1 Ewaluacja systemów czasu rzeczywistego

W przypadku potrzeby zastosowania systemu RTS nieraz potrzebne są kryteria jego jakości. Trudno jest samodzielnie tworzyć kryteria oceny i prowadzić badania.

1.1 Wymagania dla systemów czasu rzeczywistego

Wymagania jakościowe stopniowalne oceny systemów czasu rzeczywistego.

- Bezpieczeństwo (ang. *Safety*)
- Dyspozycyjność – (ang. *Dependability*)
- Przewidywalność, nawet w przypadku wystąpienia błędów – (ang. *Behavioural predictability, even in error situations*)
- Prostota – (ang. *simplicity - the simpler the better*)
- Niezawodność - (ang. *Reliability*)
- Tolerowanie błędów - (ang. *Fault tolerance*)
- Własność stopniowej degradacji w przypadku wystąpienia niesprawności - (ang. *Graceful degradation upon malfunctions*)
- Przenośność – (ang. *Portability*)
- Elastyczność – (ang. *Flexibility*)

Wymagania jakościowe binarne oceny systemów czasu rzeczywistego.

- Spełnienie ograniczeń czasowych – (ang. *ability to meet deadlines*)
- Brak nieograniczonych w czasie opóźnień i czasów wykonania – (ang. *no unbounded delays nor arbitrarily long executions*)
- Licencje bezpieczeństwa – (ang. *safety license*)
- Poprawność funkcjonalna – (ang. *functional correctness*)
- Deterministyczne zachowanie – (ang. *deterministic behaviour*)
- Zapobieganie zakleszczeniom – (ang. *deadlocks prevented*)

PDF created with pdfFactory trial version www.pdffactory.com

1.2 Ilościowe miary efektywności systemów operacyjnych czasu rzeczywistego

Podejmując decyzje o wyborze systemu RTS należy dysponować ilościowymi miarami jego oceny. Do oceny jakości działania systemu operacyjnego RTS stosowane są następujące kryteria ilościowe.

Kryteria ilościowe oceny jakości systemu operacyjnego czasu rzeczywistego:

1. Czas opóźnienia przerwania
2. Czas przełączenia kontekstu
3. Czas wywłaszczania
4. Czas przełączenia semafora
5. Czas likwidacji inwersji priorytetów
6. Czas transmisji komunikatu.
7. Czas utworzenia zadania

Dalej będą pokazane przykłady i porównania dotyczące systemów:

- QNX6 Neutrino
- Linux PREEMPT_RT

Linux PREEMPT_RT

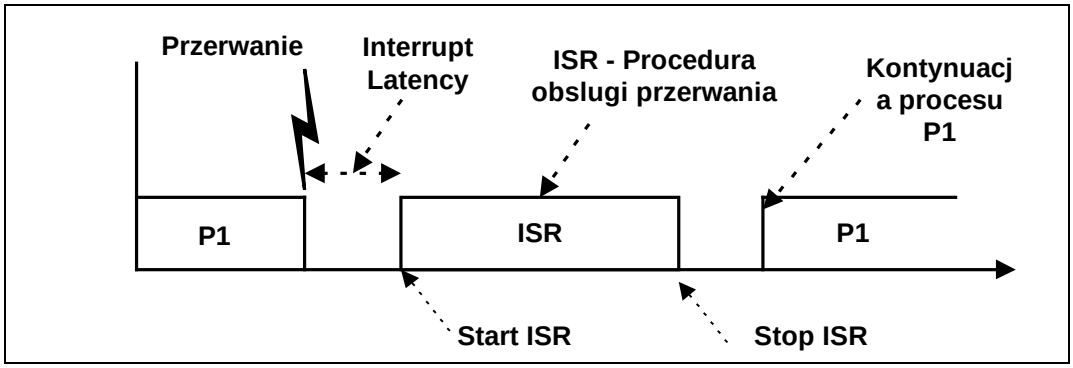
Linux PREEMPT_RT jest projektem rozwijanym przez Ingo Molnar'a (RedHat) oraz Thomas'a Gleixner'a. Jest to łatka która modyfikuje jądro Linux'a, w taki sposób aby stało się w pełni wywłaszczane. Dokonano tego poprzez zamianę „spinloków" w jądrze na wywłaszczane „rtmuteksy", przez co sekcje krytyczne staja się wywłaszczane. Wprowadza także obsługę Timerów POSIX o wysokiej rozdzielczości. Dzięki temu otrzymano rygorystyczny system czasu rzeczywistego.

Testy wykonane na procesorze Intel Pentium IV Mobile o częstotliwości 2 GHz.

PDF created with pdfFactory trial version www.pdffactory.com

1.2.1 Czas opóźnienia przerwania

Czas opóźnienia przerwania (*ang. Interrupt Latency*) jest to czas pomiędzy wystąpieniem przerwania a wykonaniem pierwszej instrukcji procedury obsługi tego przerwania (*ang. Interrupt Service Routine – ISR)*.



Rys. 1-1 Ilustracja czasu opóźnienia obsługi przerwania

Czas opóźnienia obsługi przerwania zależy od następujących czynników:

1. Czasu zablokowania przerwań - przerwania w procesorze mogą być chwilowo zablokowane przez inny wątek lub handler w celu ochrony sekcji krytycznych.
2. Obsługi innych przerwań - kontroler przerwań może nie dopuścić do przyjęcia przerwania gdyż przerwanie o wyższym priorytecie jest w trakcie obsługi

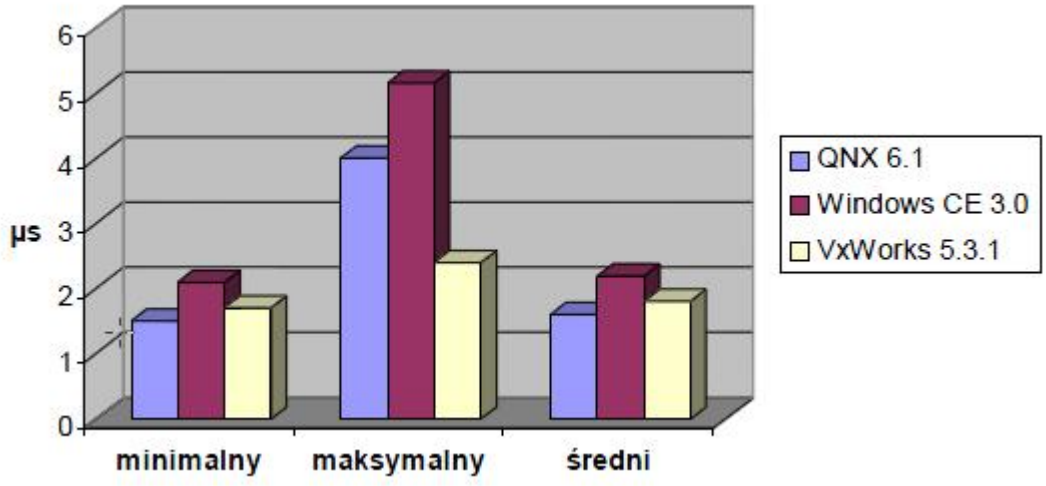
Czas opóźnienia przerwania zależy od aktualnego stanu systemu. Przerwania w danym momencie czasu mogą być zablokowane lub też nie. W związku z tym definiuje się maksymalny czas opóźnienia przerwania.

Maksymalny czas opóźnienia przerwania (*ang. worst case interrupt latency)*

Maksymalny czas opóźnienia przerwania to czas opóźnienia przerwania otrzymany dla najmniej korzystnego przypadku.

PDF created with pdfFactory trial version www.pdffactory.com

☞ W systemie operacyjnym czasu rzeczywistego najdłuższy czas w którym przerwania pozostają zablokowane powinien być jak najkrótszy.

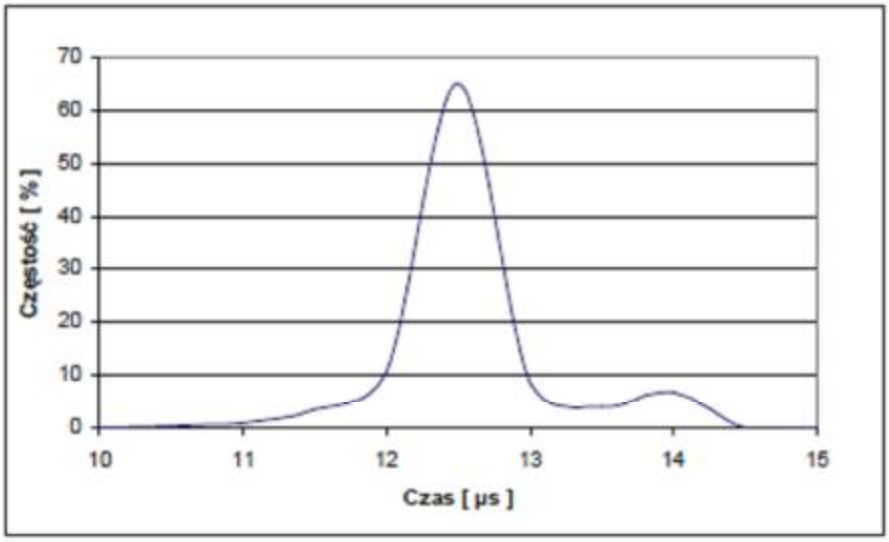


Rys. 1-2 Czas opóźnienia przerwania dla różnych systemów czasu rzeczywistego (Pentium 200 MHz) wg. Dedicated Systems

Przypadek	Maksymalny czas [µs]	Średni czas [µs]
System nie obciążony	14,807	12,339
System obciążony	17,194	14,640

Rys. 1-3 Czas opóźnienia przerwanie systemu QNX6 Neutrino - praca dypl. Łukasz Biały

PDF created with pdfFactory trial version www.pdffactory.com



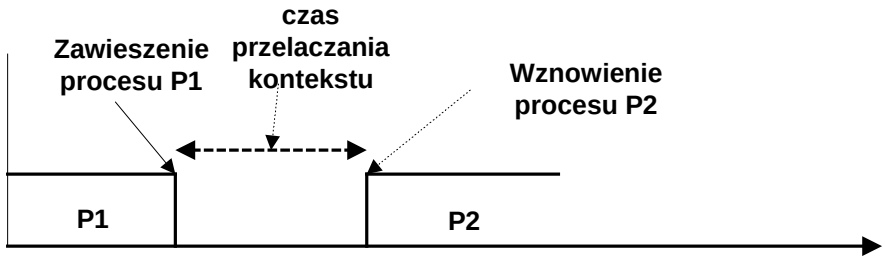
Rys. 1-4 Histogram czasu reakcji na przerwanie w systemie QNX6 Neutrino – praca dypl. Łukasz Biały

PDF created with pdfFactory trial version www.pdffactory.com

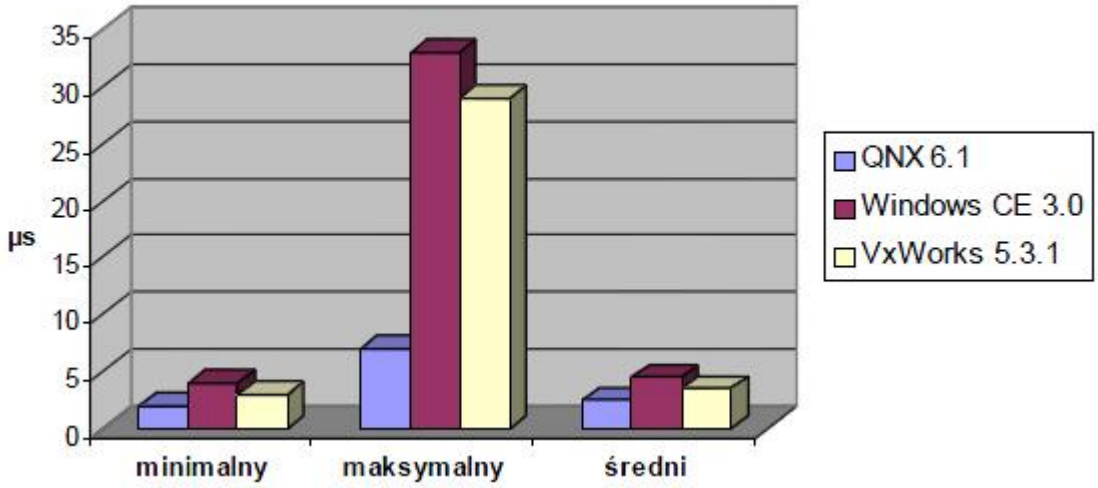
1.2.2 Czas przełączania kontekstu

Czas przełączania kontekstu zadań (*ang. task switch time*) jest to czas, w którym system operacyjny jest w stanie przełączyć aktualnie wykonywane zadanie na inne zadanie w systemie o tym samym priorytecie. Na ten czas składa się czas potrzebny na zachowanie kontekstu procesu bieżącego P1 i odtworzenie kontekstu procesu P2 podlegającego zaszeregowaniu. Długość czasu przełączania zadań uzależniona jest od:

- Złożoności struktur danych opisujących zadania
- Efektywności procedur zmieniających kontekst zadań.
- Efektywności algorytmu szeregowania



Rys. 1-5 Ilustracja czasu przełączenia kontekstu



Rys. 1-6 Czas przełączenia kontekstu dla różnych systemów czasu rzeczywistego (Pentium 200 MHz) – wg. Dedicated Systems

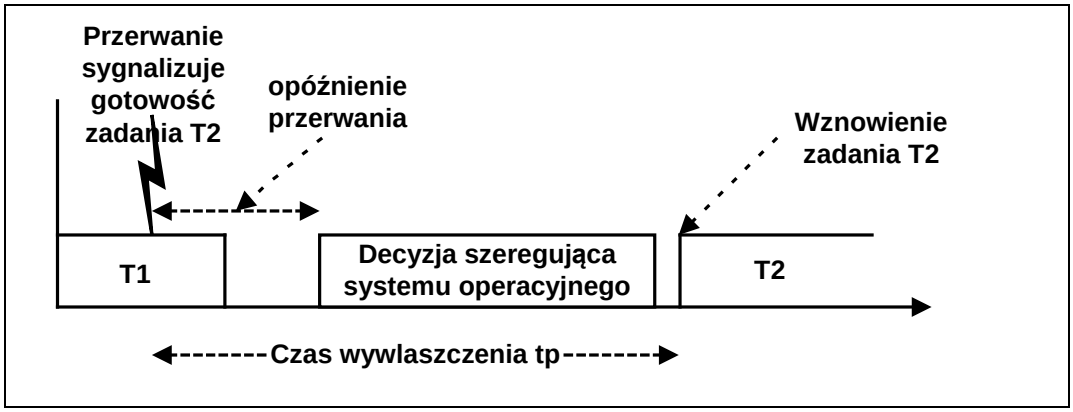
PDF created with pdfFactory trial version www.pdffactory.com

System	Ilość wątków	Maksymalny czas [µs]	Średni czas[µs]
QNX 6 Neutrino	2	7,527	0,722
	10	9,149	0,734
	128	10,737	0,745
PREEMPT_RT	2	45,906	2,234
	10	46,244	2,270
	128	47,802	2,286

Tab. 1-1 Wyniki pomiarów czasu przełączania wątków – praca dypl. Łukasz Biały

1.2.3 Czas wywłaszczania

Czas wywłaszczania (*ang. preemption time*) jest to średni czas potrzebny na wywłaszczenie zadania T1 o niższym priorytecie, przez zadanie T2 o wyższym priorytecie.



Rys. 1-7 Ilustracja czasu wywłaszczenia tp zadania T1 o niższym priorytecie przez zadanie T2 o priorytecie wyższym

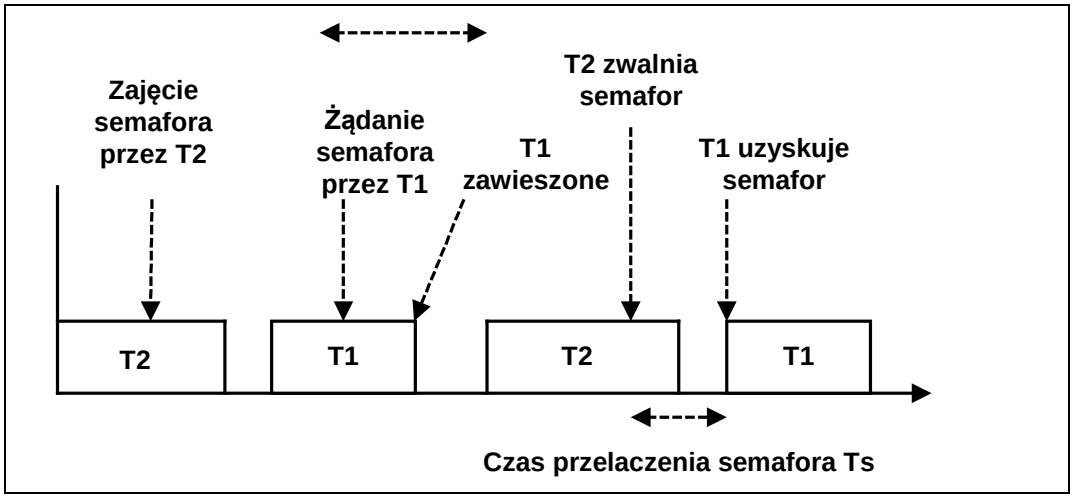
Na czas ten składa się:

1. Czas opóźnienia przerwania
2. Podjęcie decyzji szeregującej
3. Czas zachowanie kontekstu procesu bieżącego i przywrócenia kontekstu procesu wywłaszczającego

PDF created with pdfFactory trial version www.pdffactory.com

1.2.4 Czas przełączenia semafora

Czas przełączenia semafora (ang. *Semaphore shuffling time*) to średni czas pomiędzy zwolnieniem semafora przez aktualnie wykonywane zadanie, a uruchomieniem pierwszej instrukcji zadania oczekującego na ten semafor. Czas ten zależy głównie od sposobu implementacji mechanizmu semaforów w danym systemie operacyjnym. W systemie czasu rzeczywistego, gdzie wiele aplikacji korzysta ze wspólnych zasobów, krótki czas operacji semaforowych jest bardzo ważnym czynnikiem.



Rys. 1-8 Ilustracja czasu Ts przełączenia semafora

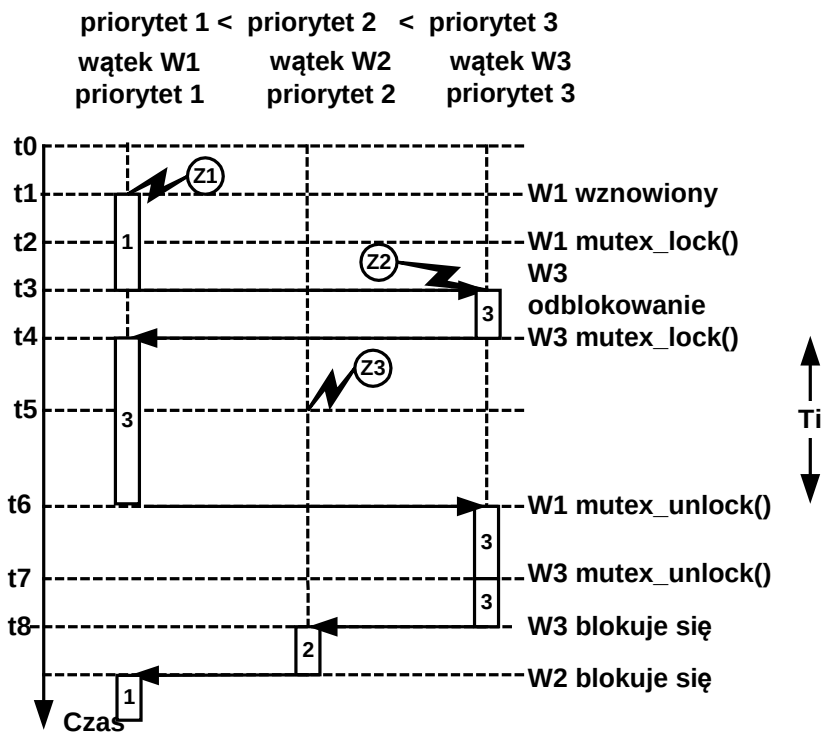
System	Obciążenie	Maksymalny czas [µs]	Średni czas[µs]
QNX 6 Neutrino	NIE	7,862	1,021
	TAK	10,653	1,049
PREEMPT_RT	NIE	53,135	10,167

Tab. 1-2 Wyniki pomiarów czasu operacji semaforowych - praca dypl. Łukasz Biały

1.2.5 Czas likwidacji inwersji priorytetów (ang. *deadlock breaking time*)

Sytuację, w której zadanie o wyższym priorytecie wywłaszczy zadanie o niższym priorytecie, które przetrzymuje zasób wymagany przez zadanie o wyższym priorytecie, nazywamy inwersją priorytetów. Czas likwidacji inwersji jest to średni czas, w którym system sobie poradzi z inwersją.

PDF created with pdfFactory trial version www.pdffactory.com

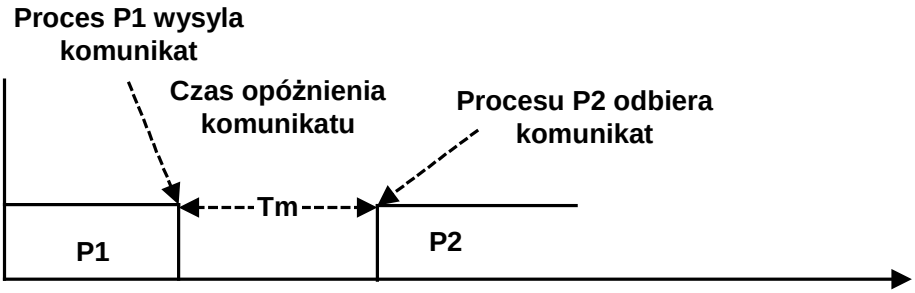


Rys. 1-9 Ilustracja czasu Ti likwidacji inwersji priorytetów (dziedziczenie priorytetów)

Czas likwidacji inwersji priorytetów to czas pomiędzy zażądaniem zasobu przez wątek o najwyższym priorytecie a jego otrzymaniem. Zależy on od jakości mechanizmu zapobiegania inwersji priorytetów.

1.2.6 Czas opóźnienia przesyłania komunikatów

Czas opóźnienia przesyłania komunikatów Tm (ang. *intertask message latency*) to czas jaki mija pomiędzy wysłaniem komunikatu przez jeden proces P1 a odbiorem komunikatu przez inny proces P2. Przepustowość komunikacji międzyzadaniowej (ang. *Datagram throughput*) to ilość bajtów na sekundę, jakie mogą zostać przesłane pomiędzy dwoma zadaniami przy pomocy komunikatów (*ang. Message passing*).



Rys. 1-10 Czas opóźnienia Tm przesyłania komunikatów

PDF created with pdfFactory trial version www.pdffactory.com

1.3 Testy

Testy (ang. *benchmark*) to programy lub zestawy programów[1], które zostały wybrane lub zaprojektowane w celu testowania i/lub pomiaru wydajności obliczeniowej systemów informatycznych. Powinny one pomagać podczas projektowania systemu lub w podjęciu decyzji o zakupie danego systemu informatycznego.

Kelvin Obenland podzielił zestaw kryteriów porównawczych, istotnych dla systemów zgodnych z POSIX, na dwie kategorie.

- Testy które mierzą determinizm danego systemu operacyjnego,
- Testy które mierzą opóźnienia szczególnie ważnych operacji

Pierwsza kategoria oprogramowania mierzy czas odpowiedzi systemu, rozstrzyga czy wątki działają w sposób przewidywalny.

Druga kategoria mierzy czas zmiany kontekstu pomiędzy zadaniami. Mierzy przepustowość danych pomiędzy procesami i wątkami oraz opóźnienie sygnałów POSIX.

Szerzej znane są następujące testy:

- Rhealstones
- Oszacowanie trójwymiarowe
- MiBench
- Hardstones

PDF created with pdfFactory trial version www.pdffactory.com

1.3.1 Rhealstone

Rhealstone jest zestawem sześciu programów napisanych w języku C. Każdy z nich mierzy czas operacji charakterystycznej dla systemów czasu rzeczywistego:

1. Czas przełączenia zadań - *task-switch time*,
2. Czas wywłaszczenia - *preemption time,*
3. Czas opóźnienia przerwania - *interrupt latency* ,
4. Czas przełączenia semafora - *semaphore-shuffle time*,
5. Czas likwidacji zakleszczenia - *deadlock-break time*,
6. Opóźnienie przesyłania komunikatów - *intertask message latency.*

Z otrzymanych czasów obliczana jest wartość średnia na podstawie wzoru:

$$t = (t_1 + t_2 + t_3 + t_4 + t_5 + t_6)/6$$

Czasy wyrażone są w sekundach. Tak zwaną wartość Rhealstone definiujemy jako odwrotność wyliczonej średniej

$$\frac{1}{t}$$

Im większa jest wartość tym system szybszy.

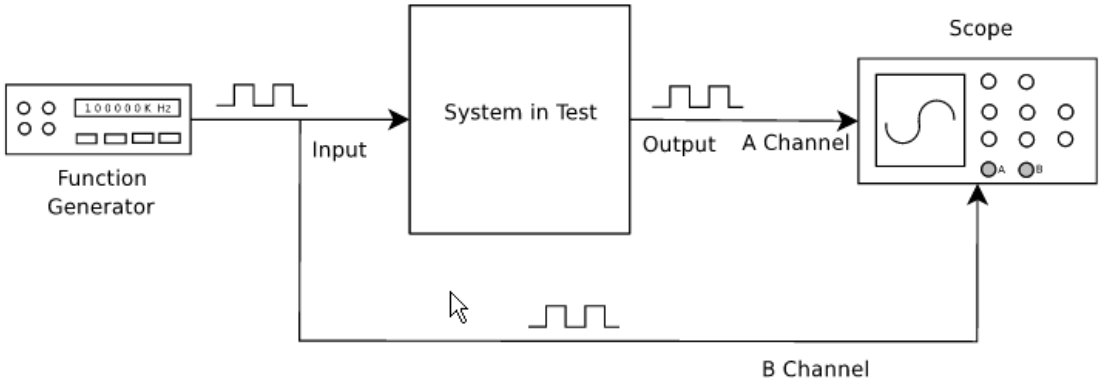
Wady testu Rhealstones:

- Mierzy wartości średnie a nie dla najgorszego przypadku.
- Wynik jest sumą ważoną składników, nie ma jednoznacznych ustaleń co do wag.

PDF created with pdfFactory trial version www.pdffactory.com

1.4 Inne testy

Inny test porównujący własności czasowe systemów operacyjnych podaje praca [5] . Pomiar zachodzi w układzie jak poniżej:



Mierzono:

A) Maksymalną częstotliwość przerwań do której przerwania nie są gubione. W tabeli podano odwrotność tej częstotliwości.

B) Zwłokę w czasie obsługi przerwań (ang. *interrupt latency*)

C) Nierównomierności w czasie reakcji na przerwanie (ang. *interrupt jitter*)

Table 1. Worst times measured during the experiments. A: Response Time (1/maximum sustained frequency), B: Latency, C: Latency Jitter

	Win XP	Win CE	Neutrino	μC/OS-II	Linux	RTAI	VxWorks
A	200μs	20μs	20μs	1,92μs	13,89μs	5μs	3,85μs
B	848μs	99μs	35,2μs	3,2μs	98μs	11,4μs	13,4μs
C	700μs	88,8μs	32μs	2,32μs	77,6μs	7.01μs	10,4μs

Tab. 1-3 Wyniki pomiarów własności różnych RTOS według pracy [5]

PDF created with pdfFactory trial version www.pdffactory.com

1.5 Certyfikaty

Certyfikaty stanowią pewien rodzaj referencji i mogą pomóc przy wyborze systemu operacyjnego.

Norma ISO 9001:2000

Dotyczy zarządzania jakością oferowanych usług oraz produktów i definiuje wszystko od odpowiedzialności kierownictwa, przez zarządzanie zasobami aż do realizacji wyrobu. Certyfikat nadany przez ISO gwarantuje ona powtarzalność procesu produkcji, a także ciągłe ulepszanie produktów i usług na rzecz zadowolenia klientów.

POSIX PSE52 Realtime Controller 1003.13-2003

Certyfikat ten nadawany przez IEEE oraz The Open Group zapewnia zarówno przenoszalność kodu a także spełnia kryteria czasu rzeczywistego wymagane przez systemy militarne, sieciowe oraz motoryzacyjne.

Common Criteria (ISO/IEC 15408) EAL4+

Common Criteria to norma pozwalająca w sposób formalny weryfikować bezpieczeństwo systemów teleinformatycznych. Udostępnia ona procedury pozwalające na zdefiniowanie zagrożeń oraz zabezpieczeń, które na te zagrożenia odpowiadają, a następnie przeprowadzenie formalnej weryfikacji ich faktycznego działania w produkcie. Certyfikacją według normy CC zajmują się niezależne, akredytowane laboratoria badawcze na całym świecie. Wynikiem procesu certyfikacji jest tzw. "profil ochrony" (PP – ang. *protection profile*), który definiuje zabezpieczenia stosowane przez produkt oraz certyfikat, potwierdzający ich faktyczną skuteczność.

Proces certyfikacji może być prowadzony według różnych poziomów szczegółowości i weryfikacji formalnej (EAL – ang. *Evaluation Assurance Level*), począwszy od EAL1 (tylko testy funkcjonalne) aż do EAL7 (formalna weryfikacja projektu oraz testy).

System QNX Neutrino w wersji 6.4 posiada certyfikat na poziomie EAL4+. Przeprowadzone testy dotyczyły jądra systemu, przetwarzania wieloprocesorowego oraz bezpiecznego partycjonowania.

PDF created with pdfFactory trial version www.pdffactory.com

IEC Safety Integrity Level (SIL)

Certyfikat nadawany przez IEC (ang. *International Electrotechnical Commision*) w zakresie bezpieczeństwa systemu. Potwierdza on najwyższą możliwą redukcję ryzyka osiągalną przy używaniu systemu jednoprocesorowego w takich gałęziach przemysłu jak motoryzacja, przemysł ciężki, górnictwo oraz w innym gdzie bezpieczeństwo jest priorytetem. Dla zapewnienia wiarygodności certyfikat jest regularnie weryfikowany.

System QNX Neutrino spełnia wymagania poziomie SIL3 (SIL – ang. *Safety Integrity Level*).

1.6 Źródła

[1] Roman Gumzej, Wolfgang A. Halay; Real Time Systems Quality of Service, wyd Springer 2010.

[2] Łukasz Biały, Praca dyplomowa, Politechnika Wrocławska 2010.

[3] Michał Stangret, Praca dyplomowa, Ocena wybranych systemów operacyjnych czasu rzeczywistego, Wrocław 2009.

[4] Larisa Rizvanovic, Comparison between Real time Operative systems in hardware and software. Mälardalens University.

[5] Rafael V. Aroca , Glauco Caurin A Real Time Operating Systems (RTOS) Comparison

[6] Strona Linux RT, https://rt.wiki.kernel.org/

[7] Rabindra P. Kar and Kent Porter, RHEALSTONE: A REAL-TIME BENCHMARKING PROPOSAL

PDF created with pdfFactory trial version www.pdffactory.com