

Static State Scheduling of Resources for Embedded Systems

Yue He

College Of Geophysics And Information Engineering,
China University of Petroleum
Beijing, China

Ming Chen

The People's Procuratorate in Changping District
Beijing, China

Abstract—Embedded computers have become pervasive and complex. This approach entails at least three serious consequences: firstly, profligate and rigid usage of resources necessitated by the binding of subsystems to hardware platforms; secondly, significantly more difficult system integration because individually developed and tested systems are not guaranteed to work in combination; thirdly, the lack of higher-level system control because conceptually indivisible functions are isolated on the hardware level. These problems have caused not only financial loss such as unusable systems due to the high cost of unjustified resource redundancy and integration failures, but also the loss of human lives as exemplified by a number of fatal accidents induced by the third problem. It is critical that system engineers have a solid basis for addressing these fundamental design problems in large-scale real-time embedded systems. Towards this end, in this dissertation we introduce the notion of a Real-Time Virtual Resource (RTVR) which operates at a fraction of the rate of the shared physical resource and whose rate of operation varies with time but is bounded. Tasks within the same task group are scheduled by a task level scheduler that is specialized to the real-time requirements of the tasks in the group. The scheduling problems on both task level and resource level are analyzed.

Keywords-Embedded System, Resource Locking, Scheduling

I. INTRODUCTION

For each district S_k (α_k, Δ_k) resource demand, constructs the dispatch on the resources. The attention, the parameter to (α_k, Δ_k) needs to obtain the quantity α_k processor resources district and the district time delay is only not bigger than DELTA. It regarding execution time and cyclical not any limit. This characteristic causes dispatcher's construction to be very flexible. In this part, first discusses the district scheduling problem in the sole stratification plane, for instance special resources direct dispatch district; Then A will delay the question to the multi-stratification planes, for instance district recursion dispatch district.

Static State Scheduling Of Resources

The paper will introduce a static state scheduling of resources algorithm. In this algorithm, the resources the minimum value circulation which $T_k = (\Delta_k / (1 - \alpha_k))$ was equal to by the cycle dispatch each district, and each district assignment's processor resources with α_k become the

proportion. If district T_k essentially is different, possible conservative to adjust them to form an overtone chain, in which regarding all i and j , if T_i is smaller than T_j , T_j is the T_i multinomial. Thus, this kind of static state scheduling of resources can duplicate in each principal period, the principal period length equates in the T_k maximum value. Each principal period divides into several lengths is the T_k minimum value minor cycle. This fundamentally reduced quantity which the context transforms.

Static RLS is suitable for all districts fixes, and their parameter known system. The algorithm basic philosophy is through the district which unites each may dispatch effectively composes a goal district. Supports this dispatcher construction to involve to two theorems.

Static resource partition model

Intuitively, a static resources district is a numerous time-gap set merely. In these gaps, in the district the duty group which dispatches may use the physical resources. In this, the research is assigns in the district the duty group scheduling problem, this district's time-gap assigns explicitly by the tabulation. Through using the supplied function technology, has analyzed this model static priority dispatch and the dynamic priority dispatch separately. Obtains based on the critical district's essential concept may dispatch the result. Finally, discussed the district (second-level) own scheduling problem.

Definition

Defines 1 Resources district Π is two tuples (Γ, P) , in which Γ is a N time-gap array: $\{(S_1, E_1), (S_2, E_2), \dots, (S_N, E_N)\}$, in which satisfies $(0 \leq S_1 < E_1 < S_2 < E_2 < \dots < S_N < E_N \leq P)$, $N \geq 1$, P is the district cycle. In this district carries out the duty group can only in $(S_i + j \times P, E_i + j \times P)$. In the time-gap obtains the physical resources. In which $1 \leq i \leq N, j \geq 0$.

The above definition enumerated has assigned to a district each time-gap, was the periodic district plan (including these dispatched dynamic production by online district) the general expression. It has provided a beginning which considered for definition district's other methods. Becomes the district the processor regarding the district not available that period of time-gap the blocking time. In the conventional model, the duty group has the special-purpose resources available, therefore does not have the blocking time. In this paper, may take it peculiar circumstance

regarding, the corresponding district may indicate like this: $\prod = \{(0, P)\}, P$

Theorem 1

If all district's use factor is some values, and the sum total is smaller than was equal to that then these districts are may dispatch.

Example 1 use factor respectively be $1/2, 1/4, 1/8, 1/8$ the district $\prod_1, \prod_2, \prod_3, \prod_4$ in the cycle is 8, assignment time slot for (1,2,1,3,1,2,1,4) the private resource on is very easy to dispatch.

Theorem 2

k supplies regularity is 1 district composition district supplies regularity is k.

Defines 1

The use factor is a, the supplies regularity for R_s the district \prod adjustment use factor (Adjusted Availability Factor) $AAF(\alpha_i, R_{si})$ constitutes the district \prod all district use factor sum total.

Regarding has N district, and the ith district's use factor is α_i , supplies regularity for $R_{si} (1 \leq i \leq N)$ the system, its district hypergene becomes the algorithm to be as follows:

According to following method computation each district $\prod_i AAF(\alpha_i, R_{si})$.

If $R_{si} = 1$ $AAF(\alpha_i, R_{si}) = \frac{1}{2^k}$, among $k = \lfloor \log_{\frac{1}{2}} \alpha_i \rfloor$,

Otherwise $AAF(\alpha_i, R_{si}) = \frac{1}{2^{i1}} + \frac{1}{2^{i2}} + \dots + \frac{1}{2^{iR_{si}}}$, among

If not for $\frac{\log_{\frac{1}{2}} \alpha_i}{2}$ integer $i_1 = \lfloor \log_{\frac{1}{2}} \alpha_i \rfloor$,

Otherwise $i_1 = \lfloor \log_{\frac{1}{2}} \alpha_i \rfloor - 1$.

At that time $k \in [2, R_{si} - 1]$,

If not for $\log_{\frac{1}{2}} (\alpha_i - \sum_{z=1}^{k-1} \frac{1}{2^z})$ integer,

then $i_k = \lfloor \log_{\frac{1}{2}} (\alpha_i - \sum_{z=1}^{k-1} \frac{1}{2^z}) \rfloor$,

Otherwise $i_k = \lfloor \log_{\frac{1}{2}} (\alpha_i - \sum_{z=1}^{k-1} \frac{1}{2^z}) \rfloor - 1$ $i_{R_{si}} = \lfloor \log_{\frac{1}{2}} (\alpha_i - \sum_{z=1}^{R_{si}-1} \frac{1}{2^z}) \rfloor$.

The record $2^{iR_{si}} = P_i$, P_i is the district \prod_i cycle.

If $\sum_{i=1}^N AAF(\alpha_i, R_{si}) > 1.0$, procedure withdrawal.

Assigns the M time slot, among $M = \max \{P_i | 1 \leq i \leq N\}$, M is the district table cycle.

Regarding each $\prod_i, 1 \leq i \leq k$, has

$$AAF(\alpha_i, R_{si}) = \frac{1}{2^{i1}} + \frac{1}{2^{i2}} + \dots + \frac{1}{2^{iR_{si}}}$$

To each $k \in [1, R_{si}]$, in the entire M time slot, assigns 1 from each ik time to give \prod_i .

The surplus time slot will assign for the non-real-time task or the system district

Will assign for same district all neighboring time slots connects.

The following example showed how does have 3 real-time

request $\{(\alpha_i, R_{si}), i \in [1, 3]\} (0.375, 2), (0.25, 2), (0.25, 1)$ like AVP system to produce the district table.

Computation $AAF(\alpha_i, R_{si})$

$$AAF(\alpha_1, R_{s1}) = 0.25 + 0.125; P_1 = 8$$

$$AAF(\alpha_2, R_{s2}) = 0.125 + 0.125; P_1 = 8$$

$$AAF(\alpha_3, R_{s3}) = 0.25 + 0.125; P_1 = 4$$

$$\sum_{i=1}^N AAF(\alpha_i, R_{si}) = 0.875$$

$$M = \max \{P_i | 1 \leq i \leq N\} = 8$$

Time slot assignment as shown in Table 1.

TABLE I. PU DISTRICT PRODUCTION TABLE

District table [i]	Hypothesized CPU ID	Start	Conclusion
0	AVP 1	0	1
1	AVP 3	1	2
2	AVP 1	2	3
3	AVP 2	3	4
4	AVP 1	4	5
5	AVP 3	5	6
6	AVP 2	6	7
7	SVP	7	8

Connect VP all neighboring time slots.

II. DYNAMIC RESOURCE DISPATCH

This method may the effective dispatcher different detention demand district. For all this, it is not suitable the district which the dynamic connection and leaves. In order to solve this problem, below will introduce the dynamic resource dispatch the concept.

Noted real time system's researchers already the extensive research similar dynamic task scheduling question, wanted through to establish the scheduling of resources and the task scheduling relation explores the past the research. With this way, may apply possibly the task scheduling result in the scheduling of resources.

Theorem 3

Supposition is regarded a district execution the real-time task execution, its computing time is C, the cycle is P. No matter uses any dispatch algorithm, the synthesis district satisfies $\alpha = C/P, \Delta \leq 2P - 2C$.

Deduces 1

A district's (α, Δ) dispatcher may transform for the execution time is $\Delta\alpha/(2(1-\alpha))$, districts Π (0.2,40ms)
 $C = \Delta\alpha/(2(1-\alpha)) = 0.2 \times 40 / (2 \times (1-0.2)) = 5ms$
 $P = \Delta/2(1-\alpha) = 40/2 \times (1-0.2) = 25ms$. Therefore, the dispatcher Π may transform for the dispatch computing time is 5ms, the cycle is the 25ms duty cycle for $(\Delta/2(1-\alpha))$ duty dispatch.

When carries out RTVR, the observation use factor's world of mortals will cause the resources the low efficiency use. In order to solve this problem, introduces one novel about two level of district theories.

This scheduling theory's basic thought is possesses these to have the small district time delay and the small use factor district combines, and regards as them is the small district time delay and a greater use factor (all small district use factor sum total) district. When this district is dispatched, divides the dispatch quantum for the small time slot and assigns them to the primitive small district. A more detailed statement is as follows:

District merge: All small districts (α_i, Δ_i) merge a district $\Pi(\alpha, \Delta)$, among $\alpha \geq \sum \alpha_i, \Delta \leq \Delta_i - Q$.

District dispatch: When a time slot assigns gives the zone time Π , will divide into N small time slot. $\lceil (N\alpha_i/\alpha) \rceil$ The small time slot is assigned for the district Π_i . Time slot by i smooth execution.

District permission: If has n time slot in N small time slot not to use, then the district Π will accept a new district $\Pi'(\alpha', \Delta')$, and the new district will satisfy the following condition:

- (a) $\Delta' \geq (\Delta + Q)$
- (b) $\alpha' \leq (n/N) \times \alpha$

The district leaves: If a district leaves, is not leaves behind one in during the small time slot "the hole", but will be possesses compared to this district order high district before this district will carry out. This meant similarly all active partition first will carry out, the small time slot which if and the existence has not used or which is left the district stays behind, then they will be in all small time slots finally are carried out. This and the compression hour crack is very similar.

Based on regularity district scheduling of resources

Because of has the unique nature based on the regularity district as a result of its entire number field, therefore separates to their scheduling of resources's discussion with other district.

Regular district

Theorem 4

A regular district except the counter-balance, determined only by its use factor.

Theorem 5

Assigns a set $\{n_i/p_i, 1 \leq i \leq m\}$ to represent the regular district the use factor, whether to have a dispatcher to be able to contain all districts the determination question is a NP question.

Although this question is the NP question, so long as but the new use factor is higher than the old use factor, always may transform the use factor to be possible to dispatch the set.

Theorem 6

If the regular district's use factor is some digital involution, and the total use factor is smaller than is equal to 1.0, then this regular district may dispatch.

The use factor respectively is $1/2, 1/4, 1/8, 1/8$ the regular district $\Pi_1, \Pi_2, \Pi_3, \Pi_4$ may very easily in the cycle be 8, assignment time slot for (1,2,1,3,1,2,1,4) the private resource on dispatches.

Theorem 7

Assigns a set $\{n_i/p_i, 1 \leq i \leq m\}$ to represent the regular district the use factor, if $\sum_{i=1}^m \frac{n_i}{p_i} \leq 0.5$, then these districts may transfer.

Irregular district

Theorem 8

Comes from the same resources two regular district Π_1, Π_2 union to compose a new supplies regularity is 2 new districts together Π_3 .

Proved: Supposes $I_1(t), I_2(t), I_3(t)$ distinguishes representative's Π_1, Π_2, Π_3 immediate regularity function. Regarding $\forall a, \forall b, a < b$, has:

$$|I_1(b) - I_1(a)| < 1$$

$$|I_2(b) - I_2(a)| < 1$$

$$\begin{aligned}
& |I_3(b) - I_3(a)| \\
&= |I_1(b) + I_2(b) - I_1(a) - I_2(a)| \\
&\leq |I_1(b) - I_1(a)| + |I_2(b) - I_2(a)| \\
&< 1 + 1 \\
&= 2
\end{aligned}$$

Requests two regular districts is comes from the same resources the reason is for the time slot which guarantees them not to be impossible to have conflicts. If the district is comes from the different resources to look like in the distributed environment situation such, then that kind of conflict occurs on the possibility. In this paper does not do to this question further studies.

When in the example district Π^1 and Π^3 combines in together, a use factor is $5/8$, the supplies regularity will be 2 new districts will produce.

The supplies regularity is the district the supply capacity which obtains actually in any time section is more than or is short the supply capacity quantity upper limit which can obtain in the district tentative plan.

These two kinds of regularities in fact may exchange.

When the temporary regularity is an integer, the supplies regularity may be a real number. The temporary regularity suits in the task scheduling because of its real-time characteristic, but supplies the regularity to suit in the scheduling of resources. Two kind of regularity exchange two rank's dispatches connect, and causes all levels of dispatches to concentrate in oneself cared.

Task Scheduling. Assigns a district, now analyzes a possibility in the district available time interval execution work team to be possible only the dispatcher question. Studies this question first in the regular district, then in irregular district.

Static Priority. The use factor boundary's computation needs to consider in the district environment the critical state concept. Is suitable similarly from the previous conclusion for the integer time domain. In this, will calculate the critical district and the regular district use factor.

Because the task time senate number field is an integer, therefore the duty execution time increases by the integer. In order to calculate the use factor boundary, presently may not dispatch the duty group through the computation worst situation in the minimum use factor to substitute in the computation same situation to be possible to dispatch the duty group the maximum use factor. Namely through to τ_1 or τ_2 the execution time increases 1 to calculate

Dynamic Priority. After having studied in the regular district static priority dispatch use factor boundary, below

gives two lemmas, afterward proved that the dynamic priority dispatch the proof is the same.

Regarding real number m and $n^{\lfloor m \rfloor + \lfloor n \rfloor \leq \lfloor m+n \rfloor}$.

Contains n periodic duty group G to use the EDF dispatch algorithm to be possible to dispatch in Π the regular district, if $U(G) \leq \alpha(\Pi)$.

Irregular district

The district Π hypothesized time $V(t) = \lfloor S(t)/\alpha(\Pi) \rfloor$, in which $S(t)$ is Π supplied function.

The district Π hypothesized time dispatch is a dispatcher restraint, as the matter stands, work's most appropriate running time is when it is released, and the releasing time does not surpass the district Π the current hypothesized time.

The hypothesized time dispatch is suitable possibly for all for the private resource design dispatch algorithm. Therefore, the existence has the hypothesized time speed monotonous dispatch, hypothesized time EDF dispatch, with other hypothesized time dispatch algorithm.

III. CONCLUSION

In this paper, we proposed an interesting question is if the operating system is not higher than the present operation by one is with the clock rate support interrupt can be what kind. But, this question will not bring a true question. Because in the reality, the computer hardware's real-time clock by is higher than 100Hz obviously the speed movement. This kind of huge frequency disparity may because in the dispatch expenses. The technology uses this attribute to raise the use factor does a higher level.

REFERENCES

- [1] R. Bettati. End-to-End Scheduling to Meet Deadlines in Distributed Systems. PhD thesis, the University of Illinois at Urbana-Champaign, 1994.
- [2] A. Burchard, J. Liebeherr, Y. Oh, and S. H. Son. Assigning real-time tasks to homogeneous multiprocessor systems. *IEEE Transactions on Computers*, 44(12):1429-1442, December 1995.
- [3] A. Burns, K. Tindell, and A. Wellings. E@ective analysis for engineering real-time fixed priority schedulers. *IEEE Transactions on Software Engineering*, 21(5):475-480, May 1995.
- [4] M. Caccamo, G. Lipari, and G. Buttazzo. Sharing resources among periodic and aperiodic tasks with dynamic deadlines. In *IEEE Real-Time Systems Symposium*, December 1999.
- [5] M. Caccamo and L. Sha. Aperiodic servers with resource constraints. In *IEEE Real-Time Systems Symposium*, pages 161-170, 2001.
- [6] M. Chan and F. Chin. General schedulers for the pinwheel problem based on double-integer reduction. *IEEE Transactions on Computers*, 41(6):755-768, June 1992.
- [7] M. Y. Chan and F. Chin. Schedulers for larger classes of pinwheel instances. *Algorithmica*, 9:425-462, 1993.
- [8] D. Chen. Real-Time Data Management in the Distributed Environment. PhD thesis, The University of Texas at Austin, 1999.

- [9] Michael Frisbie, "A Unified Scheduling Model for Precise Computation Control," Master's Thesis, University of Kansas, June 2004.
- [10] Michael Frisbie, Douglas Niehaus, Venkita Subramonian, and Christopher Gill, "Group Scheduling in Systems Software," the 12th International Workshop on Parallel and Distributed Real-Time Systems (WPDRTS 2004) at IPDPS 2004, Santa Fe, New Mexico, April 26-27, 2004.
- [11] Christopher D. Gill, Flexible Scheduling in Middleware for Distributed Rate-Based Real-Time Applications, Ph. D. Dissertation, Department of Computer Science, Washington University, May 2002.
- [12] A. Gupta and D. Ferrari. Resource partitioning for multi-party real-time communication. Technical Report TR-94-061, Berkeley, CA, 1994.
- [13] P. Sen, G. M. Namata, M. Bilgic, L. Getoor, B. Gallagher, and T. Eliassi-Rad. Collective classification in network data. *AI Magazine*, 29(3):93-106, 2008.