Methodology

- Past design effort focused on lower levels
- Higher levels lack well-defined methodology and tools
- Paradigm shift to higher levels can increase productivity
- Need methodology and tools for system level

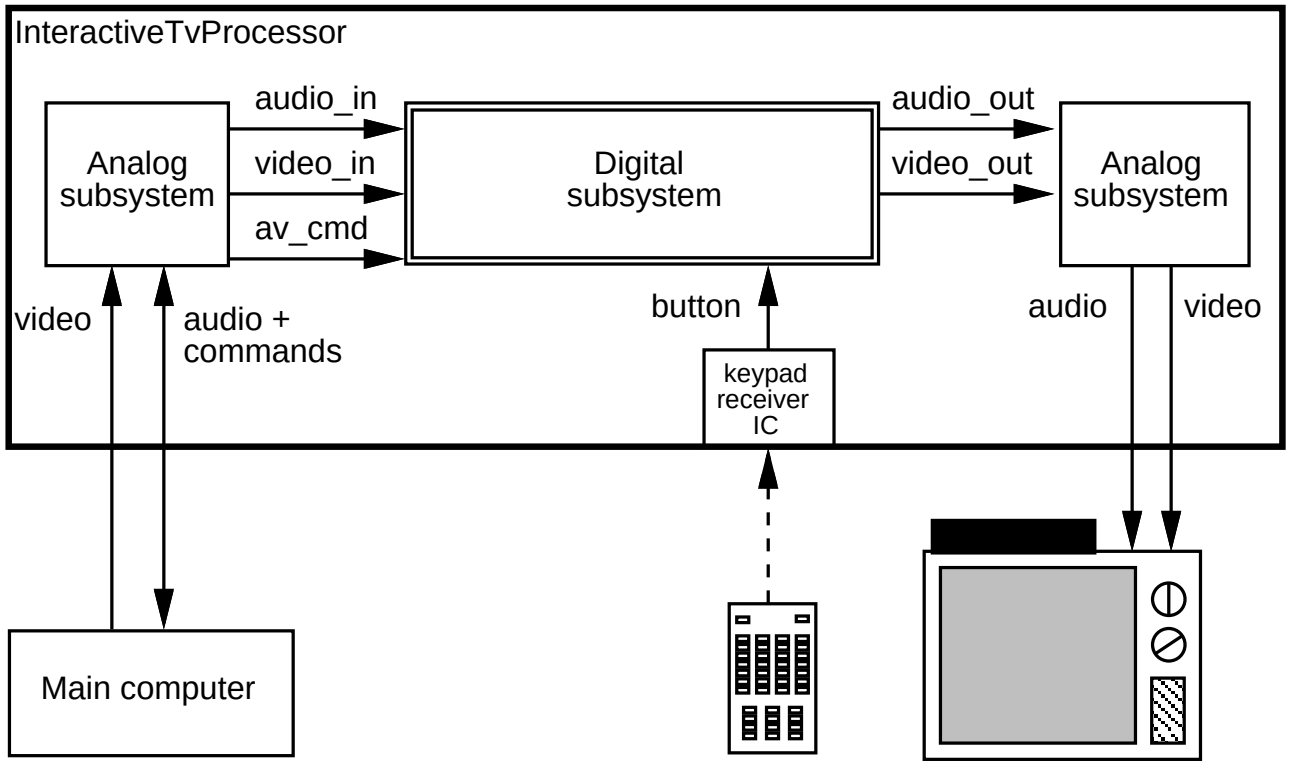
Outline

- Basic concepts in design methodology
- Example
- A design methodology
- A generic synthesis system
- Conceptualization environment

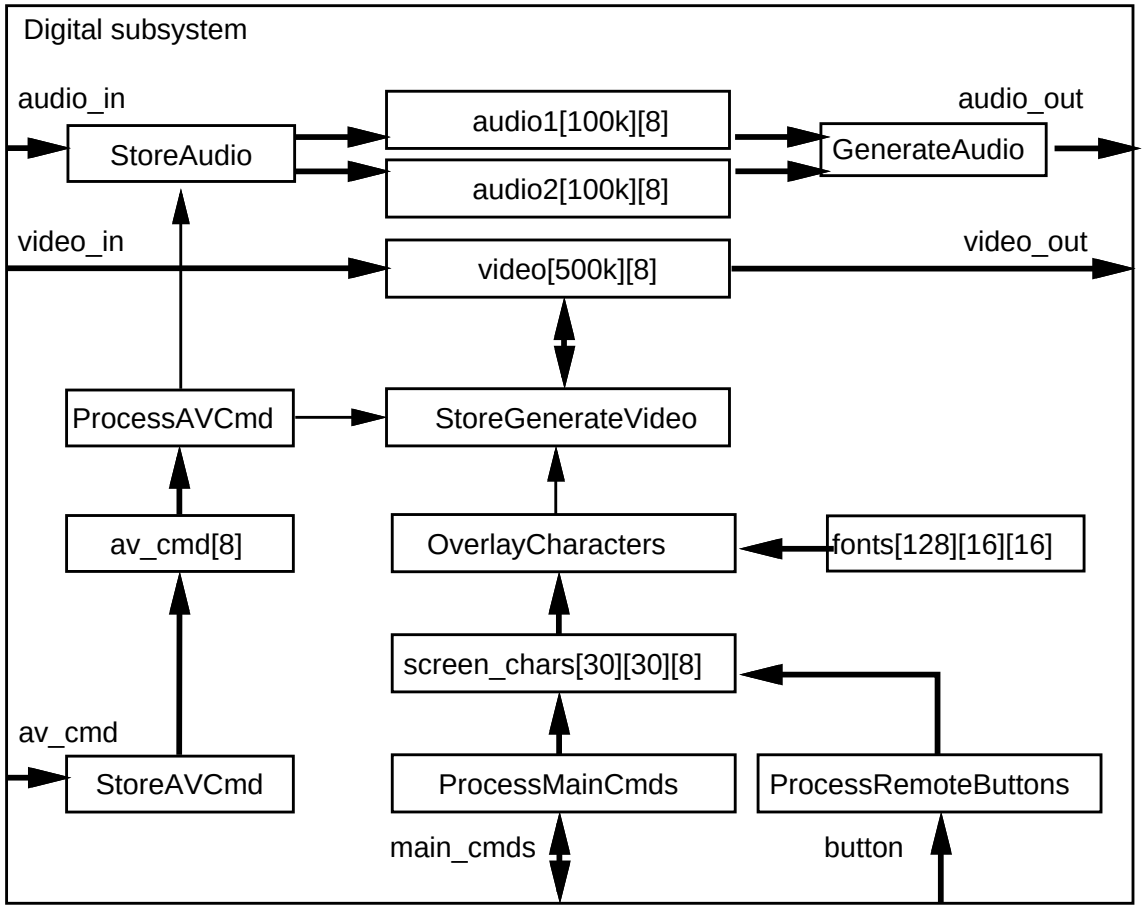
Items a design methodology must specify

- Syntax and semantics of input and output
- Algorithms for transforming input to output
- Components to be used in the design implementation
- Definition and ranges of constraints
- Mechanism for selection of architectural styles
- Control strategies (scenarios or scripts)

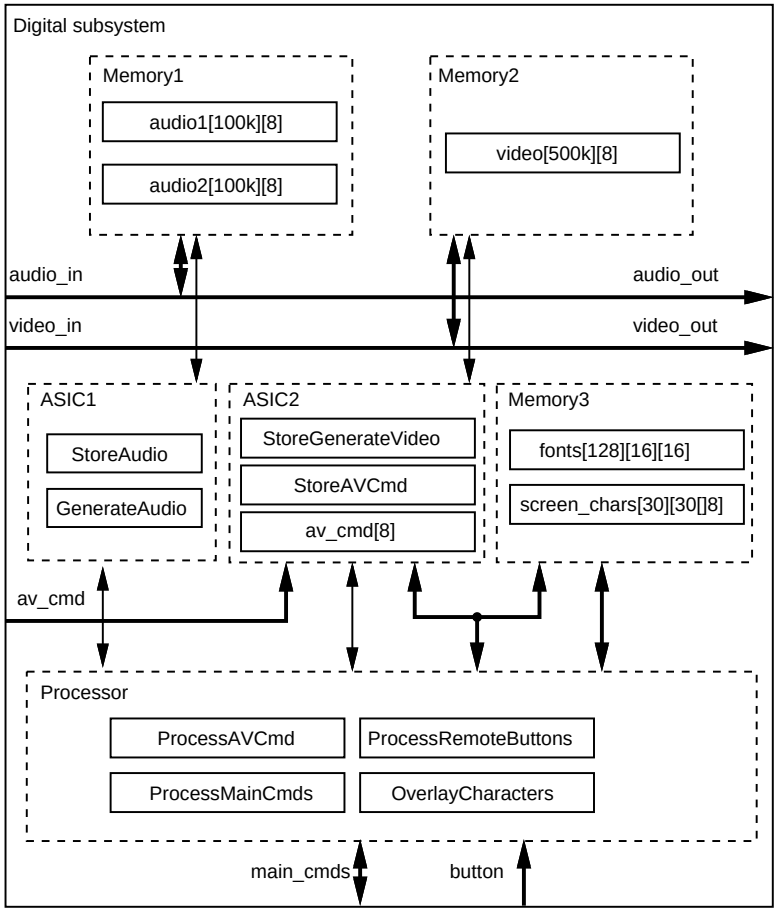
Example: Interactive TV processor



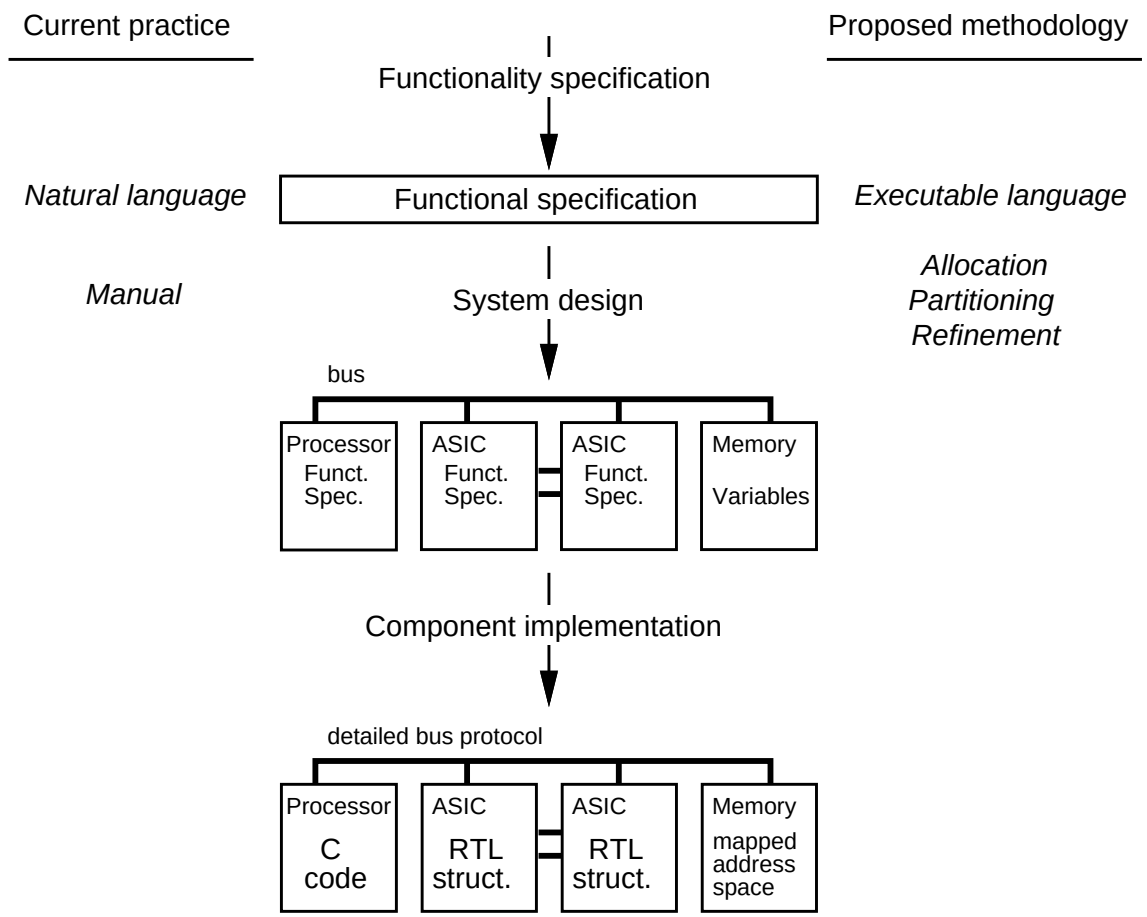
Example's data flow behavior



Example's implementation after system design



An example design methodology

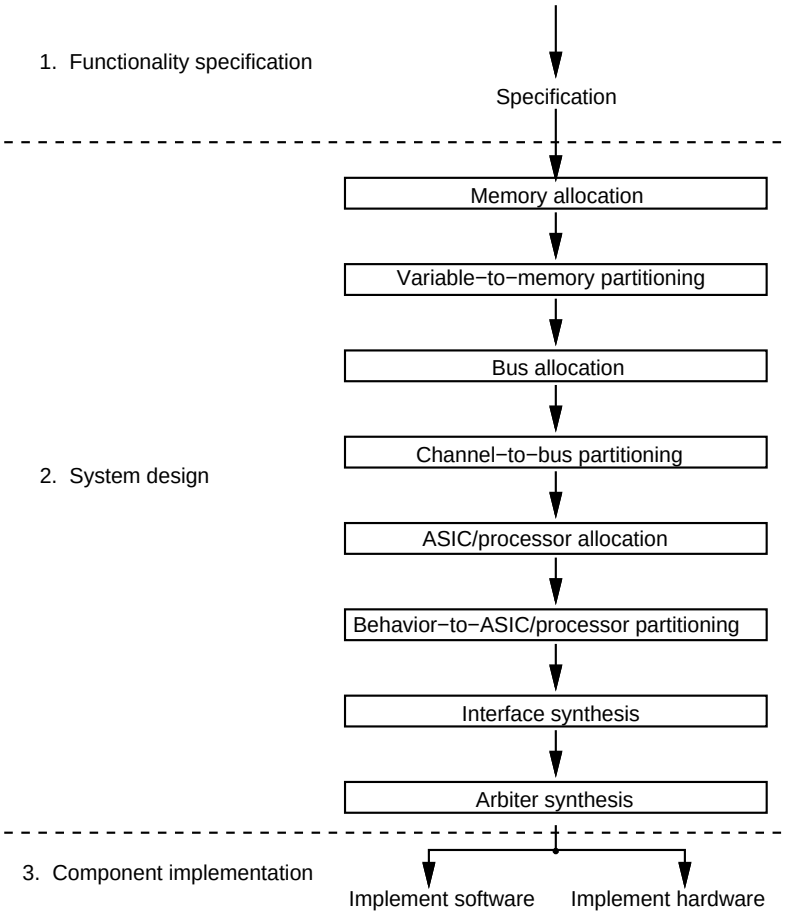


System-design tasks

System−design tasks

Functional objects	Allocation	Partitioning	Refinement
Variables	Memories	Variables to memories	Address assignment
Behaviors	Processors	Behaviors to processors	Interfacing
Channels	Buses	Channels to buses	Arbitration/protocols

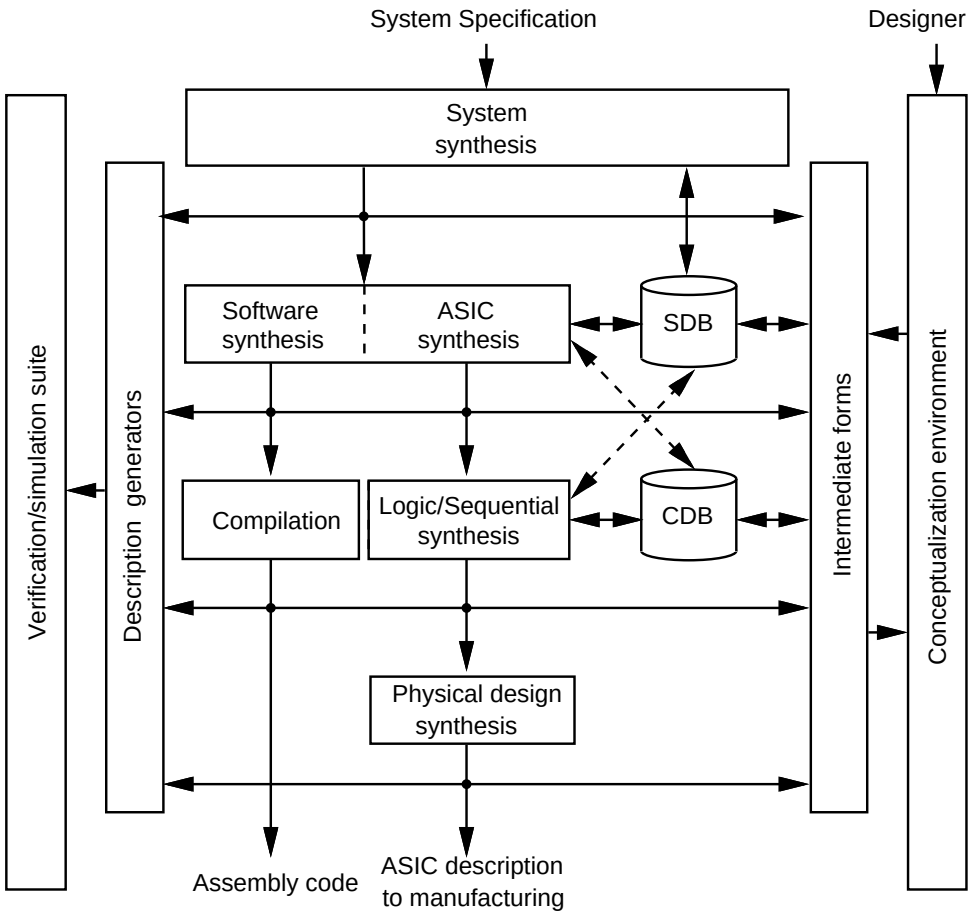
One possible ordering of tasks



Generic synthesis system requirements

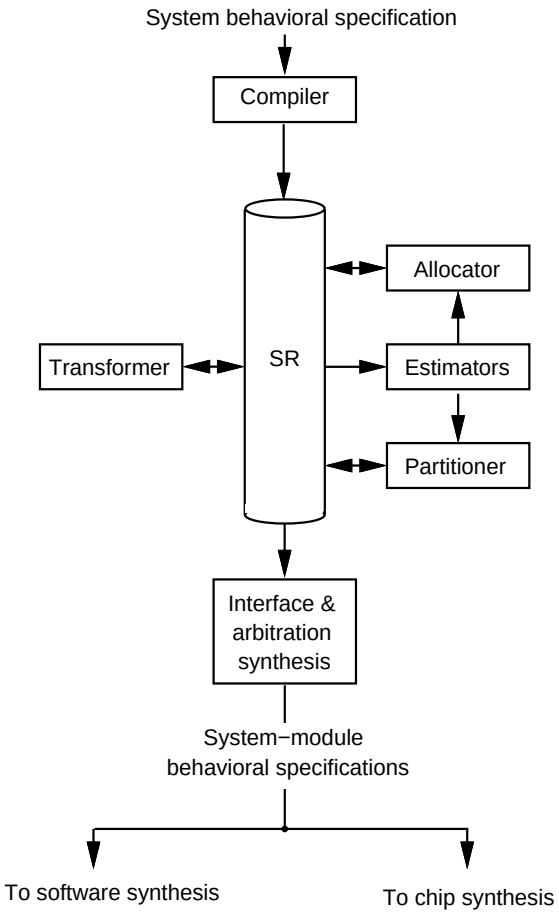
- Completeness
  - All levels of design, all implementation styles
- Extensibility
  - Allow addition of new algorithms and tools
- Controllability
  - User control of tools, design-quality feedback
- Interactivity
  - Partial design, design modification
- Upgradability
  - Evolve to describe-and-synthesize method

A generic synthesis system

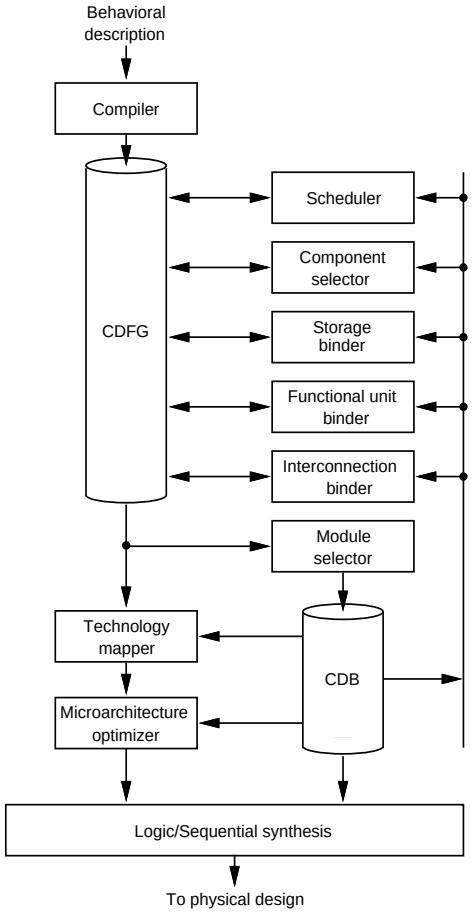


UC Irvine

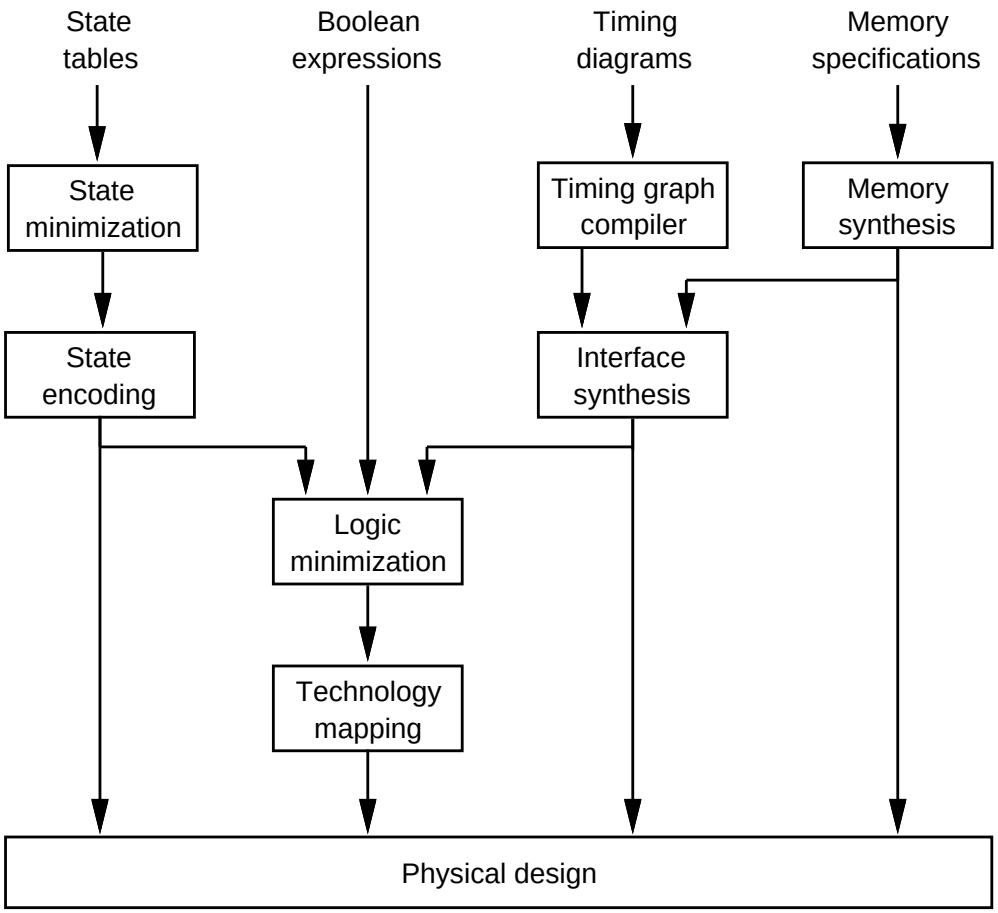
A generic system-synthesis tool



A generic chip-synthesis tool



A generic logic-synthesis tool



Conceptualization environment

- Tool is only effective if the designer can use it
  - Understandable display of data
  - Highlight design parts that need attention
- Must support many design avenues

A system-synthesis tool interface

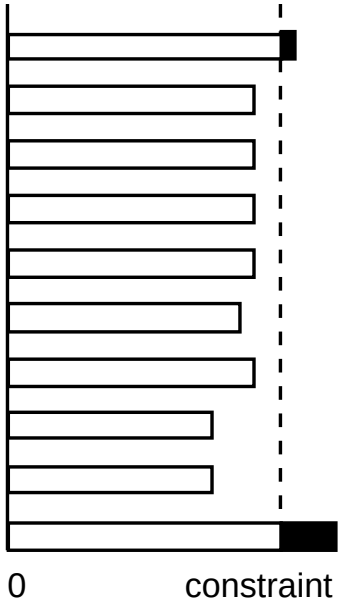
- Allocation
- Partition
- Estimates
- Constraints

Mappings	Module type	$	Execution time	Area	Pins	Instr
System		105 /100*				
ASIC1	X100	30		16000 /20000	46/60	
CaptureAudio			100/110			
GenerateAudio			100/110			
ASIC2	X100	30		18000 /20000	48/60	
CaptureGenerateVideo			100/110			
CaptureAVCmd			100/110			
Memory1	V1000	10				
audio_array1						
audio_array2						
Memory2	V1000	10				
video_array						
Processor1	Y900	25				6000 /5000*
ProcessRemoteButtons						
ProcessMiscCmds						

Cost: 5.43 | View options | Partition/Allocate | Refine

An optional design view

Quality metric	Estimate/ Constraint	Violation?
$(System)	105/100	
Execution−time(CaptureAudio)	100/110	
Execution−time(GenerateAudio)	100/110	
Execution−time(CaptureGenerateVideo)	100/110	
Execution−time(CaptureAVCmd)	100/110	
Area(ASIC1)	16000/20000	
Area(ASIC2)	18000/20000	
Pins(ASIC1)	56/60	
Pins(ASIC2)	58/60	
Instr(Processor1)	6000/5000	



Summary

- Three-step design methodology
  - Functionality specification
  - System design
  - Component implementation
- Major tasks in system design
  - Allocation
  - Partitioning
  - Refinement
- Generic synthesis tool
- Conceptualization environment
  - Crucial to practical use

Future directions

- Advanced estimation methods
- Formal verification
- Testability
- Frameworks and databases
- Regularity exploiting
- System-level transformations
- Feedback incorporation