

Збіраем цікавыя рашэнні

Картатэка — адзін з важнейшых інструментаў вынаходніка. Яе структура вызначаецца мэтай, для дасягнення якой яна прызначана. Гэта, напрыклад, можа быць даведнік або накапльльнік фактаў для наступнага аналізу і выяўлення заканамернасцей. Менавіта так выкарыстоўваюцца картатэкі распрацоўшчыкамі ТРВЗ.

Картатэка цікавых рашэнняў, якую мы пачынаем весці сёння, ставіць перад сабой задачу пазнаёміць чытачоў з некаторымі прынцыпамі рашэння задач.

Вымяральныя прыборы ўсе больш насычаюцца электронікай, яна бярэ на сябе многія сервісныя функцыі, дапамагае павысіць дакладнасць вымярэнняў, правесці статыстычную апрацоўку вынікаў. Але для работы электронных схем неабходны блок сілкавання, які значна павялічыць масу і габарыты прыбораў, расце спажыванне энергіі. У наяўнасці супярэчліvasць — блок сілкавання павінен быць большым, каб забяспечыць сілкаванне ўсіх схем. І павінен быць маленькім, каб не павялічваць масу і габарыты прыбора...

Выдзеліўшы супярэчліvasць, мы можам сфармуляваць ідэальны канчатковы вынік (ІКВ): электронныя схемы самі павінны забяспечыць малыя габарыты і вагу блока сілкавання, захаваўшы сілкаванне ўсіх вузлоў прыбора.

Сельскагаспадарчая машына для выраўноўвання глебы мае

актыўны рабочы орган — раскідвальнік глебы. Гэта некалькі лапатак, умацаваных на восі. Калі лапатка раскідвальніка, якая хутка круціцца, сустракаецца з каменем, узнікаюць моцныя ўдарныя нагрузкі, якія вядуць да паломак лапатак і моцнай вібрацыі машыны. У глыбіні гэтай задачы тоіцца супярэчліvasць: лапаткі павінны быць, каб раскідваць глебу, і іх не павінна быць, каб пры сустрэчы з каменем не стварыць вялікіх ударных нагрузак.

ІКВ — лапаткі раскідвальніка самі ўхіляюць вялікія ўдарныя нагрузкі, захоўваючы здольнасць раскідваць глебу.

Вечная праблема праграмістаў — супярэчліvasць паміж велічынёй праграмы і аб'ёмам памяці вылічальнай машыны. Праграма павінна быць малой, каб займаць невялікі аб'ём памяці. І павінна быць вялікай,

каб выконваць усе ўскладзеныя на яе функцыі.

Магчыма, вы ўжо сфармулявалі ІКВ для гэтай задачы: праграма сама забяспечвае выкананне ўсіх ускладзеных на яе функцый, займаючы пры гэтым невялікі аб'ём памяці.

Правільна пастаўленая вынаходніцкая задача абавязкова павінна ўтрымліваць супярэчнасць з ІКВ. Нярэдка задачы рашаюцца ўжо пры спробе выявіць супярэчнасць або на стадыі фармулявання ІКВ.

Прыведзеныя задачы аб'ядноўвае тое, што ўсё гэта — жорсткія, нязменныя, некіруемыя па зададзенаму параметру сістэмы: сілкаванне пастаянна падаецца на ўсе блокі прыбора, лапаткі моцна злучаны з валам, а размер праграмы не церпіць ніякіх змяненняў у працэсе яе выканання. Між тым, закон павышэння дынамічнасці і кіравання гаворыць пра тое, што развіццё тэхнічных сістэм адбываецца ад сістэм з пастаяннымі параметрамі да сістэм з параметрамі, якія змяняюцца пры змяненні рэжымаў работы сістэмы, для павышэння эфектыўнасці яе работы.

Выявіўшы тэхнічнае супярэччанне, сфармуляваўшы ІКВ і прачытаўшы фармулёўку закона, нескладана прапанаваць

рашэнні прыведзеных задач. Вядома ж, яны павінны стаць больш дынамічнымі:

1. Сілкаванне трэба падаваць толькі на тыя блокі, якія маюць патрэбу ў ім у дадзены момант — гэта так званае дынамічнае сілкаванне. Супярэччанне вырашылася — энергію атрымліваюць усе блокі прыбора, але па чарзе. Гэта дае магчымасць істотна змяніць блок сілкавання і знізіць энергаспажыванне.

2. Лапаткі раскідвальніка трэба мацаваць да вала пры дапамозе шарніра, тады ў момант сустрэчы з каменем яны будуць адхіляцца, гасячы энергію ўдараў, пасля чаго лапатка сама адносіць рабочы стан. Такім чынам, лапатка стала дынамічнай і самакіруемай.

3. У час выканання праграма павінна сама сябе мадыфікаваць, у залежнасці ад канкрэтных умоў задачы і этапу яе выканання апэратары павінны з'яўляцца, а стаўшы непатрэбнымі, знікаць. Такія магчымасці дае праграмісту, напрыклад, пралог (ад слоў лагічнае праграміраванне) — мова, прызначаная для распрацоўкі інтэлектуальных сістэм. Рэалізацыя гэтай ідэі дазволіла ўдвая скараціць размер праграмы.