CEOSmp

OCEOSmp - a multicore RISC-V RTOS to police high-reliability applications

RISC-V in Space Workshop

14th December 2022

Michael Ryan, CTO, O.C.E.Technology Ltd



The police ?

- Things can go wrong ..
 - Best to anticipate crime before it actually occurs
 - Keep an eye on the likely culprits
- All designs are based on assumptions
 - Are things happening more often than expected?
 - Are things taking longer than expected?
 - Are things taking more space than expected?
 - Are expected things not happening (or not being noticed)?
 - Is the environment worse than expected?
- Is hardware developing a fault?
 - Do harts agree?
- Robust societies require policing...



OCEOSmp on Polarfire





System policing vs. application policing

- System police automatically collect data on likely culprits
 - Respond automatically in some cases
 - Update system log
 - Call application function
 - Exit RTOS if needed
- Application police can check system data and own data
 - Policing checks (low priority task?)
 - Policing tests (higher priority task?)
- Application police can use system police to handle miscreants
 - Take hart/s out of use
 - Release hart/s back into use
 - Disable tasks, enable tasks
 - Kill tasks that are in execution
 - Exit RTOS with appropriate code



OCEOSmp on NOEL-V



OCEOSmp: design (1)

- Support policing
 - Static allocation of activity records
 - System log used by system and application
 - Record and log still visible if have to exit
- Exclude some temptations stack resource policy (Baker 1991)
 - Unbounded priority inversion cannot occur
 - Chained blocking cannot occur
 - Deadlocks excluded if single hart, warning otherwise
- Save space stack resource policy (Baker 1991)
 - Single system stack per hart rather than per task
- Symmetric all harts used equal after initialisation
 - Harts can be reserved for other uses, e.g. Linux
 - Harts can be reserved for higher priority tasks
- Support schedulability analysis



OCEOSmp on NASA HPSC







OCEOSmp: design (2)

- Task priority fixed
 - based on task importance
- Task pre-emption threshold
 - task pre-empted only by tasks with higher priority than threshold
- Multiple execution instances
 - multiple same task 'jobs' can be in execution typically using different data
- Same priority jobs run to completion in turn
 - no time slicing
- Timed actions independent of scheduling
 - being early can also be a problem...
 - data output at specific time
 - task start request at specific time



OCEOSmp for SiFive



OCEOSmp: design (3)

- Mutexes
 - unbounded priority inversion cannot occur
 - deadlock warning, cannot occur if single core
- Read-Write Mutexes
 - allow simultaneous reads when not held for write
 - prevent writing if being read
 - prioritize write requests over reads
 - order writes first come first served
 - unbounded priority inversion cannot occur
- Counting Semaphores
 - support wait with timeout
- Data Queues
 - support read with timeout







OCEOSmp: design (4)

- System time
 - in microseconds, 64 bit
- Context switch timing
 - context switching shared across all cores
 - context switch time minimized
- Interrupts
 - interrupt disabled timing minimized
 - high priority timer interrupt reserved for timed actions
- Some numbers
 - 1 <= cores <= 255
 - 1 <= tasks <= 255, 1 <= execution instances (jobs) <= 255*15
 - 0 <= mutexes <= 63, 0 <= 63 read-write mutexes <= 63
 - 0 <= counting semaphores <= 63, 0 <= data queues <= 63
 - memory < 20 KiB





GNSS system using OCEOS









OCEOSmp: Using it

- Library components not used not linked into the executable
- Servant not Master started by application main()
- Step 1 : Create application configuration, pass to oceos_init() what cores to use, what stack space, log entries how many tasks, jobs per task, timed actions, how many mutexes, semaphores, data queues
- Step 2: Create corresponding tasks, mutexes, etc. using oceos_task_create() etc.
- Step 3: Use oceos_init_finish() to complete fixed data and checksum
- Step 4: Pass fixed data and initial task (if any) to oceos_start() dynamic data area is set up multi-core scheduling begins







Debug support - DMON

>									OCEOS System	m view								
😐 ena	bled 🛛 🔴 paused																	
Log ID	TimeStamp	Log Type	Delay	DeadLineMargin	stack	CWP	SysStateVar	Log Info					St	ack				
0	00.969 602	SC=>T:0	00.000 213	j	0x6ffffbd0 (39%)	4	0×00000000		0x7000003A									
1	02.425 673 LOG	SYS OK					0x00000000	0x0000cafe										
2	02,425 772 USE	R LOG 0x20					0x00000000	0x0000caf1	0x6FFFFF40									
3	02,425 865	T:0=>T:9	00.000 075		0x6ffff890 (70%)	1	0x00000000	U										
4	03.111 704	T:9=>T:0		00.146 503	0x6ffff890 (70%)	1	0x00000000		0x6FFFFE46									
5	03.111 858	T:0=>T:3	00.685 982		0x6ffff890 (70%)	1	0x00000000											
5	03,249 052	T:3=>T:0		N/A	0x6ffff890 (70%)	1	0x00000000		UxbFFFFD4C									
7	03.249 111	T:0=>T:8	00.000 088		0x6ffff890 (70%)	1	0x00000000		0x6FFFFC52						ר ר			
3	03,437 738	T:8=>T:0		00.040 061	0x6ffff890 (70%)	1	0x00000000						- 11					
9	03,437 795	T:0=>T:4	00.188 673		0x6ffff890 (70%)	1	0x00000000		0x6FFFFB58									
10	03,780 781	T:4=>T:7	00.000 058		0x6ffff560 (101%)	6	0x00000000		×				•	u u	U U	U 4		
11	04.038 013	T:7=>T:4		00.050 989	0x6ffff560 (101%)	6	0×00000000		0x6FFFFASE									
12	04.038.084	T:4=>T:5	00.000 110		0x6ffff560 (101%)	6	0×00000000		0.0000000									
3	04.209 567	T:5=>T:4		00.035 675	0x6ffff560 (101%)	6	0x00000000		UX0FFFF904									
14	04,209,621	T:4=>T:0		00.192.995	0x6ffff890 (70%)	1	0x00000000		0x6FFFF86A									
15	04,209,683	T:0=>T:6	00.000 146		0x6ffff890 (70%)	1	0x00000000											
16	04 288 557	T:6=>T:0	001000 110	00.017.119	0x6ffff890 (70%)	1	0x00000000		0x6FFFF770									
17	04 288 604	T:0=>SC		N/A	0x6ffffbd0 (39%)	4	0x00000000											
8	04.288.657	SC=>SL		1975	0x6ffffd20 (27%)	4	0x00000000		0x6FFFF676									
9	06.474 362	SL=>ISR:9			0x6ffffca8 (31%)	3	0x00000040		0.6000000									
20	06.474.428	ISR:9=>SC			Ox6ffffca8 (31%)	3	0x00000040		UXOFFFF57C									
21	06 474 469	SC=>T:1	00.000.069		Ox6ffffad8 (49%)	2	0x00000040		0x6FFFF482									
22	06.474 478 106	SYS OK	001000 000		exerning (1976)	-	0x00000040	0x0000cafe	5	5	8 9	70	75.	8 8	8	95.	10	110
23	06,509 745	T:1=>SC		-00.035 330	0x6ffffad8 (49%)	2	0x00000040		000	000	000	000	000	000	00	0000	.000	1.000
24	06.509 752 LOG	DEADLINE MISSED				-	0x00000040	0x00000001	00	000	00	00	000	000	00	00 00	8	000
25	06.509 814	SC=>T:2	00.035 437		0x6ffffad8 (49%)	2	0×00000040							Time, s				
26	07.214 794	T:2=>SC		00.116 539	0x6ffffad8 (49%)	2	0x00000040											
27	07.214 843	SC=>SL			0x6ffffd20 (27%)	4	0x0000040				0	Max Stack (0x6FFFI	-FF0) O Stack (0x	5FFFFAD8) O M	in Stack (0x6FFFF570	0)		
								~					10.100 011					
Context	01.344 48	84 02.402 810	03.472	626 04.586	479 05.662	977	06.720 72	4 07.794 53	0 08.866 794	09.943 062	11.058 060	12.117 324	13.190 711	14.251 151	15.316 717	16.378 592	17.435 156	18.522 557
Timeline													_	_				_
						_							_					_
ISK		· · ·						· · · ·		· · ·	- k					· · ·		· · ·
1							- L-											
2			_ ·					-										
3								Log ID	: 26									
5								Timestamp_	. T.2=>SC									
								Stack	: 0x6ffffad8									
7							l l	CWP	: 2		i i				i i			
7									r • 0v40		1							
7 9 4							i	SysStateVa	· • • • • • • • • • • • • • • • • • •									
7 9 4								SysStateVa Margin	: 00.116 539	· ·								
7 9 4 6 8								SysStateVa Margin	:: 00.116 539		:							
7 9 4 6 8								SysStateVa Margin			:							· ·
7 9 4 6 8 0								SysStateVa Margin	: 00.116 539 : 00.116 539					-				
7 9 4 6 8 0 Schedu								SysStateVa Margin	: 00.116 539		: 							

10



Current Status

- OCEOS (single core)
 - SPARC and ARM versions complete (with additional support for GR716 microcontroller) ESA Flight Level B qualification ready
- OCEOSmp (multicore)
 - Checking initial scheduler design on multicore SPARC & RISC-V (ARM later)
 - Initial quad core results (Gaisler GR740 and Microchip PolarFire RISC)
 - Check scheduler distributes work symmetrically
 - 1001 task starts : Per CPU 251,250,250,250
 - Check task can be run in parallel to provide speed up
 - 4096 sample FFT (complex number entries)
 - one task, four jobs in parallel on different parts of FFT
 - speedup factor 3.6 3.7 (haven't tried to optimise)
 - Design closure pending with ESA
- Availability
 - OCEOS single-core development kit on-sale
 - OCEOS multicore beta evaluations available soon



OCEOS task usage & debug screen









e

CPU0	CPU1	CPU2	СРИЗ	% Processing (4096 array)
Initialisation Start CPUs				Preprocessing
	Startup Power down	Startup Power down	Startup Power down	
Setup control block Restart CPUs				
1⁄4 Rearrange	¼ Rearrange	¼ Rearrange	1⁄4 Rearrange	(rearrange if necessary)
¼ Main FFT n-2 stages where array size = 2^n	¼ Main FFT n-2 stages where array size = 2^n Power down	¼ Main FFT n-2 stages where array size = 2^n Power down	¼ Main FFT n-2 stages where array size = 2^n Power down	Main Stage
Setup control blocks for 2 nd last stage Restart CPUs				
Process ¼ 2 nd last stage	Process ¼ 2 nd last stage Power down	Process ¼ 2 nd last stage Power down	Process ¼ 2 nd last stage Power down	<u>2nd Last</u> 5%
Setup control blocks for last stage Restart CPUs				
Process ¼ last stage	Process ¼ last stage Power down	Process ¼ last stage Power down	Process ¼ last stage Power down	<u>Last Stage</u> 5%
Setup control blocks Restart CPUs				
Process ¼ conversion	Process ¼ conversion	Process ¼ conversion	Process ¼ conversion	<u>Conversion</u> <u>Stage</u> (if necessary)





- Thanks to ESA for their support
- Thank you for listening
- Any Questions?
- Any Answers?
 - Would welcome ideas or suggestions e.g. policing data that might be of interest



www.ocetechnology.com

michael.ryan@ocetechnology.com