**4. Стендові випробування для порівняння мікроконтролерів AVR з головними конкурентами**

Те, що застосування мови високого рівня С при складанні програм для AVR дає в результаті надзвичайно стислий код на асемблері (а значить, і машинний код), доводять також і стендові випробування базової серії мікроконтролерів AVR. При цьому результати зіставлялися з головними конкурентами (80С51, 68НС11, 80196 ...), а критерієм для оцінки був обсяг коду, згенерованого для виконання однієї і тієї ж задачі різними процесорами. Дані по порівняльним випробуванням мікроконтролерів сімейства AVR з поширеними на ринку 8 - і 16-розрядними мікроконтролера, надані компанією Atmel. Під час випробувань вирішувалися дев'ять поширених практичних завдань з різних областей застосування процесорів. Для отримання порівнянних результатів випробувань, програма для всіх процесорів була написана на мові високого рівня С.

Для трансляції коду майже у всіх випадках був застосований компілятор фірми 1AR Systems Ltd (на момент проведення випробувань компілятора цієї фірми не існувало тільки для процесора ARM7/ARM Thumb). Завдяки цьому, результати випробувань залежали не від можливостей оптимізації різних виготовлювачів компіляторів, а в значній мірі від архітектурних відмінностей процесорів. Тим часом вже з'явилися більш нові варіанти деяких компіляторів, внаслідок чого представлені тут результати могли б виглядати ще краще. Це відноситься також і до компілятору AVR-С, який є відносно "молодим" і з появою кожної нової версії помітно поліпшується в плані створення результуючого коду.

Само собою зрозуміло, стендові випробування цього виду можна назвати об'єктивними тільки умовно, і тому їх наочність обмежена, оскільки кожен процесор має свої особливі сильні і слабкі сторони.

Ефективність коду мікроконтролера сильно залежить від постановки задачі - немає ніяких універсальних "кращих мікроконтролерів" для всіх випадків застосування.

За допомогою стендових випробувань (як і з допомогою статистики) можна довести майже все, що завгодно, оскільки кожний мікроконтролер має свої "сильні сторони", і якщо його сконструювати для особливих випадків реалізації цих специфічних сильних сторін, то він може стати "ефективним з точки зору коду ".

Таким чином, на підставі цього зіставлення я не буду оголошувати мікроконтролери AVR найкращими з наявних на ринку. Мета даних випробувань - показати, що архітектура окремих представників сімейства AVR найвищою мірою оптимізована з точки зору обсягу програмного коду.

**5. Результати**

У семи з дев'яти тестів мікроконтролер АVR був серед трьох кращих і, як показує загальна оцінка, він різко відрізняється в кращу сторону від всіх протестованих контролерів.

На рис. 4.1 представлені результати окремих тестів, підсумовані і нормалізовані на основі результатів мікроконтролера AVR. Наприклад, для ARM7 обсяг коду вийшов в середньому в півтора рази більше, ніж для AVR.

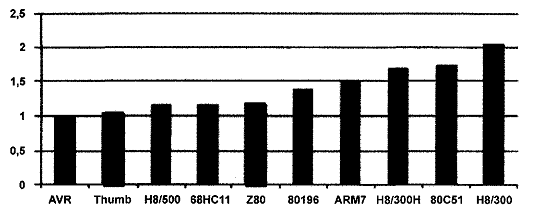


Рис.4.1 Сумарний результат

Провідним виробником З-компіляторів можна вважати компанію IAR Systems Ltd. За їх прекрасні "ноу-хау" в області компіляторів дійсно варто заплатити високу ціну. На ринку вже існують набагато більш дешеві С-компілятори SmallC AVR (природно, при обмеженій продуктивності), вартість яких нижче 50 доларів США.

Також для розв’язання задач діагностування можна використати мікроконтролер . Є два основних типи : PIC-контролери і мікроконтролери AVR. Але мікроконтролери AVR мають ряд переваг в порівнянні з PIC. Перш за все , мікроконтролери AVR мають досконалішу архітектуру і можуть виконувати команди в кожному такті ( відміну від PIC, яким для виконання команди потрібно чотири такти ). Тому при тій же тактовій частоті мікроконтролери AVR працюють в 4 рази швидше . Крім того , вони мають 32 робочих регістра ( відміну від одного єдиного ,наявного в PIC) і майже в 3 рази більше команд. Завдяки цьому програми для AVR практично завжди будуть коротші за аналогічні програми для PIC. Всі мікроконтролери AVR мають FLASH-пам’яті програм , що дозволяє здійснювати їх багатократне перепрограмування.

**Висновок**

Отже, серед численних технічних пристроїв особливе місце займають так звані мікроконтролери, які дуже поширені в наш час. Мікроконтролери можна зустріти у величезній кількості сучасних промислових та побутових приладів: верстатах, автомобілях, телефонах, телевізорах, холодильниках, пральних машинах ... і навіть кавоварках. Основною особливістю сучасного етапу розвитку мікропроцесорних систем (МПС) є завершення переходу від систем, виконаних на основі декількох великих ІС, до однокристальних мікроконтролерів (МК), які поєднують в одному кристалі всі основні елементи МШС: центральний процесор (ЦП), постійний запам'ятовуючий пристрій (ПЗП), оперативний запам'ятовуючий пристрій (ОЗП), порти вводу/виводи, таймери.

В даній роботі ми розглянули що таке мікроконтролер,структуру та їх класифікацію. Метою даної роботи є розглянути будову,принцип дії та класифікацію найпоширеніших мікроконтролерів PIC та AVR. Розглянути принцип роботи та різні методи програмування мікроконтролерів PIC та AVR, найбільш поширених у використанні. Порівняли швидкодію роботи різних типів мікроконтролерів, результати яких було побудовані у вигляді графіка. Було проведено дев’ять тестів, з яких було визначено трійку кращих мікроконтролерів. AVR мікроконтролер у семи з дев’яти тестів по результатам увійшов у трійку кращих. Але мікроконтролери AVR мають ряд переваг в порівнянні з PIC. Перш за все, мікроконтролери AVR мають досконалішу архітектуру і можуть виконувати команди в кожному такті ( відміну від PIC, яким для виконання команди потрібно чотири такти ). Тому при тій же тактовій частоті мікроконтролери AVR працюють в 4 рази швидше. Це показано на рис.4.2.

Роблячи висновок з даної роботи, можна сказати що розглянуті два типи мікроконтролери мають свої переваги та недоліки,але визначити кращого з них не можливо.

Вони використовуються для керування електронними пристроями. По суті, це — однокристальний комп'ютер, здатний виконувати прості завдання. Використання однієї мікросхеми значно знижує розміри, енергоспоживання і вартість пристроїв, побудованих на базі мікроконтролерів.