MathWorks
Accelerating the pace of engineering and science

4月 - 北京 · 上海 · 深圳

2015 MATLAB 巡回研讨会

技术融合的时代

基于模型的设计在DSP中的应用

周德斌
系统工程师

广播科学研究院

© 2015 The MathWorks, Inc.

单位介绍

广播科学研究院 —— 国家新闻出版广播电影电视总局下直属的一个研究单位 。它有电视所，无线所，有线所，信息所，互联网所，技术经济所 ，检测中心和高科技中心等单位和部门组成。目前有在职员工近200人。主要进行广播、电视技术，网络电视，信息安全，版权保护等技术的研究和探索。

项目背景

- 需求： 在传统模拟音频中高速、高可靠的传输数字信息
- 考虑采用的技术：
  - 采用现代调制技术FSK，MSK，CPM 等，相对于传统技术较为复杂
  - FEC技术，卷积码、LDPC码等
  - 交织、去交织
- 接收端一般包含：
  - 信号调理，高阶数字滤波器（低通、带通，几十到上百阶 ），
  - 位同步，帧同步
  - 解调、去交织、解码等

具体项目系统构架图


上位机：
控制字
RS232
FEC编码、
调制
音频输出
发射机
接收电路
音频输出
滤波、解调、
FEC译码
RS232
下位机：
控制字

项目挑战

- 如何快速进行项目研究？

选择MATLAB做为算法研究及仿真平台

- MATLAB有非常多的信号处理和通信的算法
- MATLAB的可视化很方便
- 音频处理、播放很方便
- 有更多时间进行算法的研究，而不是编程序

算法仿真的三个阶段

- 普通MATLAB代码做流处理方式
- 借助系统对象做流处理方式
- Simulink方式

普通MATLAB代码做流处理

```
%% Streaming the MATLAB way
% set up initializations
filename = 'dspafxf_8000.wav';
Fs = 8000;
info = mmfileinfo(filename);
num_samples = info.Duration*Fs;
frame_size = 40;
bLP = fir1(40, 0.8,'high');
zLP = zeros(1,numel(bLP)-1);
output = zeros(1,num_samples);

%% Processing in the loop
index= 1;
while index < (num_samples-frame_size+1)
    data = audioread(filename,[index index+frame_size-1]);
    [datafilt, zLP] = filter(bLP,1,data,zLP);
    output(index:index+frame_size-1) = datafilt;
    index = index + frame_size;
end
sound(output,Fs);
```

显式的状态管理
需要程序员控制一些隐含的细节

索引很烦
而且常常引入错误

需要人为的维护输出缓存
调用**sound**前需要准备好所有的数据

新挑战

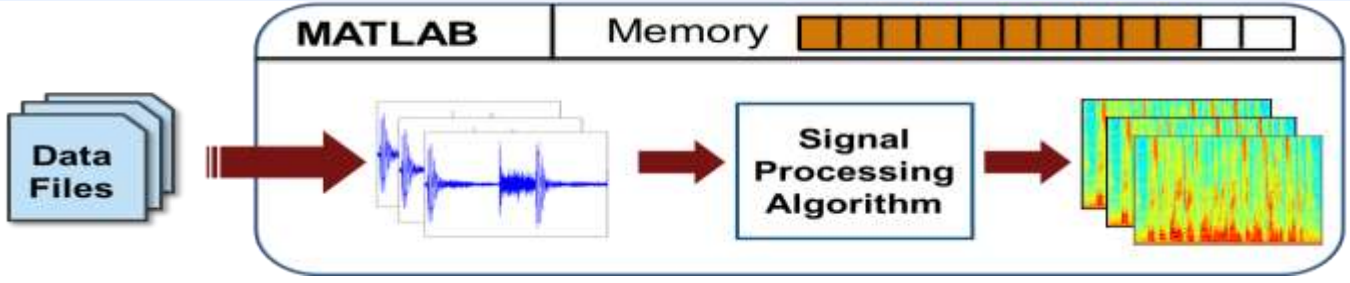
- 音频文件很大，几个G的文件
  - Out of Memory
- 解决办法：
  - 32 OS 换成 64 OS，增加物理内存
  - 采用System Object

算法仿真的三个阶段

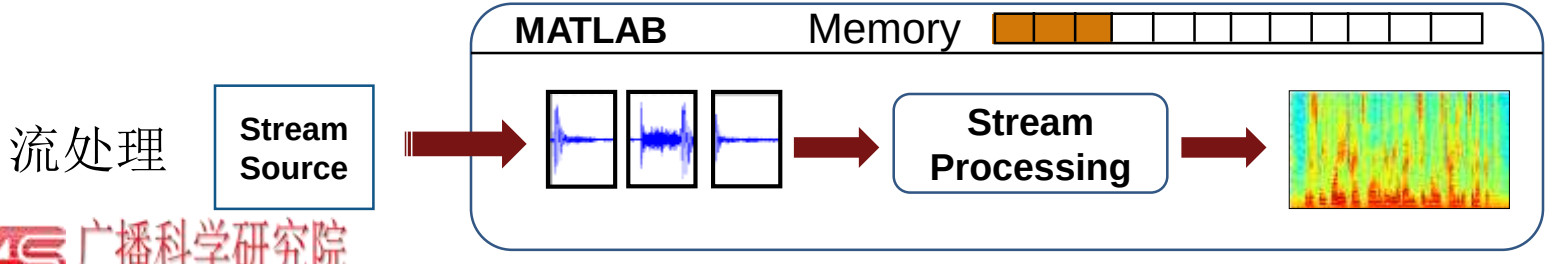
- 普通MATLAB代码做流处理方式
- 借助系统对象做流处理方式
- Simulink方式

借助系统对象做流处理原理

- 流技术将连续的数据分块为许多帧来处理
  - ✓ Memory efficient
  - ✓ 隐含的数据缓存，内部状态管理，索引
  - ✓ 减少开销提高仿真速度



批处理



流处理

广播科学研究院

借助系统对象做流处理程序

```
% set up initializations
filename = 'dspafxf_8000.wav';
hFilter = dsp.DigitalFilter;
hFilter.TransferFunction = 'FIR (all zeros)';
hFilter.Numerator = fir1(40, 0.8,'high');
hAudioSource = dsp.AudioFileReader(filename,  ...
        'SamplesPerFrame',40,'OutputDataType','double');
hAudioOut = dsp.AudioPlayer('SampleRate', 8000);

%% Processing in the loop
while ~isDone(hAudioSource)
    data     = step(hAudioSource);
    datafilt = step(hFilter, data);
    step(hAudioOut, datafilt);
end
```

初始化对象

有多种方式设置对象参数

数据源和FIR滤波器的状态是隐含的

循环体里面的代码变的简洁

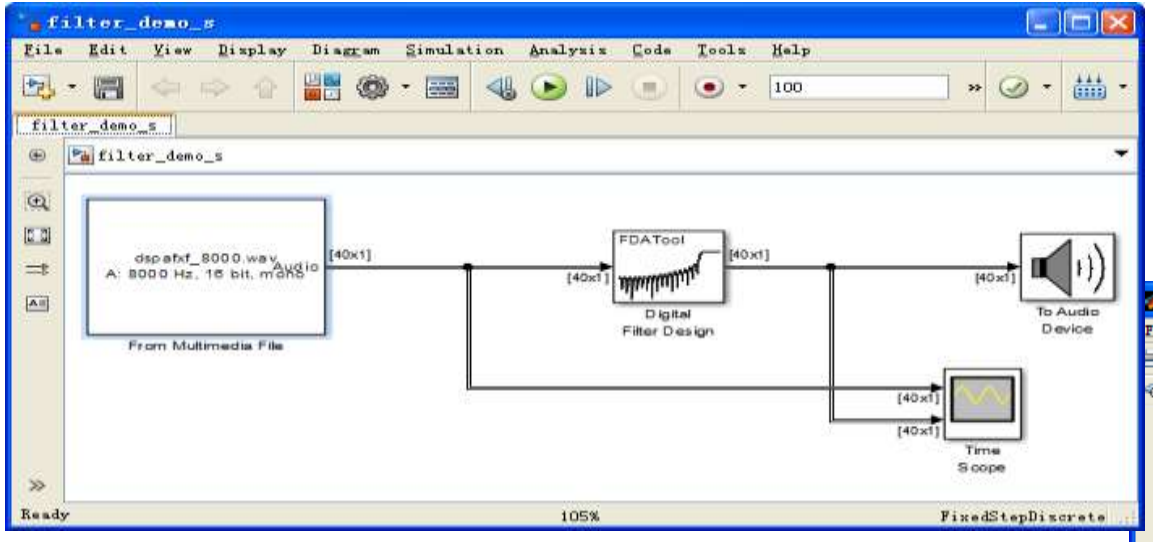
不用再管理索引

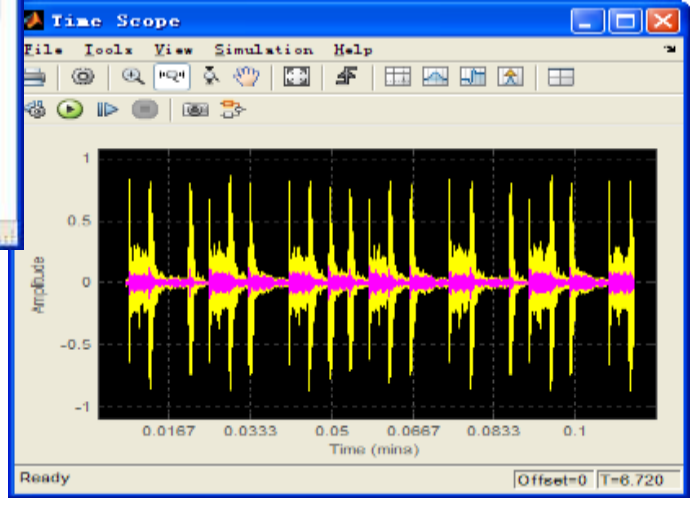
在循环体中播放当前一帧，避免开较大的缓存

算法仿真的三个阶段

- 普通MATLAB代码做流处理方式
- 借助系统对象做流处理方式
- Simulink方式

Simulink 流处理





算法仿真结果：

加入高斯信噪源仿真，在信噪比为0db甚至更低的情况下，帧误码率达到万分之几的量级，满足项目要求。

项目挑战

- 如何快速进行项目研究？
- 研究成果能够产品化吗？

产品化的曲折之路

- 考察：
  - 嵌入式应用
  - 实现比较容易（编程）
  - 处理能力足够强（实时信号处理、通信算法）
  - 有更多的资源可以借鉴
  - 功耗比较低
  - 成本控制
- 选取实现的载体：
  - ARM
  - DSP
  - FPGA
  - 专有芯片

广播科学研究院

定点还是浮点？

考　察　项	Floating Point	Fixed Point	Fixed Point with MathWorks Tools
RAM 和 ROM 消耗	↑	↓	↓
执行速度	↓	↑	↑
功　耗	↑	↓	↓
嵌入式硬件成本	↑	↓	↓
开发时间	↓	↑	↓
实现复杂度	↓	↑	↓
错误引入	↓	↑	↓

定点目标

- 定点DSPs (TI, Analog Devices 等)
  - 固定字长
  - 价格更低
  - 功耗更低 – 利于电池供电应用
  - 更高时钟速度

DSP	MHz	MMACS	US$
5510A	200	400	15
674x	456	3648	15
6671	1250	40000	100
6416	1000	8000	100
6713	300	2400MIPS	30

广播科学研究院

常规方式产品化的团队构成和实现步骤

- 系统工程师
  - 完成系统算法（已完成）
- 硬件工程师
  - 评估硬件需求
  - 设计DSP板卡
  - 调试DSP板卡
- 算法实现工程师
  - 实现DSP算法
  - 调试DSP驱动
- 测试验证工程师

广播科学研究院

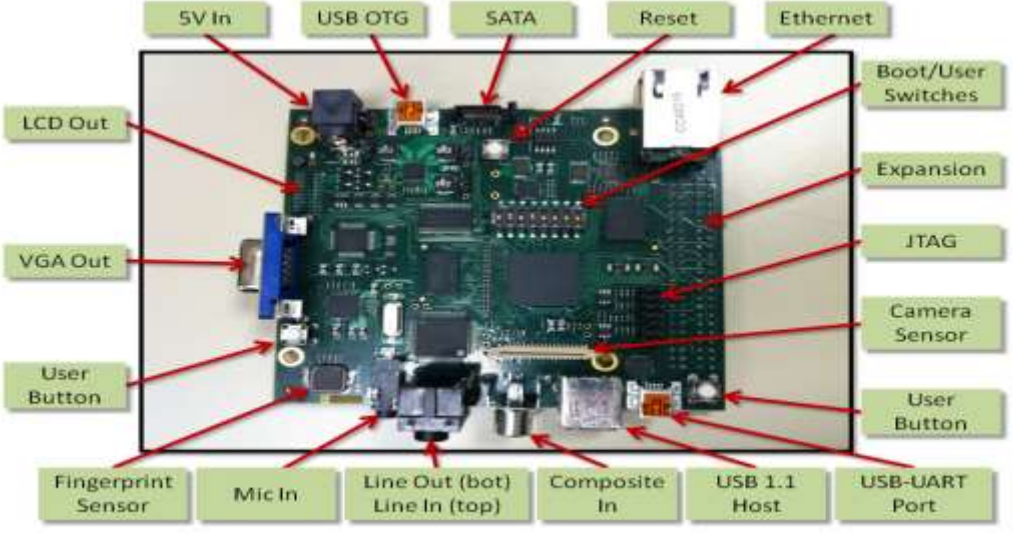
自动生成代码：Embedded coder

- 产生产品级代码
  - 专用处理器，优化代码
  - Simulink 模块和优化库(FIR, FFT, …)
- 工程自动化
  - 自动产生整个工程
  - CCS、编译器、连接器的API函数

- 快速原型
  - 特定目标和集成环境
  - Simulink硬件模块，支持设备驱动（ADC, DAC, RTDX, 子卡)
- 支持
  - 开发板: TI 6713 DSK, C6416DSK, DM642 EVM,C6455EVM, DM6437EVM, DM648EVM, C6747EVM

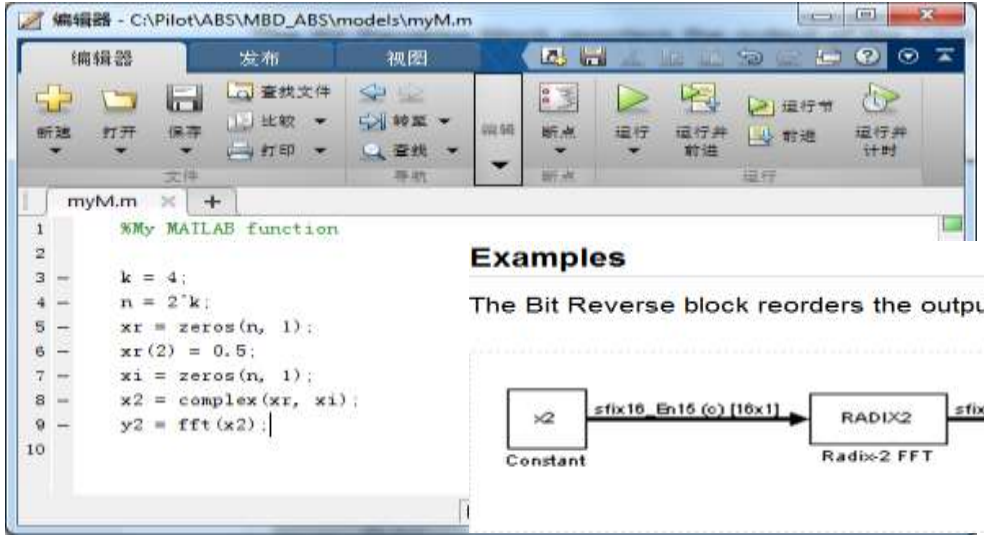
快速原型的意义

- 可靠的硬件平台验证算法
- 算法性能评估
- 评估硬件需求
- 指导硬件设计



MATLAB到Simulink

- Simulink更适合实时流处理
- 层次结构更加直观



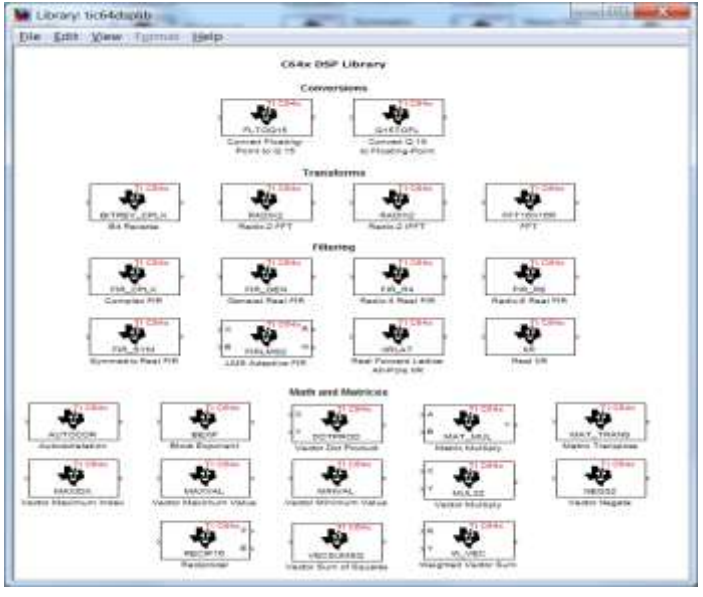
MATLAB与Simulink定点的对比：

[y2,	c]
0.5000	0.5000
0.4619 - 0.1913i	0.4619 - 0.1913i
0.3536 - 0.3536i	0.3535 - 0.3535i
0.1913 - 0.4619i	0.1913 - 0.4619i
0 - 0.5000i	0 - 0.5000i
-0.1913 - 0.4619i	-0.1913 - 0.4619i
-0.3536 - 0.3536i	-0.3535 - 0.3535i
-0.4619 - 0.1913i	-0.4619 - 0.1913i
-0.5000	-0.5000
-0.4619 + 0.1913i	-0.4619 + 0.1913i
-0.3536 + 0.3536i	-0.3535 + 0.3535i
-0.1913 + 0.4619i	-0.1913 + 0.4619i
0 + 0.5000i	0 + 0.5000i
0.1913 + 0.4619i	0.1913 + 0.4619i
0.3536 + 0.3536i	0.3535 + 0.3535i
0.4619 + 0.1913i	0.4619 + 0.1913i

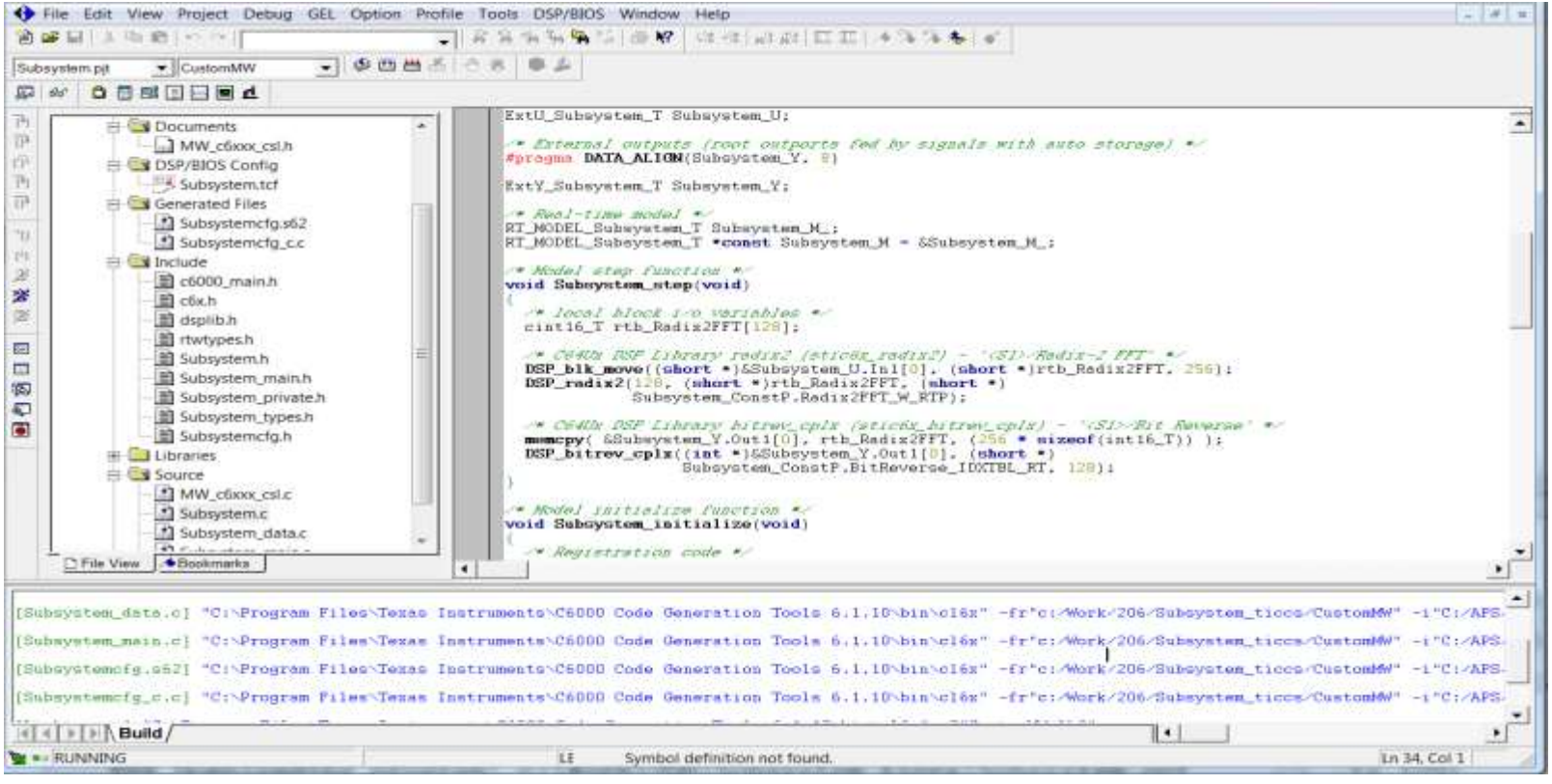
广播科学研究院

产生的代码效率够好吗？

- 在评估板上实时运行，评估算法效率
- 调整算法结构（经验、尝试）
  - 音频输入多少点为一帧？
  - 64点？128点？256点？512点？
- 优化关键算法
  - FIR，FFT，



自动产生的CCS工程文件和DSP代码



两点补充体会

- MATLAB流水线处理是靠缓存重叠技术来实现帧与帧之间的无缝连接的,对于用户是透明的，这是基于系统对象和Simulink方式实现的关键点

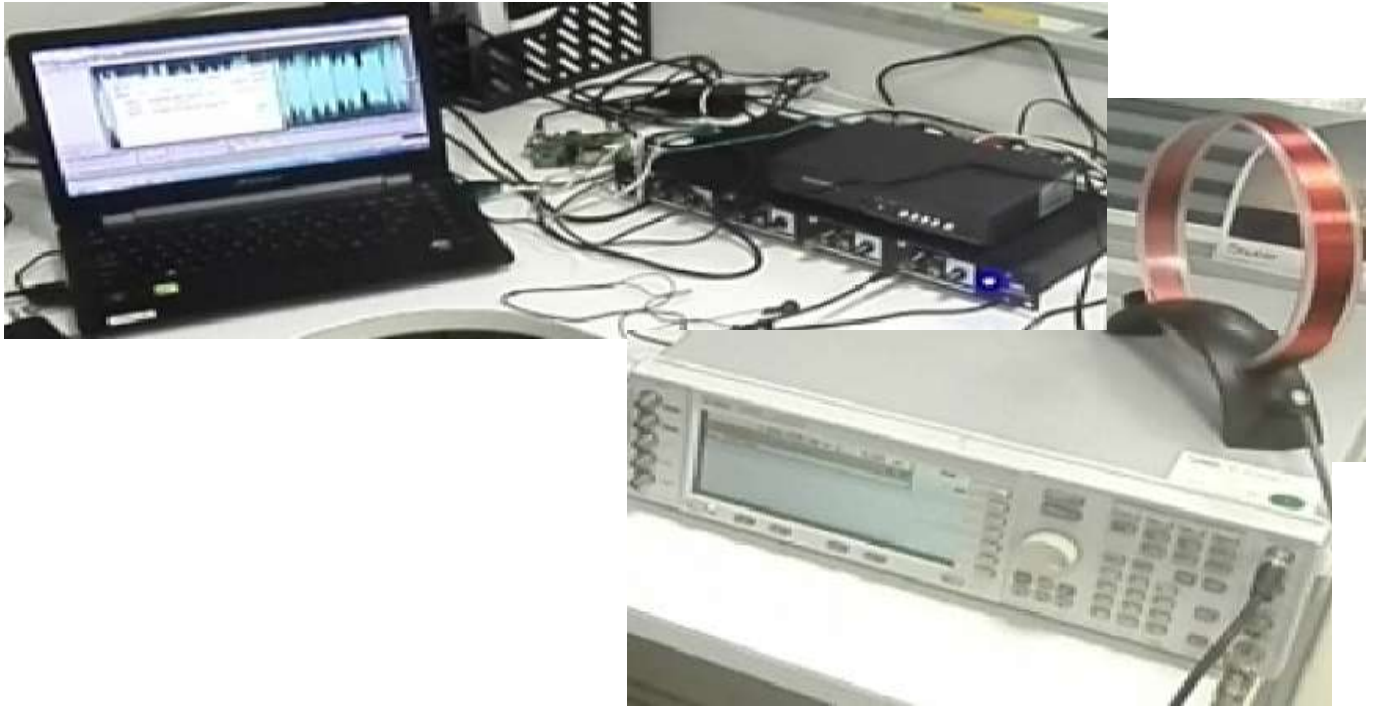
- 能用MATLAB函数的方式编写一段M程序算法嵌入到Simulink里一起执行，这是标准库里没有用户需要的算法时的关键点

工作成果

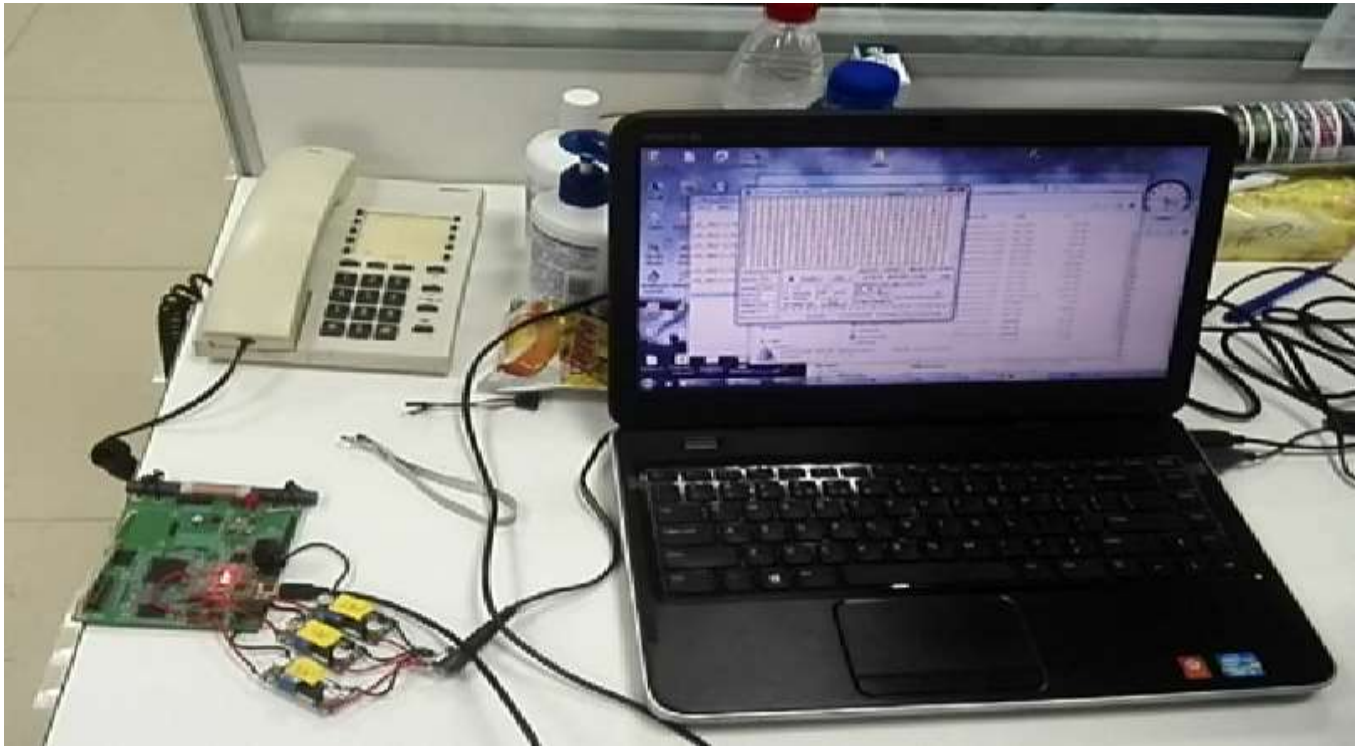
- 实验环境
  - 在实验室环境下通过矢量信号发生器发射实验（几米距离）
  - 在北京广播设备器材厂实际中波发射机上发射实验（几十米距离）

- 结论
  - 在中波广播中插入数据通信能高可靠的接收
  - 达到与理论仿真同等效果

实验室环境下的发射端：

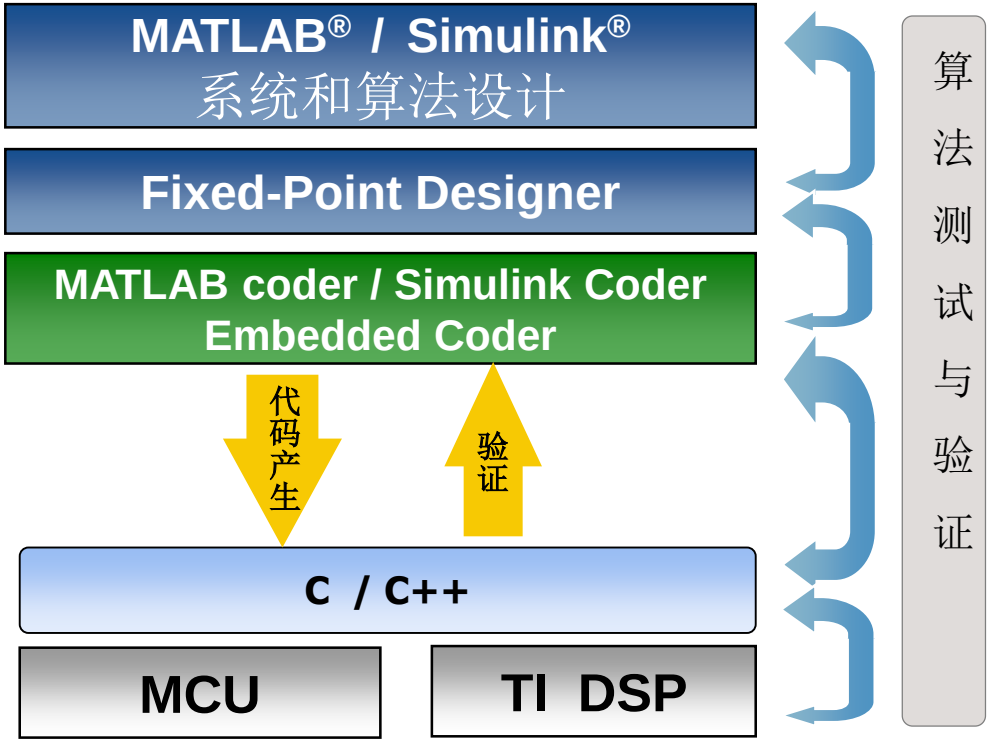


接收端：



实际中波发射机

基于模型的设计总结与回顾



谢谢！

- 领导和同事
- 销售部
- 技术工程师
- 市场部