-V兼容性测试。
  - 在商业产品中使用RISC-V商标需要成为RISC-V基金会成员。
  - （例外）在非商业研究和学术项目中可以免费使用RISC-V商标。
  - RISC-V对处理器本身的商业形式没有**任何限制**。

RISC-V的主要会员

- 白金会员：
  - 谷歌：TPU
  - Microsemi：FPGA
  - nVidia: GPU controller, (**安全扩展**)
  - 西部数据: SD controller
  - 高通, Rambus, Marvell, Micron Technology, NXP, 三星, SiFive, UC Berkeley, 中兴微电子
- IC公司
  - BAE Systems， 晶心科技(**快速中断，ABI,GCC,LLVM**)， MediaTek，Lattice，西门子，华为，VeriSilicon，CEVA， Dolphin Integration，GlobalFoundries， IAR Systems， QuickLogic，希捷，C-sky
- EDA公司
  - Bluespec，UltreSoC，Imperas，Mentor
- 其他
  - IBM，特斯拉
- 中国公司
  - 华为， VeriSilicon，中科院计算所，中兴微电子，C-sky

lowRISC项目

- lowRISC项目的产生
  - 树莓派项目很成功，但树莓派并不是一个开源硬件项目。
  - OpenRISC项目逐渐没落，缺少64位指令集和高性能。
  - 开源社区和中小企业需要低成本的片上原型系统。
  - RISC-V指令集和Rocket处理器的同时出现。
  - 天使投资人Gavin Ferris在2014年找到了树莓派的创始人Robert Mullins。
  - 为中小企业和研究机构提供免费并开源的处理器平台。

- lowRISC的创始成员
  - Gavin Ferris，天使投资人。
  - Robert Mullins，剑桥大学副教授，树莓派的创始人。
  - Alex Bradbury，剑桥大学博士生，LLVM维护者之一。
  - Wei Song，曼彻斯特大学博士后，lowRISC的第一个员工。

lowRISC项目的目标和设计思路

- lowRISC项目是一个从剑桥大学成立的非盈利开源处理器组织。
  - 提供开源和免费的片上系统设计
  - 可以运行Linux操作系统
  - 高度可定制
  - 支持多核和一致性的末级缓存
  - 使用小核来实现可在线编程的外设系统
  - 使用标签内存来提高系统的安全特性、方便调试和提高性能
  - 提供方便用户使用的片上系统：
    - 使用Rocket处理器为主处理器
    - 使用PULPino处理器为小核（外设处理器）
    - 使用SystemVerilog来搭建外围片上AXI4总线。
    - 纯FPGA的原型平台。
    - 提供完整的使用文档。
    - Trace调试机制。
    - 稳定的代码版本。

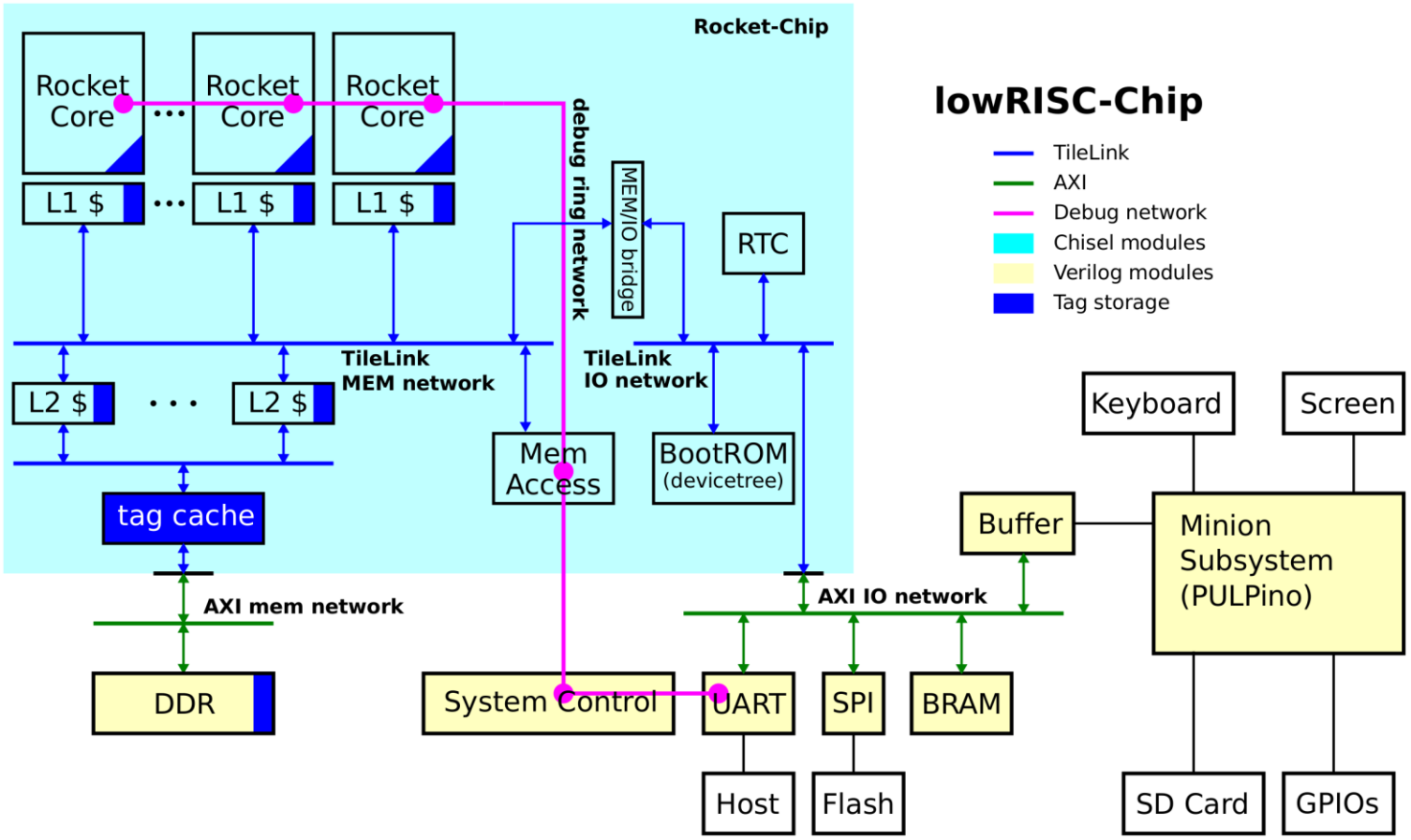
lowRISC项目现有版本

- www.lowrisc.org
- V0.1 简单的标签缓存
  - 提供最初版的标签内存实现。
- V0.2 能独立启动的lowRISC片上系统
  - 去掉了对ARM处理器的支持。
  - 实现MMIO，SD支持。
  - 从SD启动Linux。
- V0.3 带由Trace调试的lowRISC片上系统
  - Trace能够自动搜集处理器的运行状况。
  - 非中断式的硬件调试。
- V0.4 可配置的标签内存支持
  - 系统性的支持标签内存和标签内存的动态配置。
  - 使用PULPino支持简单的可编程外设。
- V0.5 支持以太网连接
  - 可以连接10/100M以太网并从网络启动。

lowRISC的用户群

- 截至2017年10月已有21个研究机构使用了lowRISC（使用者主动联系）
  - 上海交通大学
  - 国防科技大学
  - 新加坡国立大学
  - 中国科学院计算机研究所
  - 中国科学院信息工程研究所
  - 鲁汶大学
  - 维吉尼亚大学
  - 康奈尔大学
  - 佐治亚大学
  - 苏黎世理工
  - Fedora
  - Coreboot
  - FreeBSD
  - NetFPGA
- 每月的源码下载量~250次，网站访问量上千次。

lowRISC V0.4 的内部结构



lowRISC V0.4版的特性

- Rocket处理器部分
  - UCB 2016年夏天
  - 使用TileLink连接的L2
  - Privileged spec ~ 1.9.1
  - GCC and Linux Kernel (2016年底)

- lowRISC的附加特性
  - SystemVerilog 顶层连接，AXI4外设总线
  - Trace调试器
  - 内置的标签内存支持

- 使用PULPino实现的SD卡接口
- 使用Nexys4-DDR的完整FPGA原型实现

lowRISC V0.5版的特性

- 100Mbps 以太网支持。
- 支持从运行Linux的主机远程启动实验板。
- 支持中断的Linux驱动。
- 优化的SD接口。
- NFS网络文件系统支持。
- 多用户Linux系统支持(poky)。

一些较有名的RISC-V处理器实现

- Rocket
  - UC Berkeley, SiFive, lowRISC
  - 6级有序流水线，Chisel实现，SiFive的主打产品，对标ARM Cortex-A53
  - 多次流片
- BOOM
  - UC Berkeley
  - 乱序流水线，Chisel实现，对标ARM Cortex-A57
  - 多次流片
- PULP
  - ETH Zurich
  - 从1级到5级流水线，SystemVerilog实现，AXI总线，低功耗，微控制器。
  - 多次流片验证。

一些较有名的RISC-V处理器实现

- GRVI
  - Jan Gray针对FPGA高度优化的微处理器
  - Verilog设计，250~320 LUT
  - 在一个FPGA上实现1240个处理器的互联。
  - 商用授权，非开源。

- PicoRV32
  - Clifford Wolf设计的可流片的微处理器小核
  - Verilog设计，700~2000 LUT
  - >1GHz 主频，但是CPI在4~6之间。

- SHAKTI
  - 印度的国家处理器计划，选择RISC-V为国家指令集
  - 从微处理到服务器级别的处理器设计
  - Bluespec设计

CNRV

- CNRV: RISC-V @ China
  - CNRV通过微信群、网站和公众号，与大部分国内RISC-V的使用者、参与者和爱好者建立了联系，包括软件/IC/硬件开发者、爱好者、相关公司高管、高校研究人员以及投资人等。
  - 每两周通过网站和公众号发布《CNRV双周简报》，向中文读者介绍和总结RISC-V相关领域的最新进展，可理解为RISC-V大参考～
  - CNRV网站搜集关于RISC-V的各种信息，为国内的RISC-V使用人员提供帮助。

- CNRV的主要成员
  - **郭雄飞**：景略半导体（上海）设计总监，RISC-V独立会员，CNRV微信群的管理员，RISC-V双周简报编辑，国内最早接触和推广RISC-V的人。
  - **黄柏玮**：国立台湾大学咨询网路与多媒体研究所硕士，RISC-V独立会员，RISC-V调试小组和原比特操作小组成员，脸书RISC-V@Taiwan小组管理员。
  - **宋威**: 中科院信工所副研究员，原lowRISC工程硬件负责人， RISC-V双周简报编辑。

CNRV的网站

- https://cnrv.io/
- RISC-V双周简报
- 资源列表
- RISC-V镜像服务

- 访问量：
  每月访问次数~1100次

CNRV
为推广RISC-V尽些薄力

View On GitHub

关于

本站点希望能够为国内的RISC-V开发者和爱好者提供便利。

RISC-V双周简报

国内的RISC-V爱好者利用Github协作的方式，以双周简报的方式为大家带来最新的RISC-V相关咨询，同时在微信公众号和CNRV站点上发布。内容覆盖RISC-V邮件列表、行业新闻、项目进展以及各类点评。也欢迎大家关注CNRV公众号获取最新信息。

- **第0x15弹(2018-04-13): 第八届Workshop会议议程公布**
- 第0x14弹(2018-03-30): 宗师获图灵奖实质名归 [繁体]
- 第0x13弹(2018-03-16): 想把事情做简单可不是那么简单的事情 [繁体]
- 第0x12弹(2018-03-02): 看看AI和RISC-V碰撞出的火花 [繁体]
- 第0x11弹(2018-02-15): 跑Linux的CPU一次来俩 [繁体]
- 第0x10弹(2018-02-01): 走在"时尚的前沿" [繁体]
- 第0x0f弹(2018-01-18): Google开源RISC-V CPU [繁体]
- 第0x0e弹(2018-01-04): Intel漏洞是非多！ [繁体]
- 第0x0d弹(2017-12-21): Esperanto Technologies会成功么？ [繁体]
- 第0x0c弹(2017-12-07): Workshop上干活多多 [繁体]
- 第0x0b弹(2017-11-23): 3家RISC-V相关的公司上榜"EE Times Silicon 60: Startups to Watch" [繁体]
- 往期存档

文章列表

- 2017-10-12: RISC-V 双周简报问卷调查结果分析

资源列表

- **RISC-V 资料搜集页面**
- **RISC-V 参考文献页面**

如何在国内快速搭建SiFive的Freedom环境

下载freedom/rocket-chip/riscv-tools完整压缩包

因为国内用户访问github比较慢，而且clone过后submodule更加慢，以下的压缩包是在clone了freedom之后，执行`git submoudle update --init --recursive`之后打包的。所以已经以submodule的形式包含了rocket-chip和riscv-tools以及下面的诸多submodules。

网盘中还新增了freedom-e-sdk和freedom-u-sdk的完整压缩包。

- 百度网盘freedom/rocket-chip/riscv-tools完整打包文件
  - GitHash: ec70d85
  - MD5Sum: b3643841ff41083f004871359dd3ffe4
  - 打包时间: 2017-Aug-19
- 百度网盘freedom-e-sdk完整打包文件
  - GitHash: fcbcd44
  - MD5Sum: 1b5c97dd71918cfaa1c43fb7f38ecade

CNRV的资源列表

- https://cnrv.io/resource

- 已经搜集了26个RISC-V处理器设计。其中开源设计23个。

- 操作系统
- 开发工具
- 验证环境
- 文档

CNRV
为推广RISC-V尽些薄力

View On GitHub

RISC-V资源列表

处理器实现

- BOOM: Christopher Celio的RV64乱序处理器实现。Chisel, BSD Licensed。[GitHub] [Doc]
- BottleRocket: RV32IMC微处理器。Chisel，Apache Licensed。[GitHub]
- bwitherspoon: RV32微处理器。SystemVerilog, ISC Licensed。[GitHub]
- Clarvi: 剑桥大学教学用RISC-V处理器。SystemVerilog, BSD Licensed。[GitHub]
- F32: 针对FPGA的RV32微处理器，VHDL，BSD Licensed。[GitHub]
- GRVI: Gray Research LLC. 针对FPGA优化的RV32微处理器，commercial licensed。[Web]
- Hummingbird E200. 二级流水线，目标替代Cortex-M0/8051, Verilog, Apache 2.0 licensed。[GitHub]
- invicta: 一级流水线的RV32微处理器。Verilog，BSD Licensed。[GitHub]
- Kamikaze: RV32微处理器。Verilog，MIT Liencensed。[GitHub]
- KCP53000: Samuel A. Falvo II的RV64处理器实现。Verilog, MPL Licensed。[GitHub]
- nanorv32: 2机流水线的RV32实现。Verilog, GPLv2 Licensed。[GitHub]
- OpenV: 支持RV32的开源微处理器，Verilog，MIT Licensed，OnChipUIS，来源于哥伦比亚的Universidad Industrial de Santander。[GitHub]
- ORCA: 支持RV32的开源微处理器，VHDL，BSD Licensed，VectorBlox。[Github]
- PicoRV32: Clifford Wolf设计的(针对FPGA)RV32微处理器，Verilog，ISC Licensed。[GitHub]
- Potato: 针对FPGA的RV32微处理器。VHDL，BSD Licensed。[GitHub]
- RI5CY：支持RV32的开源微处理器
  - PULPino: SystemVerilog，Solderpad Licensed, 来源于苏黎世理工和博洛尼亚大学的PULP项目。[GitHub] [Web]
- RIDERCORE: RISC-V乱序处理器设计。Verilog, BSD Licensed。[GitHub]
- River: GNSS Senor Ltd.基于Rocket架构开发的RV64处理器。VHDL, BSD Licensed。[GitHub]
- Rocket: 支持RV64/32的开源处理器
  - Rocket-Chip: Chisel，BSD Licensed, Free chips project, UC Berkeley分离的开源工程。[GitHub]
  - Freedom: Chisel，Apache Licensed, SiFive, UC Berkeley分离的初创企业。[GitHub] [Web]
  - lowRISC：Chisel+SystemVerilog，Solderpad Licensed, 从剑桥大学发起的非盈利组织。[GitHub] [Web]
  - RoCC: the Rocket customized coprocessor interface 和Rocket处理器紧密互联的的协处理器接口。[BSG]
- RV12: RoaLogic的RV32微处理器。Verilog, RoaLogic non-commercial Licensed。[GitHub]
- SCR1: Syntacore的RV32开源微处理器。SystemVerilog，Solerpad Licensed。[GitHub]

CNRV的微信群和双周简报

- 受500人人数上限限制，现在已经有3个子群，约1000人。
- 每天的活跃用户~100人，消息约500~2000条每天。

- 微信公众号的关注到~1100人，日增长~10人。

- 双周简报的阅读数量在~600人，已经有了21期。

RISC-V在中国的发展情况

- 大家都很关心，讨论热烈。
- 已有自己的RISC-V处理器（非RISC-V会员）
  - 蜂鸟200系列（胡振波）https://github.com/SI-RISCV/e200_opensource
  - 对标ARM Cortex-M0
- 大公司多在观望。
- 小公司已经在行动，闷声开搞。

中国科学院信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING,CAS

Thank You!