

Содержание:

1. Введение

Без информационных систем уже немыслима жизнь современного человека. Информационные технологии прочно проникли во все сферы деятельности людей. Для разработки и поддержания в работоспособном состоянии информационных систем требуются знания об их проектировании.

Знания в области проектирования информационных систем включают в себя знания системного анализа прикладной области, формализации решения прикладных задач и процессов информационных систем (ИС), разработку требований к созданию и развитию ИС и ее компонентов, разработку проектов автоматизации и информатизации прикладных процессов и создание ИС в прикладных областях, а также управление проектами информатизации предприятий и организаций.

Для визуализации разрабатываемой системы с разных точек зрения требуется использование диаграмм. Диаграмма в UML - это графическое представление набора элементов, изображаемое чаще всего в виде связанного графа с вершинами (сущностями) и ребрами (отношениями). Согласно теории диаграммы могут содержать любые комбинации сущностей, однако в практике моделирования применяется сравнительно небольшое количество типовых комбинаций, каждая из которых соответствует одному из пяти наиболее необходимых видов, составляющих архитектуру программной системы.

Диаграммы классов (class diagram), на которых показывают классы, интерфейсы, объекты и кооперации, а также их отношения. При моделировании объектно-ориентированных систем этот тип диаграмм использует наиболее часто. Диаграммы классов соответствуют статическому виду системы с точки зрения проектирования. Диаграммы классов, которые включают активные классы, соответствуют статическому виду системы с точки зрения процессов.

2. Схематичное изображение классов

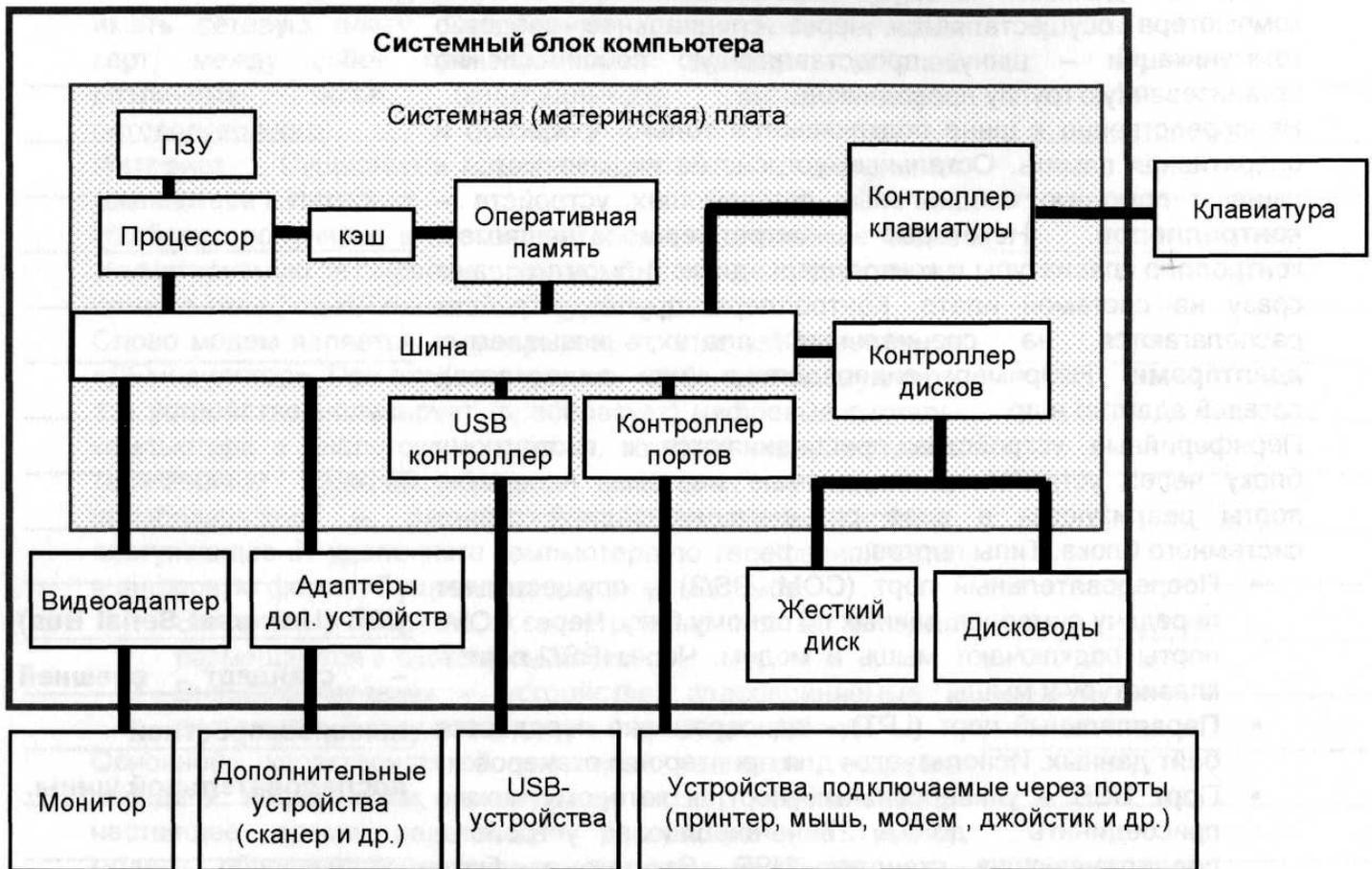


Рис.1 Схематичное изображение классов для аппаратной платформы персонального компьютера.

3. Анализ выбранных сущностей

Процессор

Процессор – это главная микросхема компьютера, его 'мозг'. Он выполняет программный код, находящийся в памяти и руководит работой всех устройств компьютера. Чем выше скорость работы процессора, тем выше быстродействие компьютера. Процессор имеет специальные ячейки, которые называются регистрами. Именно в регистры помещаются команды, которые выполняются процессором, а также данные, которыми оперируют команды. Работа процессора состоит в выборе из памяти в определенной последовательности команд и данных

для их последующего выполнения.

Во время процесса процессор считывает последовательность команд, содержащихся в памяти, и исполняет их. Такая последовательность команд называется программой и представляет алгоритм полезной работы процессора. Очередность считывания команд изменяется в случае, если процессор считывает команду перехода — тогда адрес следующей команды может оказаться другим. Другим примером изменения процесса может служить случай получения команды останова или переключение в режим обработки аппаратного прерывания.

Основными характеристиками процессоров являются: разрядность и быстродействие.

Быстродействие – это параметр, показывающий количество тактов, выполняемых процессором в секунду. Измеряется в мегагерцах (МГц), 1 МГц = 1 000 000 тактов в секунду. Чем данный параметр выше тем быстрее процессор.

Разрядность – это параметр который является важным для таких устройств компьютера, как внутренние регистры, шина ввода вывода данных, шина адреса памяти.

Процессор
+Марка компании: string
+Серия: string
+Разрядность: float
+Возможность разгона: bool
+Температура: float
+Гнездо: string
+Ввод данных()
+Расчет данных()
+Выгрузка данных()

Рис.2 Пиктограмма процессора

4. Внутренняя память компьютера

Оперативная память. Название «оперативная» эта память получила потому, что она работает очень быстро, так что процессору практически не приходится ждать при чтении данных из памяти или записи в память. Однако содержащиеся в ней данные сохраняются только пока компьютер включен.

Оперативная память
+Тип: string +Размер в битах: integer +Тактовая частота: integer +Температура: float
+Модифицирование данных()

Рис 3. Пиктограмма оперативной памяти

Постоянное запоминающее устройство (ПЗУ), в котором в частности хранится информация, необходимая для первоначальной загрузки компьютера в момент включения питания. Как очевидно из названия, информация в ПЗУ не зависит от состояния компьютера.

ПЗУ
+Энергоемкость: integer
+Выгрузка() +Загрузка()

Рис 4. Пиктограмма ПЗУ

Кэш-память. Основное назначение кэш-памяти в компьютере — служить местом временного хранения обрабатываемых в текущий момент времени кодов программ и данных. То есть ее назначение служить буфером между различными устройствами для хранения и обработки информации.

Кэш
+Тип: string +Размер: integer +Тактовая частота: float +Структура
+Векторизация() +Блокировка доступа() +Разблокировка доступа() +Считывание() +Выгрузка()

Рис.5 Пиктограмма Кэш

5. Внешняя память компьютера

Внешняя память обычно располагается вне центральной части компьютера

К внешней памяти относятся различные магнитные носители (ленты, диски), оптические диски. Внешняя память дешевле внутренней, но ее недостаток в том, что она работает медленнее устройств внутренней памяти.

Существуют диски CD-ROM - диски с однократной записью, стереть или перезаписать их невозможно.

Позже были изобретены перезаписываемые лазерные диски - CD-RW.

Внешняя память реализуется в виде довольно разнообразных устройств хранения информации и обычно конструктивно оформляется в виде самостоятельных блоков. Сюда, прежде всего, следует отнести накопители на гибких и жестких магнитных дисках (последние несколько жаргонно пользователи часто именуют винчестерами), а также оптические дисководы (устройства для работы с CD ROM).



Рис.6. Пиктограмма контроллера дисков

6. Устройства ввода-вывода

Устройства ввода и вывода - устройства взаимодействия компьютера с внешним миром: с пользователями или другими компьютерами. Устройства ввода позволяют вводить информацию в компьютер для дальнейшего хранения и обработки, а устройства вывода - получать информацию из компьютера.

Устройства ввода и вывода относятся к периферийным (дополнительным) устройствам.

К устройствам ввода относят клавиатуру, к пользовательскому выводу - монитор, это минимальная комплектация для ввода-вывода, все остальные устройства

относятся к дополнительным.



Рис 7. Пиктограмма контроллера клавиатуры, клавиатуры, видеоадаптера, монитора

7. Дополнительные устройства



Рис.8 Пиктограмма дополнительных устройств.

8. Проектирование выбранных сущностей

Необходимо создать объект, который бы позволил обмениваться данными между всеми остальными объектами, нечто вроде буфера и агрегатора, который бы инкапсулировал в себе компоненты аппаратной платформы. Такой объект – “Шина”, и описанные ранее сущности связываются между собой непосредственно через данный объект.

Далее установим парность каждого объекта:

- Процессор подключается к шине как один-к-одному.
- ПЗУ подключается к процессору как многие-к-одному.
- Кэш подключается к процессору как многие-к-одному с тем условием, что необходим хотя бы один объект кэша.
- К кэшу подключается оперативная память как многие-к-одному с тем условием, что необходим хотя бы один объект оперативной памяти.
- Все контроллеры подключаются к шине как один к одному с тем условием, контроллеров может и не быть в системе.
- Адаптеры подключаются к шине в неограниченном количестве.
- От таких классов-объектов как контроллеры – могут быть наследованы другие классы, или такие классы, которые могут взаимодействовать с ними, как например клавиатуры или монитор.

На рисунке 9 представлена полная UML-схема объектов аппаратной системы персонального компьютера:

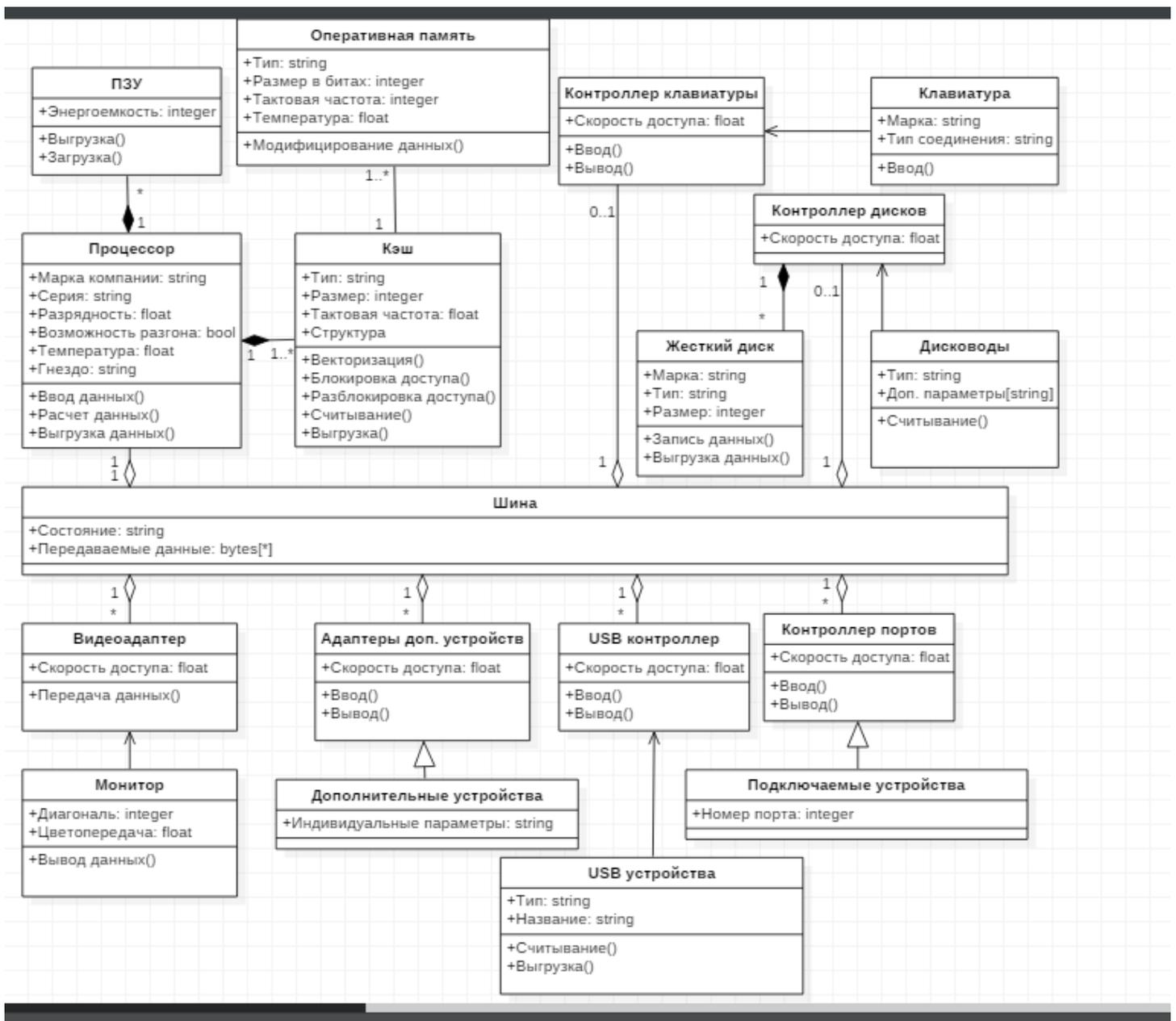


Рис 9. UML схема персонального компьютера.

9. Заключение

Основная цель объектно-ориентированного программирования, как и большинства других подходов к программированию – повышение эффективности разработки программ. Идеи объектно-ориентированного программирования оказались плодотворными и нашли применение не только в языках программирования, но и в других областях Computer Science, например, в области разработки операционных систем.

Основные идеи объектно-ориентированного подхода опираются на

следующие положения:

- программа представляет собой модель некоторого реального процесса, части реального мира.

- модель реального мира или его части может быть описана как совокупность взаимодействующих между собой объектов.

- объект описывается набором параметров, значения которых определяют состояние объекта, и набором операций (действий), которые может выполнять объект.

Благодаря средствам UML была реализована выбранная спроектированная объектная диаграмма классов аппаратного обеспечения персонального компьютера.

10. Список использованной литературы

1. Биллинг В.А., Дехтярь М.И. VBA и Office 97. Офисное программирование. М.: Русская редакция. 1998.
2. Буч Г. Объектно-ориентированный анализ и проектирование. М.: 1996.
3. Гайсарян С.С. Объектно-ориентированные технологии проектирования прикладных программных систем. М.: ЦИТ. 1998.
4. Гради Буч, Объектно-ориентированное проектирование. - Киев: "Диалектика" и М.: "И.В.К.", 1992.
5. Иванов А.Г. «Объектно-ориентированные системы: Состояние и перспективы», Москва 1992 г.
6. Мизрохи. Turbo Pascal и объектно-ориентированное программирование. — М.: Финансы и статистика, 1992.
7. Патрикеев Ю.Н. «Объектно-ориентированный анализ программирование» - Московский государственный университет экономики, статистики и информации, 2002
8. Салли Шлеер, Стефан Меллор. Объектно-ориентированный анализ: моделирование мира в состояниях. - Киев: "Диалектика", 1993.