

Reduced Switching Delay for Networked Music Performance

George Xylomenos, Christos Tsilopoulos, Yannis Thomas and George C. Polyzos

Mobile Multimedia Laboratory, Department of Informatics

Athens University of Economics and Business, Greece

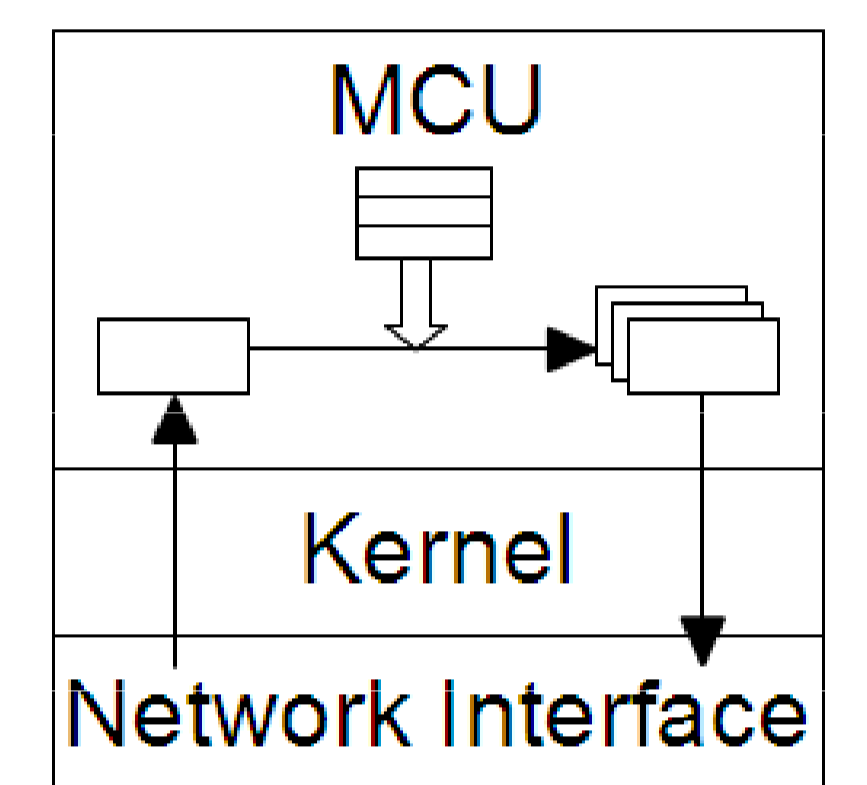
{xgeorge,tsilochr,thomasi,polyzos}@aueb.gr

Motivation

- NMP: Networked Music Performance
 - Ultra-low delay variant of conferencing
 - End-to-end delays of 25 rather than 150 ms
- The MusiNet project
 - Ultra-low delay audio and video coding
 - Optimized media capture and packetization
 - What else can we optimize?
- Multipoint Conferencing Unit (MCU)
 - Receives data streams from each participant
 - Mixes all data streams together
 - Relays the resulting data stream to each participant

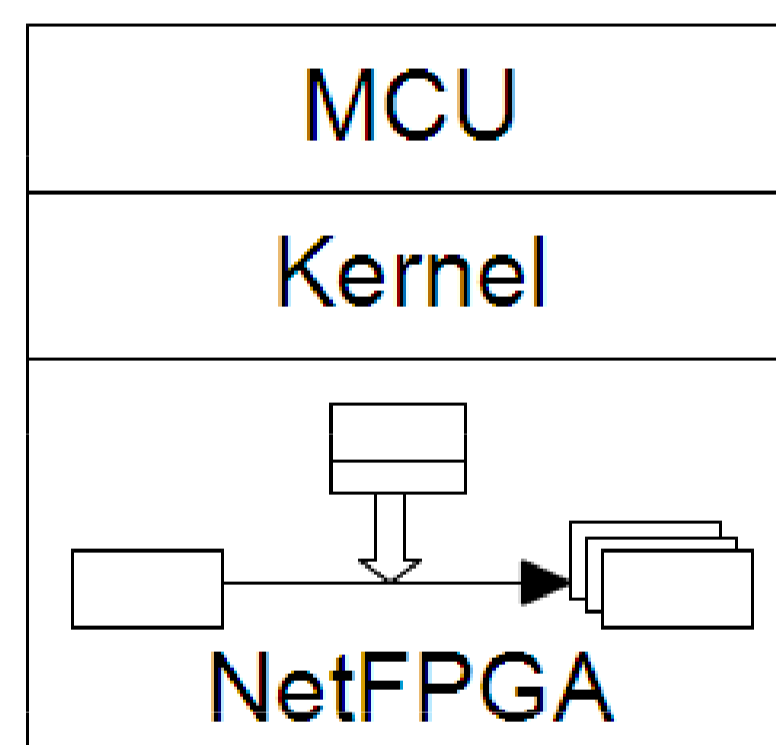
The MusiNet MCU

- NMP is not the same as conferencing
 - Participants prefer to do their own mixing
 - The MCU should only relay data streams
 - Each participant indicates what it wants to receive
 - The MCU maintains a stream routing table
 - Media packets are replicated and forwarded
- A relaying MCU costs 20 ms of delay
 - Too much context switching
 - Too much packet copying
 - Too many packet exchanges



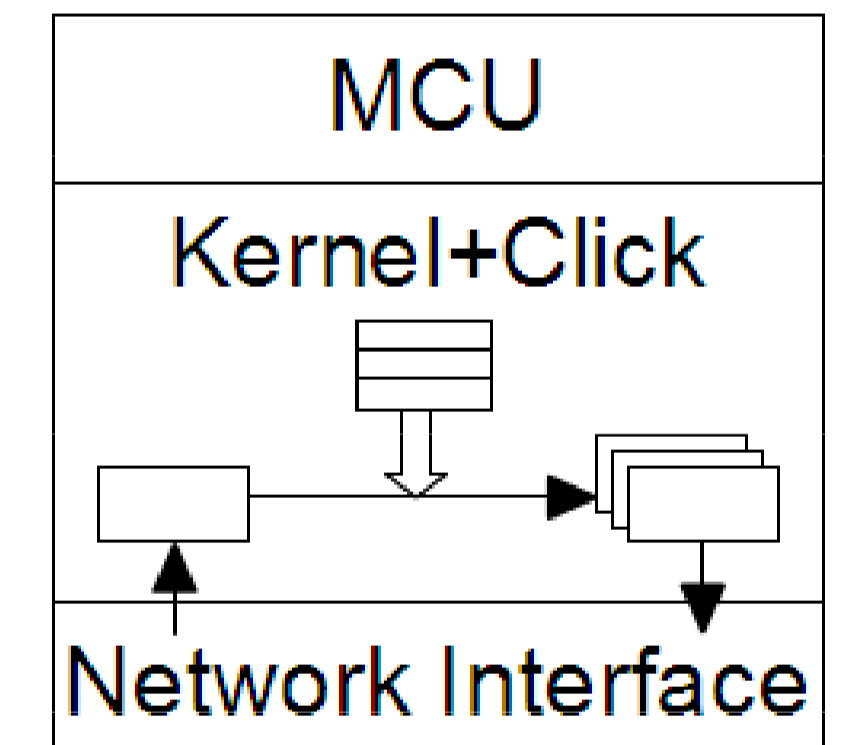
MCU with NetFPGA

- NetFPGA: four network interfaces plus an FPGA
 - Arbitrary processing at the hardware level
 - Split processing between MCU and NetFPGA
 - The MCU receives only signaling packets
 - The routing table resides at the NetFPGA
 - Media packets handled by the NetFPGA
 - Virtually no context switching
 - Packet copying can be eliminated
 - No CPU load for packet routing



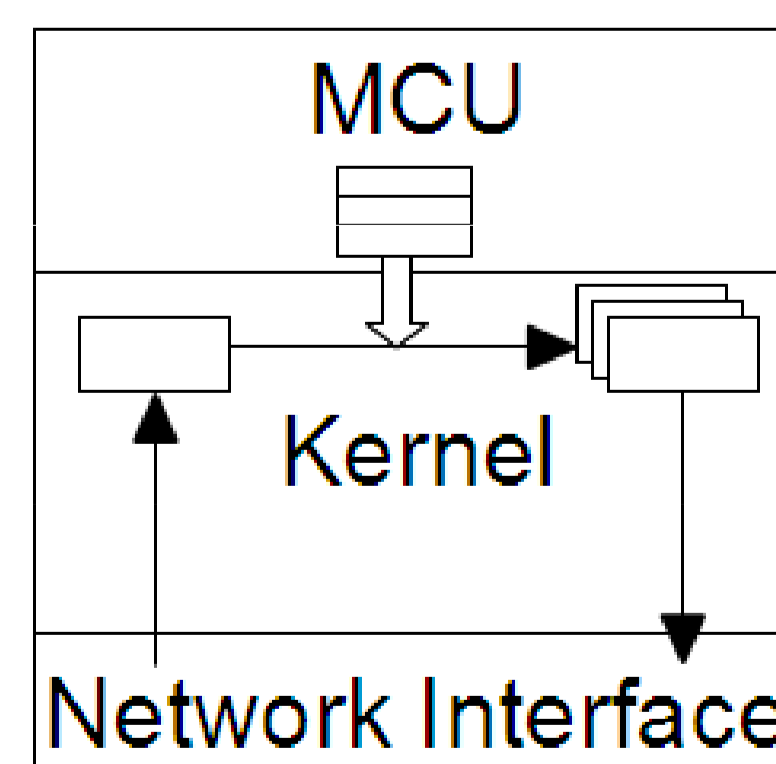
MCU with Click

- The Click modular software router
 - Consists of a set of routing modules
 - Operates at either user or kernel level
 - The MCU receives only signaling packets
 - The routing table resides within Click
 - Media packets handled by Click
 - Test at user level, operate at kernel level
 - Virtually no context switching
 - May be able to eliminate packet copying



MCU with netmap

- The netmap framework for packet handling
 - Applications handle packets in kernel memory
 - No system calls needed to manipulate packets
 - The entire MCU resides at the application level
 - Both signaling and media packets handled by MCU
 - Can use any programming language
 - Packet copying can be eliminated
 - Context switching may be reduced



Conclusion and Future Work

- Three ways to reduce MCU delays
 - Take advantage of hardware (NetFPGA)
 - Move processing to kernel level (Click)
 - Manipulate packets in the kernel (netmap)
- Current work
 - User level Click implementation started
 - Netmap implementation started
- Future work
 - Kernel level Click implementation
 - NetFPGA implementation if needed

