

Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Гуманитарный институт

Кафедра культурологии и искусствоведения

УТВЕРЖДАЮ

Заведующий кафедрой

\_\_\_\_\_ Н.П.Копцева

«\_\_\_» \_\_\_\_\_ 2020 г.

## **БАКАЛАВРСКАЯ РАБОТА**

по направлению 51.03.01 Культурология

**СТАНОВЛЕНИЕ ПЕРСОНАЛЬНОГО КОМПЬЮТЕРА КАК АКТОРА  
В США 1970-ЫХ ГОДОВ НА ПРИМЕРЕ МОДЕЛИ APPLE II**

Руководитель \_\_\_\_\_ канд. филос. наук А.В. Кистова

Выпускник \_\_\_\_\_ Т.К. Ермаков

Консультант \_\_\_\_\_ К.В. Резникова

Красноярск 2020

Продолжение титульного листа бакалаврской работы по теме «Становление персонального компьютера как актора в США 1970-ых годов на примере модели Apple II».

Нормконтролер

*Релс-*

Д.С. Пчелкина

## РЕФЕРАТ

Бакалаврская работа по теме «Становление персонального компьютера как актора в США 1970-ых годов (на примере модели Apple II)» содержит 117 страниц текстового документа и 81 использованный источник.

Цель исследования состоит в выявлении особенностей становления персонального компьютера модели Apple II как актора в США 1970-ых годов.

Данная цель предполагает следующие исследовательские задачи:

1. Рассмотреть основные подходы к изучению техники в первой половине XX века.
2. Рассмотреть основные подходы к изучению техники во второй половине XX века.
3. Выявить основные черты Акторно-сетевой теории Бруно Латтура как метода исследования техники.
4. Описать персональный компьютер Apple II как актер в аппаратной плоскости.
5. Описать персональный компьютер Apple II как актер в программной плоскости.
6. Описать персональный компьютер Apple II как актер в социальной плоскости.

В результате проделанного исследования были рассмотрены основные подходы к исследованиям техники на протяжении XX века, выявлены основные черты акторно-сетевой теории как методологической основы для исследования техники, а также проведено исследование персонального компьютера Apple II как актора, на основании которого были сделаны выводы о становлении первых персональных компьютеров как акторов.

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	6
1 ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ИССЛЕДОВАНИЯ ТЕХНИКИ .....	11
1.1 Исследования техники в первой половине XX века: основные философские подходы.....	11
1.1.1 Первые исследования философии техники.....	12
1.1.2 Спекулятивно-умозрительные подходы к философии техники.....	19
1.1.3 Социально-гуманитарный подход в философии техники.....	24
1.1.4 Гуманитарно-антропологический подход в философии техники ...	26
1.1.5 Особые подходы к философии техники .....	30
1.2 Исследования техники во второй половине XX века: научные модели в исследованиях техники .....	32
1.2.1 Линейные модели развития техники .....	32
1.2.2 Интерактивные модели развития техники .....	37
1.2.3 STS модели в исследованиях техники (Social Construction of Technology) .....	40
1.2.4 SI модели в исследованиях техники (Multi-Level Perspective).....	43
1.3 Акторно-сетевая теория как база для исследования техники .....	46
1.3.1 Теория ассамбляжей М. Деланда и спекулятивный реализм как философская основа ANT .....	46
1.3.2 Основные положения акторно-сетевой теории Б. Латура.....	51
1.3.3 Возможности применения ANT для исследования техники.....	58
2 СТАНОВЛЕНИЕ ПЕРСОНАЛЬНОГО КОМПЬЮТЕРА МОДЕЛИ APPLE II КАК АКТОРА.....	63
2.1 Хардварная составляющая компьютера Apple II.....	64
2.1.1 Apple II как целое.....	65
2.1.2 Процессор компьютера Apple II.....	68
2.1.3 Память компьютера Apple II.....	70
2.1.4 Видеоконтроллер компьютера Apple II.....	72
2.1.5 Устройства ввода компьютера Apple II.....	74
2.2 Программная составляющая компьютера Apple II.....	76



2.2.1 Встроенное ПО компьютера Apple II и базовые операции с ним.....	77
2.2.2 Сторонние программы для компьютера Apple II. ....	81
2.2.3 Практики программирования на компьютере Apple II.....	94
2.3 Социальная составляющая компьютера Apple II .....	101
2.3.1 Компьютерные журналы и компьютер Apple II .....	102
2.3.2 Сообщества, формирующиеся вокруг компьютера Apple II.....	106
ЗАКЛЮЧЕНИЕ .....	108
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	111

## ВВЕДЕНИЕ

### **Актуальность темы исследования**

Акторно-сетевая теория является весьма популярной за рубежом концепцией, используемой для решения широкого спектра задач: от узконаправленных исследований техники и науки до построения обширных философских моделей. При этом в отечественной науке концепция всё ещё является относительно мало разработанной и используемой при её огромном исследовательском потенциале, ярко проявленном как в работах Бруно Латура отца-основателя концепции, так и его учеников и последователей.

Одной из областей, активно использующей ANT, является история и теория техники – направление исследований, место которого в современном научном знании определить довольно сложно. Являясь междисциплинарной областью, история и теория техники требуют от исследователя, как гуманитарной подготовки, так и понимания инженерно-технической составляющей исследуемого объекта. Из-за этого большинство исследований склоняются или исключительно в сторону технической эволюции, полностью игнорируя социально-исторический контекст, или наоборот полностью сосредотачиваются на «гуманитарной» составляющей, при этом избегая исследований сложных устройств, понимание работы которых требует специальной подготовки.

Именно такая судьба и постигла историю цифровой техники. «Гуманитарное» направление практически не прикасается к ней, в связи со сложностью технической основы, а исследования, проводимые инженерами, направлены на построение простой эволюционной линии развития самих устройств без учёта социально-исторического контекста. С учётом того, что цифровая техника вообще, и компьютеры в частности, являются неотъемлемым элементом жизни современного человека – потребность в устранении этого пробела становится всё более и более острой.

Но осмысление процессов, происходящих сейчас, невозможно без понимания истоков, без множества исторических исследований, направленных на обнаружение базовых характеристик тех или иных технических устройств и механизмов. Более того, подобные «исторические» исследования важны и в общем контексте изучения культуры второй половины XX века, поскольку уже тогда цифровые устройства начинали оказывать серьёзное влияние на повседневную жизнь человека, способствовали формированию новых практик и направлений искусства.

Таким образом, актуальность работы обусловлена именно необходимостью формировать историю цифровой техники, которая станет базой для дальнейших исследований «цифровой культуры», «цифрового искусства» и так далее. Помимо этого, использование АНТ в качестве основной теоретической модели позволит выявить некоторые универсальные характеристики исследуемого объекта, которые позволят чётко определить его возможные отличия от более «традиционных» технических устройств.

### **Степень изученности**

Сама по себе история техники – это дисциплина, оформляющаяся примерно к середине XX века на стыке инженерных и исторических наук, занятая сбором и изучением фактов, касающихся развития инженерного дела и способов производства в человеческом обществе. В рамках истории техники выделяется и история вычислительной техники, как отдельный раздел, занимающийся изучением различных ЭВМ.

Среди работ, посвящённых вычислительной технике, можно выделить несколько групп. Первая группа представляет собой работы, лишь затрагивающие исторический аспект в связи с определёнными узкими вопросами. Так, разделы, посвящённые истории вычислительной техники, содержат работы Танненбаума [37], посвящённые архитектуре компьютера, или работы группы авторов под руководством Альфреда В. Ахо [49], посвящённые построению компиляторов. Для этих работ характерно обращение к конкретным узким вопросам и акцент на эволюции аппаратной



или программной составляющей без обращения к общеисторическому контексту.

Во вторую группу можно выделить различные работы, посвящённые построению общей модели исторического развития ЭВМ. Это и различные учебные пособия, написанные для студентов-информатиков, и отдельные монографии Апокина и Майстрова [2], Иванова [14] и Малиновского [26] в отечественной науке. И работы Р.А. Ална [50], М.С. Махоня [76] в зарубежной практике.

Для этих работ характерно обращение к широкому контексту и попытка создания некой единой, общей модели истории ЭВМ. При этом авторы жертвуют деталями, касающимися отдельных машин, и не всегда отражают реальное многообразие устройств, существовавших в тот или иной период.

Третья группа исследований больше характерна для зарубежной практики – это различные по объёму работы, посвящённые отдельным компьютерам, существовавшим в то или иное время или даже отдельным элементам архитектуры компьютера. Большинство этих статей, тем не менее, сводится именно к сбору и компиляции фактов, касающихся выбранной проблемы без попытки объяснить возникающие связи или вписать их в какую-либо существующую модель развития техники.

Отдельно стоит отметить работу Яна ван ден Енде и Рене Кемпа 1998 года [61], которая до сих пор остаётся единственным примером использования одной из теорий развития техники (в данном случае – Multy-Level Perspective) к изучению истории ЭВМ. Данная статья показывает, что первые ЭВМ возникают не на пустом месте, а оказываются приуроченными к уже существовавшим практикам «вычисления», а дальнейшая история компьютеров, согласно авторам статьи также связана с решением проблем, связанных с потребностью в сложных вычислениях, постепенно возникающей в разных областях человеческой жизни.

Таким образом, несмотря на довольно большой объём литературы, посвящённой истории цифровой техники, преобладающее её большинство не пытается осмыслить происходящие процессы, вписав их в более широкий контекст или постаравшись выявить внутреннюю логику развития устройств.

### **Объект исследования**

Объектом исследования является персональный компьютер Apple II в США 1970-ых годов.

### **Предмет исследования**

Предметом исследования является становление персонального компьютера Apple II как актора в США 1970-ых годов.

### **Цель исследования**

Выявление особенностей становления персонального компьютера модели Apple II как актора в США 1970-ых годов.

### **Задачи**

1. Рассмотреть основные подходы к изучению техники в первой половине XX века.
2. Рассмотреть основные подходы к изучению техники во второй половине XX века.
3. Выявить основные черты Акторно-сетевой теории Бруно Латура как метода исследования техники.
4. Описать персональный компьютер Apple II как актер в аппаратной плоскости.
5. Описать персональный компьютер Apple II как актер в программной плоскости.
6. Описать персональный компьютер Apple II как актер в социальной плоскости.

### **Методы**

Ключевым методом исследования является метод акторно- сетевого анализа.

### **Научная новизна бакалаврской работы**

В научной практике не встречаются работы, посвящённые исследованию персонального компьютера Apple II как актора методом акторно-сетевого анализа.

### **Практическая значимость бакалаврской работы**

Результаты данной работы могут быть использованы как основа для дальнейших работ по изучению цифровой техники с точки зрения акторно-сетевой теории. На их основе можно проводить сравнения со спецификой развития персональных компьютеров в других странах и в другие временные отрезки. Также исследование персонального компьютера может лечь в основу исследований различных практик, формирующихся вокруг него (например, компьютерных игр или демосцены).

### **Структура бакалаврской работы**

Бакалаврская работа включает в себя введение, две главы основной части (6 параграфов), заключение и список литературы, включающий в себя 81 источник. Общий объём работы 117 страниц.

# 1 ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ИССЛЕДОВАНИЯ ТЕХНИКИ

## 1.1 Исследования техники в первой половине XX века: основные философские подходы

Традиционно исследования техники складывались в рамках двух дисциплин: это история техники и философия техники. Первая является исключительно исторической дисциплиной, сосредоточенной на выявлении фактов, описывающих общую динамику исторического процесса в развитии техники и технологии.

Отечественная традиция истории техники тесно связана с моделью исторического развития, предложенной Марксом. В её рамках техника вообще понимается как набор инструментов, связанных с производственными возможностями, а история техники – относительно самостоятельный раздел истории, занимающийся изучением развития производительных сил [12,25].

Философия техники, в свою очередь, оформляется в последней четверти XIX века (годом её создания можно считать 1877 год) и направлена на осмысление места техники в человеческом обществе и мире вообще. Она является одной из самых молодых философских дисциплин, хотя отдельные философы более ранних периодов обращались к осмыслению техники как метафоры других процессов (Беркли, Бойль, Ньютон) или обращались к отдельным аспектам существования техники в мире людей (Юр, Бекманн). Тем не менее, как отдельная область знания, нацеленная на изучение техники и осмысления её места в человеческом обществе и мире вообще, философия техники оформляется в работах Эрнста Каппа [1, 6, 7, 29].

Формирование философии техники связано как с развитием археологии, повлекшим за собой реактуализацию проблем, связанных с выделением человека из мира животных, так и с развитием инженерии [6].

Периодизация философии техники на данный момент разработана весьма условно, что связано с тем, что возникнув на рубеже XIX – XX веков, данное направление затем долгое время находилось на периферии философского дискурса. Вновь данное направление обретает вес лишь в конце XX века в связи с активным внедрением технического во многие сферы жизни человека и изменением самого существа техники.

Тем не менее, мы рассмотрим ряд концепций технофилософии, основываясь на классификации, предложенной Аль-Ани [1], согласно которой выделяются работы раннего этапа, концепции «спекулятивно-умозрительного», «социально-гуманитарного» и «гуманитарно-антропологического» характера. Также Аль-Ани обращает внимание на ряд примечательных концепций, не подпадающих под его классификацию.

Обзор технофилософских концепций первой половины XX века важен для понимания возникающих затем подходов к изучению техники, поскольку традиция исследования техники выростала из сообществ, объединявшихся вокруг тех или иных философских моделей.

### *1.1.1 Первые исследования философии техники*

Начнём наш обзор с творчества ряда исследователей, заложивших основы философии техники. Их концепции отличает необходимость обоснования существования философии техники как отдельной дисциплины, обращение к определению техники и особый упор на создание модели исторического развития техники, что связано с необходимостью объяснить факты уже накопленные к тому времени археологией (ранние орудия) и историей техники (более сложные устройства и машины, начиная с периода античности).

Эрнст Капп (1808 – 1896 гг.) – немецкий философ. Первоначально увлекается работами Гегеля, но довольно быстро отходит от идеализма и начинает активно использовать в своих работах материалистические концепции. Так, в работе «Общая и сравнительная география» (1845 г.) –



отрицает понимание исторического процесса как саморазвёртывание мирового духа, объясняя историю с точки зрения взаимодействия человека и природы [29].

В последствие, подпадает под влияние антропологического материализма Фейербаха, чьим последователем и остаётся до конца жизни. Капп делает основой своей философии антропологический критерий, который помещает человека в центр мира, понимая его как цель и причину всего существующего. При этом человек Каппа – это существо материальное, его телесная составляющая крайне важна (именно на основе этого, Капп переворачивает тезис Декарта – «человек мыслит, ибо он существует»).

Для понимания психической природы человека, это означает поворот в сторону физиологии. Психические процессы, как и вся мыслительная деятельность человека, понимаются как подчиненные его телесной составляющей и обусловленные ей.

Окружающий же человека мир – это результат взаимного творчества человека и природы. Человек оказывается неудовлетворённым окружающим его миром, поэтому он начинает преобразовывать его, но эти изменения он производит в соответствии со своей натурой. Он выносит своё внутреннее содержание во вне, сливая его с природной средой и изменяя тем самым последнюю.

Именно эта концепция и ложится в основу философии техники Каппа. Основные положения данного учения были изложены в книге «Основные направления философии техники» (1877 г.). На русский язык переведены отдельные статьи мыслителя (включая и некоторые главы из данной работы) – [16, 17, 18]. Ключевой же принцип подхода Каппа получил название «органопроекции».

Суть органопроекции состоит в том, что создавая орудия труда в процессе выражения своего внутреннего содержания вовне, человек создаёт их соответствующими его собственным органам. Таким образом, техника

становится естественным продолжением человеческого тела, устроенным по тем же законам, что и сам человеческий организм.

Для этого процесса характерно то, что он протекает непрерывно, и связан с бессознательным самообнаружением себя во вне. Так, перед тем как начать использовать палку, человек воспринял её как продолжение руки. Во-вторых, эта связь носит необходимый характер, иными словами без её возникновения техническое творчество невозможно в принципе. На этапе осознания этой необходимости человек переходит к сознательным отношениям с техническим устройством, в противовес первоначальным бессознательным отношениям. Наконец, органопроекция – это активное взаимодействие между человеческим и природным, в котором каждый из участников важен.

При этом органопроекция Каппа распространяется не только непосредственно на техническое творчество. Важным её проявлением философ считает и создание различных измерительных систем, опирающихся на человеческое тело. В первую очередь – это различные меры расстояния.

Органопроекция и вообще философия техники Каппа стала мощным толчком для дальнейшего развития в данной области. В целом, стоит отметить, что принцип органопроекции не ушёл из понимания техники окончательно. Мы можем наблюдать возвращение к нему в различных работах, связанных с кибернетикой и искусственным интеллектом, в которых делается попытка построения машин на основе принципов человеческого мышления.

Людвиг Нуаре(1829 – 1889 г.) может быть назван первым философом техники, поднявшим вопрос о влиянии последней на человеческое общество. Интересно, что его основные работы выходят практически одновременно с работами Каппа – «Происхождение языка» в 1877 г. и «Орудие и его значение в историческом развитии человечества» в 1880 г [30]. При этом



Нуаре предлагает, в каком-то смысле, противоположный подход к пониманию техники.

Если для Каппа в центре стоит человек, создающий орудие, то Нуаре рассматривает вопрос влияния орудия на человека и даже больше, влияние создания орудий как такового. Именно Нуаре создаёт теорию о трудовой эволюции человека, в которой ключевым звеном становится переход последнего к созданию искусственных орудий труда.

Нуаре выделяет гносеологическое значение технических произведений человека. Орудия двояко связаны с развитием сознания – с одной стороны, их создание возможно лишь при определённой степени разумности, с другой – создавая вещи человек развивает свои умственные способности. При этом каждое техническое изобретение и каждый отдельный предмет становятся самостоятельными элементами мира, из которых человек также может получать знание, изучая способы взаимодействия с окружающим миром.

Вообще, орудия труда в концепции Нуаре тесно связаны с расширением возможностей человека по взаимодействию с окружающим миром. Так, даже человеческая кисть переосмысливается человеком по ходу создания новых устройств – расширяется спектр её применения, в ней обнаруживаются такие свойства, которые до этого нельзя было в ней наблюдать.

Здесь важно отметить, что в концепции Нуаре также можно усмотреть черты органопроекции. Они связаны с тем, что орудия всё ещё создаются человеком в соответствии с его возможностями. Они не уподобляются человеческим органам как в концепции Каппа, но создаются удобными для человеческого использования, благодаря чему их внешний вид и способы применения оказываются зависимыми от человеческого тела.

Альфред Эпинас (1844 – 1922 гг.) – французский философ, чья концепция с одной стороны является естественным продолжением идей Каппа, с другой – привносит систематичность в философию техники и

сплетает данную дисциплину с исторической динамикой. Ключевая работа философа – «Происхождение технологии» 1890 года [46].

В исходной посылке создания философии техники Эспинасом можно проследить связь с концепцией Нуаре. Французский философ предпринимает попытку создания философии «действия», то есть осмысления сугубо практической сферы человеческого бытия, в противовес чистой духовной жизни, которой до этого занималась вся философская мысль.

В своём понимании техники, Эспинас полностью опирается на уже заявленный принцип органопроекции. Он приводит и примеры, связанные с проецированием человека вовне и восприятием инструмента как естественного продолжения руки, и обращается к ранним системам мер, также иллюстрирующим именно проявление принципа органопроекции.

При этом вводя историческую перспективу, Эспинас уже на довольно ранних этапах развития человеческого общества (а именно в Античности) начинает говорить о влиянии религиозных представлений на технику. В этом отношении философ подчёркивает слияние мифологического и технологического творчества, особое внимание обращает на религиозно-магическое складывание тех или иных технических изобретений. Этот характер отношений с техникой выражается как в стимулировании технического прогресса религиозными учреждениями (хрестоматийный пример – влияние храмов на развитие технологии чеканки монет), так и в мифах, посвящённых изобретению и использования новых технических устройств.

Продолжая исследования античной техники, Эспинас периодически осуществляет перенос технического творчества из сферы утилитарной в сферу эстетическую. Техника – это искусство, но искусство особого толка. Это прекрасные по своей сути вещи, созданные ради утилитарной функции, и подчиненные, прежде всего, ей. Тем не менее, их создание требует сложного комплекса действий, а поддерживаются они, как мы уже видели, из идеалистических областей жизни общества (мифологии и религии).

Также Эспинас вводит своеобразную классификацию направлений изучения техники, выделяя три области интереса данной дисциплины. Во-первых, описание «ремёсел, существующих в данный момент», за счёт чего возникает морфология техники, представители которой заняты исключительно описательной работой. Во-вторых, это исследования закономерностей становления технологий, то есть изучать динамику процессов технического творчества. Наконец, в-третьих, совокупность упомянутых подходов позволяет определять общие закономерности исторического развития техники, позволяющие установить характерные для этого глобального процесса пики и спады, отражающие интенсивность технического развития.

Фред Бон – немецкий философ, не занимавшийся отдельно исследованиями техники, но обратившийся к проблемам философии техники в своей работе «О долге и добре» 1898 года.

Техника понимается Боном во многом в традиции Нуаре, то есть важна в первую очередь как источник действия, в данном контексте – действия этического. Исходя из нравоучительного толка всей работы, естественен и ключевой вывод Бона, согласно которому техника является для человека одним из важнейших инструментов борьбы со злом и установления добра [1; 6].

Гораздо интереснее проводимое Боном чёткое отличие между наукой и техникой с одной стороны, и техникой и практикой с другой. В первом случае, отличие сводится к тому, что цель науки определяется как выработка чистого знания – понимание причинно-следственных связей, построение системы мироотношений. Перед наукой не стоит сугубо практических задач, которые уже решает техника, но опираясь на научные достижения, разработки и знания.

С другой стороны, техника отличается и от утилитарной практики. Отчасти это отличие проводится в контексте уже отмечавшегося этического вопроса и определения нравственной составляющей техники. С другой –

техника определяется, как сфера человеческой жизни, требующая активного творческого начала. Что и отличает её от простой утилитарной практической деятельности, сводящейся к воспроизведению уже известных схем.

Таким образом, Фред Бон интересен как философ, пожалуй, первым обозначившим проблему отношения техники и науки, к которой затем будут обращаться последующие философы.

Пётр Климентьевич Энгельмейер (1855 – 1942 гг.) – российский инженер и первый отечественный философ техники. Энгельмейер может быть назван последним представителем ранней философии техники и, вместе с этим, её вершиной.

Ключевая работа философа носит название «Философия техники». Она был опубликована в нескольких частях в период 1913 – 1914 годов [45]. Работа опирается на лекции Энгельмейера, прочитанные в 1912 году в Петербурге.

Начинает свой труд российский инженер с обширного литературного обзора, направленного в первую очередь на труды изобретателей различных времён. Предпринимается это историографическое исследование с целью определить сущность техники и её место в человеческом мире.

Исходными положениями концепции Энгельмейера являются представления о технике как средстве борьбы с природой. Из этого вытекают и сложные отношения между техникой и наукой, которые чётко отделены друг от друга с одной стороны, и взаимозависимы с другой.

Также техника, вслед за Нуаре, понимается философом как необходимая составляющая человека. В сущности, именно способность к техническому творчеству, а не разумность как таковая и определяет специфику человека как биологического вида. Энгельмейер определяет человека как «техническое животное», преобразовывающее природное окружение в соответствии со своими нуждами. При этом не порывает философ и с традицией органопроекции, которая в его учении заключена именно в этой биологической необходимости существования техники –

последняя целиком зависит именно от человека, формами существования которого она и ограничена.

Ещё одной важной характеристикой техники становится её утилитарность – все технические устройства нацелены на достижения пользы. Это отличает человека от чистого искусства, которое в понимании Энгельмейера сосредоточено исключительно на эстетических аспектах, не приносящих пользы. С позиций утилитаризма рассматривается и разделение науки и техники – первая заинтересована в выработке знаний ради знаний, не заботясь об их практическом применении, чем и занимается наука.

Наконец, важным пунктом учения Энгельмейера является концепция «моральной нейтральности» техники. Это учение представляется нам крайне важным для понимания специфической диалектики раннего этапа философии техники. Технические устройства необходимы человеку, являются его проекцией вовне, но при этом сами по себе не зависимы от человека.

Эта независимость максимально и постулируется в принципе «моральной нейтральности». Техника сама по себе не обладает никакими человеческими свойствами – она не принадлежит миру людей, хотя и происходит из него. Человек дополнительно включает её в свои собственные связи и категории, не просто создавая технические устройства, но и используя их.

Таким образом, можно сказать, что философия Петра Климентьевича Энгельмейера является логическим финалом в развитии ранней философии техники и представляет собой относительно законченную концепцию. После его работы мы уже можем говорить об окончательно сложившейся философской дисциплине, принципы которой претерпевают определённые изменения в трудах философов следующих периодов.

### *1.1.2 Спекулятивно-умозрительные подходы к философии техники*

Первая половина XX века была ознаменована целым рядом событий, серьёзно повлиявших на философию техники – это и

электрификация и механизация, начатые ещё во второй половине XIX века и достигших своего пика. Благодаря этим процессам техника начала активно вмешиваться в жизнь человека и изменять её. Из-за чего возникла необходимость в осмыслении не только исторической динамики в развитии технического, но и понимании специфики современной ситуации.

Помимо этого на восприятие техники серьёзное влияние оказали две мировые войны [7]. Их жестокость была во многом связана с техническим прогрессом, из-за чего возникла необходимость соотнесения проблем техники и вопросов морали. Проблема отношений категорий морали и различных устройств становится одной из центральных в технофилософии XX века.

По сути, все представленные в дальнейшем обзоре направления развивались параллельно друг другу. Тем не менее, «спекулятивно-умозрительный» подход может быть назван первым направлением, оформившимся в изучении техники в XX веке.

Фридрих Дессауэр (1881 – 1963 гг.) является первым философом техники, который относится к обозначенному направлению. Ключевыми работами автора являются «Техническая культура» (1908 г.), «Философия техники. Проблема реализации» (1927 г.), «Душа в сфере техники» (1945 г.) и «Споры вокруг техники» (1956 г.) [10, 11].

Концепция Дессауэра может быть названа «метафизической», поскольку автор, прежде всего, полностью отказывается от утилитарного понимания техники как инструмента улучшения жизни человека. Кардинально пересмотреть подход к технике Дессауэру позволяет его понимание техники как связующего звена между миром феноменальным и ноуменальным.

Здесь важно понимать, что философ под «техникой» понимает преимущественно техническое творчество. То есть именно в процессе изобретательства человек оказывается способным выйти за пределы ограниченного материального мира и проникнуть в мир идеальный.

Сам процесс технического творчества возможен благодаря трём составляющим: человеческому целеполаганию, природному материалу и внутренней обработке. Если до Дессауэра, по сути, доминирующим считался первый компонент – компонент целеполагания, решаемой задачи, то теперь он становится лишь своеобразным пусковым механизмом. Сама по себе утилитарная задача, вызвавшая к жизни изобретение, отходит на второй план, важен процесс сам по себе, ведь именно он позволяет человеку возвыситься над самим собой.

Природный материал подразумевает согласованность технического творчества с природными законами. За счёт этого элемента и создаётся своеобразная религиозность акта изобретения, ведь природа понимается Дессауэром как проявление божественных законов, а потому и работа над изобретением есть приобщение человека к божественному.

Однако природные законы накладывают и рамки, ограничивая творческий потенциал человеческого существа. Именно поэтому в творческий процесс и включается звено «внутренней обработки», позволяющее человеку расширить пределы своего познания, своего восприятия, сделать шаг в сторону бесконечного.

Таким образом, в философии Фридриха Дессауэра осуществляется метафизическое переосмысление философии техники. Акцент смещается с собственно вещи на процесс её создания, что позволяет рассматривать философию техники в контексте философии вообще, включая процесс технического творчества в отношения между идеальным и материальным, субъектом и объектом.

Стоит отметить, что Дессауэр является, пожалуй, единственным философом данного этапа, активно разрабатывавшим именно философию техники. Остальные представители умозрительно-спекулятивного подхода обращались к философии техники лишь как к одной из областей, зачастую подчиняя её своим основным задачам.

Эрнст Блох (1885 – 1977 гг.) является одним из самых ярких примеров подобного подхода. Будучи последователем Карла Маркса, Блох известен прежде всего за разработку учения об утопии, изложенном им в многотомном труде «Принцип надежды» 1954 – 1959 гг. [4, 5].

Для Блоха, техника – один из способов построения утопического общества, для которого характерен постоянный прогресс и стремление к совершенству, обеспечивающиеся во многом именно за счёт технического творчества.

При этом само техническое творчество – это в первую очередь раскрытие природного потенциала (а чём и заключается принципиальное отличие от учения Дессауэра). То есть человеческое начало в процессе изобретательства минимизируется, техника создаётся в первую очередь в соответствии с законами природы. Каждая вещь, созданная человеком, по факту заранее присутствует в окружающем его природном пространстве.

Таким образом, в философии Блоха мы видим максимальный уход от принципа органопроекции, характерного для ранней философии техники. По сути, мы наблюдаем диаметрально противоположную картину, в которой техническое творчество сливается с миром естественных, природных вещей, окружающих человека и неподвластных ему без дополнительных усилий.

Мартин Хайдеггер (1889 – 1976 гг.) как, пожалуй, один из крупнейших философов XX века, не мог обойти вопросы и проблемы философии техники. Его технофилософия может быть рассмотрена как завершающая стадия развития спекулятивно-умозрительных построений в области философии техники, и тесно связана с концепцией Дессауэра.

Связь с концепцией Дессауэра проявляется в соотнесении техники с категорией бытия, что, с другой стороны тесно связана с общей направленностью философии Хайдеггера. В концентрированной форме взгляды философа на технику представлены в докладе «К вопросу о технике» («Вопрос о технике») 1956 года [42].



Техника для немецкого философа – есть «один из способов раскрытия потаённости». То есть через технику человек может явить скрытое, раскрыть до этого не существовавшие в реальности аспекты объекта. Так, гидроэлектростанция раскрывает реку как источник энергии, которым она до этого в явной форме не являлась.

Сущность техники, определяющая специфику способа, которым она раскрывает потаённое, выражается Хайдегером в термине «по-став». Это такой способ раскрытия потаённости, который является «собирающим началом той установки, которая ставит, заставляет человека выводить действительное из его потаённости способом поставления его как состоящего-в-наличии». То есть раскрытие потаённости с помощью техники подразумевает извлечение из неё чего-либо. В примере с рекой и гидроэлектростанцией – поставление энергии.

Концепция «по-става» в философии техники Хайдегера связывает техническое с риском, риском увлечение одним способом раскрытия потаённого, из-за которого человек может оказаться слеп к ранее существовавшим способам раскрытия потаённого. Однако в этом риске и заключён спасительный росток, позволяющий человеку яснее увидеть саму опасность и спасение от неё.

Таким образом, можно сказать, что «спекулятивно-умозрительная» традиция философии техники тесно связана с осмыслением техники как общефилософской категории. В этом смысле, технику стремятся ввести в круг обще-философских проблем, связанных с онтологией и гносеологией, за счёт чего сама проблематика техники становится значительно шире. В этой традиции техника, как правило, понимается не просто как набор вещей и способов их производства, но как особый способ бытия, формирующий качественно иные способы отношений между субъектом и объектом.

### *1.1.3 Социально-гуманитарный подход в философии техники*

Другая сфера концепций, получивших название «социально-гуманитарного подхода», развивалась практически в одно время с концепциями «спекулятивно-умозрительного» толка. Традиционно, развитие этого подхода ведёт своё начало от работ Карла Маркса, который не занимаясь специально техникой, тем не менее, уже в своей работе «Капитал» (1867 год) поставил вопрос о существовании техники в социальной плоскости и её взаимодействии с социальной реальностью.

Жак Эллюль (1912 – 1994 гг.) – французский философ, который считается наиболее последовательным представителем «социально-гуманитарного» подхода в философии техники. Ключевые работы, посвящённые этому вопросу это «Техника» 1954 года и «Техническое общество» 1965 года [44]. При этом остальные работы автора, посвящённые скорее политической и социальной философии, также предлагают модели, опирающиеся именно на представления о технике.

Эллюль понимает технику максимально широко, по сути подводя под это определение всю сознательную деятельность человека по изменению окружающего его мира. При этом автор фиксирует некое качественное изменение в мире технического, позволяющее ему говорить о кардинальном отличии между техникой XX века и предшествующими техническими операциями.

Для новой техники Эллюль выделяет пять качественно отличающих её параметров: рациональность, артефактность, самонаправленность, саморост, универсальность, неделимость и автономность. Все они определяют современную технику как особый вид человеческой деятельности, отличный от того, что было до этого и влекущий за собой ряд качественных изменений в обществе.

Специфика технофилософии французского мыслителя заключается в своеобразном техническом детерминизме, с помощью которого Эллюль не только описывает современное ему общество, но и делает определённые

прогнозы. При этом он находится в рамках социально-философской модели Маркса, из-за чего его размышления отталкиваются от тех изменений, которые техника привнесла в производственный сектор.

Так, для современного «технического общества» характерно сокращение времени занятости людей на производстве и постепенное вытеснение их техникой. Также техника значительно увеличивает темпы производства, позволяя современному миру избавиться от тех экономических проблем, которые оказывались движущей силой всех социальных изменений в предшествующие эпохи.

Однако Эллюль подчёркивает и опасность, которую таит в себе техническое общество. Риск он видит в том, что развитие технике ведёт к потере видимых альтернатив, способных противостоять современным машинам в освоении природы. Техника приобретает всё большую значимость, из-за чего её кажущееся влияние на общество становится гораздо больше, чем оно есть на самом деле.

Подобное, «настороженное» отношение к технике XX века наиболее ярко проявила себя в работах представителей франкфуртской школы, в первую очередь в работах Герберта Маркузе (1898 – 1979 гг.), Макса Хоркхаймера (1895 – 1973 гг.), Теодора Адорно (1903 – 1969 гг.) и Юргена Хабермаса (1929 г.) [1].

Для работ Маркузе и Хоркхаймера характерно отношение к технике как к инструменту манипулирования и порождения массовой культуры. В этом отношении техника рассматривается как важный элемент современного общества, оказывающий влияние на формирование его негативных черт.

Более глубоко вопрос о технике рассмотрен у Адорно, который не просто обращается к роли технике в современном западном обществе, но и рассматривает технику как самостоятельную единицу, которая может быть взята и в отрыве от общества.

Благодаря этому, Адорно приходит к выводу об изначальной нейтральности техники. Сама по себе она не плоха и не обязательно связана с

манипулированием и обнищанием человека. Но она может быть использована в этих целях, и используется в современном западном обществе.

Также Адорно наиболее последовательно обращается к важному для «социально-гуманитарного» подхода вопросу о соотношении общества и техники. Связано оно с противоречием между техническим и общественным. Ведь техника создаётся как бы независимо от общественных интересов самыми разными людьми, из-за чего возникает необходимость в социальном ограничении технического творчества общественными интересами.

Но это же ограничение в результате и приводит к тому, что над техническим надстраиваются различные институты, в частности, позволяющие использовать технические изобретения для достижения негативных целей.

Результатом работы франкфуртской школы становится модель Хабермаса, который во многом опирается на концепцию Адорно. В частности, он также провозглашает изначальную нейтральность техники и её зависимость от того, как именно она будет использована людьми.

При этом Хабермас активно критикует именно ту модель использования техники, которую он наблюдает в современном ему обществе, в котором формируется своеобразная «техническая идеология», приводящая к уменьшению творческой свободы отдельных индивидов.

Таким образом, можно заключить для «социально-гуманитарного» подхода характерно рассмотрение техники как одного из элементов социального. При этом данные модели, как правило, оказываются инструментом критики современного общества, и реакцией на возросшее в начале XX века влияние техники на все сферы общественной жизни.

#### *1.1.4 Гуманитарно-антропологический подход в философии техники*

«Гуманитарно-антропологический» подход – условное название для третьей ветви развития философии техники в первой половине XX века. Для

этого подхода в целом характерно обращение к изучению философии техники в рамках философской антропологии – её влияние на существование конкретного индивидуума, без акцента на влияние техники на общество, как у представителей «социально-гуманитарного» подхода.

Карл Ясперс (1883 – 1969 гг.) – немецкий философ, один из представителей экзистенциализма. Непосредственно исследованию техники у Ясперса посвящена одна работа – «Современная техника», в которой прослеживается явное влияние Дессауэра [47].

Для Ясперса характерно отношение к технике как к инструменту, позволяющему человеку преодолеть собственную природу. В этом смысле особое внимание уделяется технике современной, как достигшей наибольших успехов в этом направлении по отношению к предшествующим этапам человеческой истории.

Рассматривая оппозицию природное и техническое, Ясперс вновь обращается к проблеме их соотношения. Согласно немецкому философу, техника преодолевает природу, но её суть остаётся согласованной с природным. Она зависит от устройства природного пространства и не может полностью оторваться от него. Это сближает технофилософию Ясперса с более ранними построениями Каппа и других представителей органопроекции.

Также обращается Ясперс и к проблематике «морали техники», не предлагая, тем не менее, в этом смысле ничего принципиально нового. Он, как и представители франкфуртской школы, постулирует изначальную нейтральность техники, говоря о том, что её использование зависит исключительно от человека.

Ключевое отличие от представителей «социально-гуманитарного» подхода состоит в том, что для Ясперса важен индивидуальный выбор каждого человека, а не решения социального механизма, как это было у Адорно или Хабермаса.

Люис Мамфорд (1895 – 1990 гг.) – один из наиболее ярких представителей философии техники, посвятивший данному



значительную часть своего творчества. Ключевые работы – «Техника и цивилизация» 1934 года и «Миф машины. Техника и развитие человечества» 1967 года [27, 28].

Мамфорд начинает исследования техники с решения вопроса взаимодействий между ней и человеком и роли технического в развитии человека. По сути, американский философ начинает серьёзным образом исследовать обратное отношение – влияние техники на человека (в то время как до него именно человек являлся единственным активным компонентом в паре человек-техника). При этом Мамфорд возвращается и к концепции техники как средства скорейшего достижения цели, но не выносит это её качество в качестве основного.

Техника, оказывая влияние на человека, тем не менее, остаётся продуктом его деятельности, причём деятельности естественной. Человек не может не создавать технические устройства, вся его деятельность пронизана своеобразной техничностью и материальные машины – это лишь один из результатов этой деятельности.

Рассматривая эволюцию техники Мамфорд обращается и к хорошо известной нам паре техника-природа. На основе изменений этих отношений он предлагает свою модель стадийного развития общества – эотехнический этап (гармоничное сосуществование природы и техники), палеотехнический (переход к паровым машинам – природа как источник ресурсов для техники) и неотехнический этап (современная техника, вновь пытающаяся восстановить гармоничные отношения с природным).

Развивая представления о технике как присущем человеку виду деятельности, который не может быть оторван от его существа, Мамфорд предлагает концепцию «мегамашины». По сути, это модель, предлагающая рассматривать общество как технический механизм – оригинальная функционалистическая теория общества. Важно понимать, что «мегамашина» Мамфорда – это не метафора, общество действительно является сложной машиной, что следует из самой сути человеческой





деятельности, сводящейся к постоянному воспроизводству рациональных технических устройств не только в качестве отдельных вещей, но и как идеальных структур.

Хосе Ортега-и-Гассет (1883 – 1955 гг.) также в своём творчестве обращался к вопросам философии техники и предложил в своей работе «Размышления о технике» 1939 года оригинальную модель, которая также может быть отнесена к «гуманитарно-антропологическому» подходу [31].

Ортега-и-Гассет рассматривает технику в ключе своего представления о потребностях. Философ выделяет естественные и сверхъестественные потребности. Первые проистекают из биологической природы и могут быть удовлетворены с помощью ресурсов самого организма. Для удовлетворения же вторых необходимо усилие, превышающее то, что изначально заложено в биологический организм.

Второй тип потребностей присущ исключительно человеку и именно для их удовлетворения и существует техника. Интересно, что концепция испанского философа не есть очередной вариант понимания техники как способа достижения целей. Сверхъестественные потребности Ортеги-и-Гассета не просто удовлетворяются с помощью техники, но и порождаются возможностью создавать технические изобретения, из-за чего они вступают в сложной взаимодействие с человеческим индивидом.

Таким образом, можно сказать, что «гуманитарно-антропологический» подход не только опирается на построение технофилософии исходя из места техники в жизни отдельного человека, но и стремится обосновать техническое как естественный элемент человеческого существа. Специфика человека как биологического вида определяется исходя из его возможности к техническому творчеству и потребности в создании технических устройств.

### *1.1.5 Особые подходы к философии техники*

Обратимся ещё к ряду технофилософских концепций начала века, не укладывающихся в рассмотренные ранее направления, но оказавших, тем не менее, определённое влияние на развитие представлений о технике.

Освальд Шпенглер (1880 – 1936 гг.) – один из влиятельных мыслителей XX века обращался к проблеме техники в своей работе «Человек и техника» 1931 года [43].

Концепция Шпенглера – максимальное проявление «биологического» подхода к пониманию техники, поскольку строится из посылки о том, что техника, понимаемая философом как целенаправленная деятельность вообще, присуща не только человеку, но и животным. В этом смысле – техника есть деятельность, направленная на выживание, поэтому Шпенглер зачастую подменяет это понятие термином «тактика».

Согласно этой концепции машины и инструменты есть результат технического действия, присущий уже исключительно человеку, так как последний обладает способностью производить их в соответствии со своим внутренним образом необходимого ему инструмента. Именно этим «технически» используемые человеком орудия отличаются от предметов, которые могут использовать некоторые животные.

Таким образом, концепция Шпенглера является ярким примером биологически-эволюционного осмысления техники именно как инструмента выживания биологического организма. Благодаря этому философ предлагает, пожалуй, один из самых гибких взглядов на техническое, не уходя в область чистых философских категорий, как представители «спекулятивно-умозрительного» подхода.

Николай Александрович Бердяев (1874 – 1948 гг.) также обращался к проблемам философии техники. В наиболее концентрированном виде его концепция изложена в статье «Человек и машина (проблема социологии и метафизики техники)», опубликованной в Париже в 1933 году [3].

Философ определяет технику как орудие достижения цели, расширяя область «технического», включая сюда не только вещи, но и различные практики, в том числе духовные и религиозные. Техника, по Бердяеву, это деятельность, направленная на достижение максимального результата при наименьших усилиях.

При этом философа интересует роль техники в современном обществе, взаимовлияние технических достижений начала XX века и человеческого существа. Для Бердяева ключевой проблемой в этих отношениях становится проблема религиозности человека.

Мыслитель постулирует изменение типа религиозности под влиянием технического. Это изменение, на его взгляд, будет связано с отрывом человека от Лона Земли, качественным изменением его мышления, приводящим к утрате ведущих позиций религиями, основанными на авторитетах.

Можно сказать, что Бердяев не предложил оригинальной технофилософской концепции, тем не менее, он вернулся к вопросу о религиозных аспектах технического, рассмотрев их не как одну из ранних стадий развития технического мышления, а как актуальный процесс, происходящий всегда.

Таким образом, в первой половине XX века было создано значительное количество технофилософских концепций самого разного толка, посвящённых решению следующих проблем:

1. Соотношение природного и технического
2. Соотношение социального и технического
3. Соотношение морального и технического
4. Определение техники
5. Обще-онтологический статус техники.

К тому же весьма распространёнными становятся представления о согласованности техники с природой самого человека, неотрывность её от

биологических характеристик Homo Sapiens и тесная связь с эволюцией последнего.

Также большинство исследователей понимают технику как способ «рационализации» и сокращения затрат энергии на совершение определённых действий. При этом интересно, что даже если эта рационализация не определяется как сущностная составляющая техники, она, тем не менее, оказывается необходимым следствием из технического прогресса.

Наконец, все концепции выделяют определённые стадии в развитии техники. В этой стадильности особое внимание уделяется технике XX века, которая оказывается качественно отличной от технических устройств предыдущих эпох, на основе чего делается вывод о соответствующих изменениях в человеческом обществе.

В результате, можно сказать, что технофилософия первой половины XX по сути сформировала круг проблем, которые необходимо решать в исследованиях техники и постулировала ряд положений, которые остались фундаментальными для ряда будущих концепций.

## **1.2 Исследования техники во второй половине XX века: научные модели в исследованиях техники**

В данном параграфе мы обратимся к научным моделям исследования техники, оформляющимся к 1950-ым годам и вытесняющим философию техники с позиций главного поставщика знаний касательно места и роли техники в человеческом обществе.

### *1.2.1 Линейные модели развития техники*

Подходы к исследованию техники, выделявшиеся из общего дискурса философского осмысления феномена, берут своё начало ещё в первой половине XX века и связаны по большей части с различными обществами

инженеров, возникавших во многих развитых странах и ставивших своей целью продвижение изобретений [6].

Подобные объединения существовали параллельно с философами техники, изредка пересекаясь (как, например, в Германии, где инженерные общества способствовали распространению работ Каппа), тем не менее, основной их задачей было не столько изучение феномена технического, сколько поддержание изобретательской деятельности, формирование особых социальных институтов, которые будут способствовать созданию и продвижению инноваций.

Именно проблема создания «потока инноваций» и привела к тому, что уже во время Первой Мировой войны многие государства, в том числе и США, приходят к необходимости проведения особых исследовательских работ, которые позволят понять механизмы, по которым функционирует социальная система создания инновационного. Возникает острая потребность в моделях, которые смогут отвечать на конкретные вопросы, связанные с функционированием не просто техники, но технического новшества в рамках человеческого общества [69].

Очевидно, что философские теории, заложившие базу для дальнейших исследований и сформировавшие, по большей части, понятийный аппарат будущих теорий были не способны решать конкретные управленческие задачи. Для этого требовалось на время отойти собственно от проблемы технического и обратиться к иным областям социальной действительности, задействованных в процессе создания инноваций. Во многом именно сменой социальных пластов, на основе которых строится модель инноваций, можно объяснить смену ведущих моделей исследования техники.

Первая модель, решавшая поставленные вопросы, была сформирована на рубеже 1940 – 1950-ых годов, преимущественно в США, на основе традиции исследования науки. Данная модель носит линейный характер и опирается на представления о взаимоотношениях между инженерией, прикладной и фундаментальной наукой.

Первоначально, в центре внимания оказалась именно проблема отношений между фундаментальной и прикладной наукой. Наиболее ярко она была поставлена в работе В. Буша 1945 года «Science: The Endless Frontier» [57]. В этой работе проводилось исследование отношений между фундаментальными исследованиями и техническим развитием общества вообще. Во многом, сама по себе проблема прикладной науки опускалась и на её место вставала непосредственно изобретательская деятельность.

Схема, созданная Бушем, сводилась к тому, что теоретические открытия затем воплощаются в прикладных разработках и новых технологиях. Уже на тот момент она воспринималась как несколько упрощённая, но была охотно принята, поскольку давала конкретный ответ на вопрос о дальнейшем развитии: необходимо поддерживать фундаментальную науку, за счёт чего и будет происходить дальнейшее развитие техники и технологии.

В то время как Буш предлагал сосредоточиться на проблемах, связанных с наукой, уже в 1920-ые годы сформировалось представление о важности изучения самого процесса изобретения и производства новых технологий. В данной области формировались более сложные модели, как, например, модель Холланда, описанная в 1928 году как «исследовательский цикл» [71]. Она включала в себя:

1. Фундаментальное научное исследование (открытие базовых принципов)
2. Прикладное научное исследование (следствия из уже открытых положений)
3. Изобретение (теоретическая модель нового технического устройства, опирающаяся на предшествующие открытия)
4. Индустриальное исследование (формирование представлений о возможности производства нового продукта)
5. Индустриальный эксперимент (первые практические опыты изготовления продукта)

6. Стандартизация (формирование стандартной производственной схемы)

7. Массовое производство (введение нового устройства в область массового потребления)

Данная модель также устанавливала простую причинно-следственную связь между отдельными элементами процесса производства новых устройств, но из-за включения в неё множества промежуточных ступеней оказалась слишком сложной для решения прикладных задач, связанных с менеджментом инновационных предприятий.

В результате, в 1940-ых годах в работах Бишовски [54] и Фурнаса [64] происходит уменьшение количества ступеней до трёх: исследование, изобретение, производство. Данная схема опиралась в большей степени именно на вопросы, связанные с производством новых продуктов, из-за чего существовала параллельно с моделью Буша, подходившей к вопросу со стороны проблем научного развития.

В результате к началу 1950-ых обе модели «синтезируются», из-за чего возникает наиболее принятая в данном подходе модель, состоящая из пяти ступеней [77]:

1. Фундаментальное исследование
2. Инженерия
3. Производство
4. Маркетинг
5. Продажи

Получившая название модели «научного прорыва» («sciencepush») данная модель отражает в себе смену интереса исследователей: если раньше инновации были интересны как проявление других проблем, то теперь исследователя интересуют уже сами новые продукты и, прежде всего, процесс их внедрения в массовый рынок.

Однако именно в силу этого интереса данная модель просуществовала недолго. Устанавливая простую причинно-следственную связь между

новыми технологиями и изначальными наработками учёных, она не отвечала на вопрос, который интересовал большинство – почему, те или иные инновации становятся популярными, приобретая статус массовых продуктов, а другие нет. Процесс производства инноваций оказывался естественным следствием из научного развития, что слабо вязалось с наблюдаемыми реалиями рынка.

Разрешить эту проблему была признана линейная модель, получившая название «рыночного спроса» («marketpull»), предложенная в середине 1960-ых, когда быстро развивающийся рынок технологий уже не устраивала «sciencepush» модель [69].

«Marketpull» включает в себя четыре ступени [72]:

1. Потребности рынка
2. Разработка
3. Производство
4. Продажи

По сути своей – перед нами чисто экономический подход, переворачивающий существовавшие до этого представления с ног на голову. Теперь сами по себе научные исследования полностью исключались из рассматриваемых связей, причиной возникновения новых устройств становится рынок.

Важно понимать, что данная модель не является предшественником «социального переворота», который будет происходить в 1990-ых годах. Здесь не социум выступает источником и причиной развития технологий и появления новых продуктов, а обезличенный «рынок», функционирующий прежде всего по экономическим законам.

«Market Pull» модель вытеснила «science push» и оставалась ведущей до начала 1970-ых годов. Тем не менее, игнорирование проблем, связанных с развитием науки и техники, оторванных от законов рынка, не могли удовлетворить большую часть учёных. Было это связано, скорее, с общим кризисом линейных моделей, который, по мнению Година [70], с самого



начала своего существования воспринимались как временная мера и простой способ дать быстрый и достаточно точный ответ на интересующие заказчиков вопросы.

На самом деле, простые казуальные связи между рынком и различными исследовательскими и инженерными институтами явно упрощали картину, ведь прикладные исследования показывали, что все элементы находятся в сложных отношениях взаимовлияния. Ответом на оформившиеся противоречия становится группа моделей, получивших название «интерактивных».

### *1.2.2 Интерактивные модели развития техники*

Интерактивные модели берут своё начало в 1970-ых годах, когда стало очевидным, что в процессе появления инноваций участвует множество акторов, чьи отношения не сводятся к прямым связям. Первая подобная модель опиралась на синтезе «market-pull» и «science-push» моделей.

По сути, использовалась схема с теми же ключевыми элементами, но её стартовой точкой теперь может быть как научное открытие, так и потребности рынка. Подобный поход сделал сам анализ причин появления инноваций более гибким, но не отвечал на другой вопрос, связанный с институционализацией процесса и необходимостью уметь инициировать появление нового технического устройства [70].

Для решения этой проблемы были выработаны принципиально иные интерактивные модели, отошедшие от попыток решения проблемы через анализ абстрактного процесса развития научных идей к готовым коммерческим продуктам. Теперь речь шла о социальных институтах, действующих на разных организационных уровнях, из-за чего внимание исследователей привлекал вопрос о том, какой из уровней является здесь главным – чья инициатива способна запустить процесс инновационной разработки на оставшихся уровнях. При этом сами социальные институты

вставляли на место абстрактных звеньев цепочки, выработанной в результате интерактивного подхода как такового.

Подобные исключительно управленческие модели возникают примерно в одно время на рубеже 1980-ых и 1990-ых годов. К тому времени уже набирают определённый вес научные теории, обращающие своё внимание на социальные причины развития техники, из-за чего данная группа теорий использовалась, прежде всего, в менеджменте предприятий и в политических теориях.

Так, модель NSI (National Systems of Innovation) [78, 79] обращала внимание преимущественно на уровень государственного устройства. Данная модель рассматривает особый набор элементов, представленных различными публичными институтами, оказывающими прямое воздействие на сам процесс разработки инноваций.

При этом специфика функционирования каждого элемента в данном случае зависит от национальной и государственной специфики. NSI – модель востребованная именно государственным аппаратом, поскольку все её прогнозы и методы регулирования опираются на представление о том, что государственная политика способна напрямую влиять на способ функционирования отдельных институтов, отвечающих за создание новых продуктов.

При этом NSI оказывалась слабо применимой для государств с ярко выраженными локальностями, создающими пёструю картину развития технологий, сильно вырывающихся вперёд в одних областях, и отстающих в других. Специально для таких случаев была разработана модель RSI (Regional Systems of Innovation) [48,52,58].

Что интересно, RSI не только переводит фокус с анализа глобального контекста в пользу локальностей, но и разрабатывает решения исходя из априорного положения о том, что отдельные игроки действуют в ситуации крайней неопределённости, когда не существует единой системы инновационного развития. В этом случае регионы опираются, прежде всего,

на свои собственные нормативные акты, а объединяют их исключительно самые базовые положения.

При этом RSI предпочитает работать с коалициями, а не отдельными объектами, из-за чего речь всегда идёт о некоторых группах заинтересованных лиц. Из этого следует и главная слабость подхода – инновационное развитие в его рамках понимается как единый, относительно равномерный процесс, то есть не учитывается отраслевая специфика, поскольку все организации представляются неким единым объектом региональной политики.

В результате возникли SSI (Sectoral Systems of Innovation) модели [56], которые напротив полностью сосредотачивались на отдельной отрасли как носителе инициативы, задающем правила игры для всех остальных институтов, включённых в процесс инновационного производства.

Принципиальным отличием SSI от остальных моделей становится не просто обращение к более мелким иерархическим единицам, но и внимание к неинституциональным явлениям в рамках отрасли. Так, помимо непосредственной организации как производства, так и изобретения новых устройств, внимание специалистов обращено теперь и к изобретательской культуре, отношениям внутри коллективов и т.д. Из-за этого SSI несмотря на уменьшение масштаба объекта включает в него большее число важных элементов, оказывающих непосредственное влияние на оставшихся участников инновационного процесса.

Вершиной развития интерактивных моделей становится предложенная в Швеции в 1990-ые годы модель TIS (Technological Innovation System) [62], задачей которой было разрешение противоречий, возникавших между предшествующими моделями. Основой для решения этих противоречий стало обращение к новому для исследований технике объекту – пользователю.

Для TIS пользователь ещё не играет ключевой роли, но через него в общую схему входит понятие «пользовательской практики», за счёт которой

возникает внимание к индивидуальному отношению между человеком и уже существующим устройством. Именно индивид, в результате, и становится главной особенностью TIS – предложившей понимать под причиной технического развития действия отдельных индивидов, помещённых в особые условия, регулируемые на самых разных уровнях.

Это очень важный вывод, который будет подхвачен последующими моделями. Теперь инновации разрабатываются не сложными организациями, состоящими из множества институтов, но человеком. Пожалуй, именно это внимание к человеку, возникающее в среде активно использующихся на практике моделей, помогло возвращению и сугубо научных теорий развития техники, произошедшему в те же 1990-ые годы.

### *1.2.3 STS модели в исследованиях техники (Social Construction of Technology)*

В результате возникает группа моделей, получивших общее наименование STS (Science and Technology studies). Для этого подхода характерно обращение в первую очередь к пользователям, которые как определяют финальный вид нового продукта, так и оказывают непосредственное влияние на процессы его производства. Первоначально, данные модели разрабатывались как исключительно научные, направленные на осмысление уже произошедших событий с целью понять сам механизм появления новых устройств, но с появлением TIS, они также начинают использоваться для прикладных исследований.

Наибольшую известность и вес приобрела модель SCOT (Social Construction of Technology), окончательно сформировавшаяся к 1987 году, когда вышел отдельный сборник статей, посвящённых как конкретно этому методу, так и общим принципам, лежащим в его основе [55].

Для SCOT характерно обращение именно к социальным факторам, влияющим на формирование инновационного устройства. Общество выступает здесь своеобразным заказчиком, под требования которого и

подстраиваются оставшиеся институты, занятые в непосредственном производстве готовой продукции. Особенностью данной модели является представление о самом обществе как о чём-то дифференцированном, состоящем из множества отдельных группировок, заинтересованных в том, чтобы конечный продукт выглядел определённым образом.

Являясь, прежде всего, методической моделью, направленной на изучение процесса становления технологических новшеств в качестве массового продукта, SCOT стал одной из первых моделей в исследованиях техники, обратившейся к анализу конкретных исторических случаев, вместо исследования актуальной действительности. Связано это было с уже упоминавшимся фундаментальным интересом исследователей. Для них было важно выявить базовые закономерности, определяющие всё техническое развитие общества.

В ходе исследования ряда конкретных случаев был выработан методологический алгоритм, отражающий и сами этапы возникновения инновации:

#### 1. Деконструкция объекта.

Этот этап подразумевает, что в обществе уже существуют аналоги того устройства, которое вот-вот появится, при этом существуют отдельные группы, по тем или иным причинам заинтересованные в дальнейшем совершенствовании этого артефакта.

Выделение этих релевантных групп – самый важный этап исследования, поскольку именно их интересы и будут в дальнейшем видоизменять уже существующие артефакты или создавать новые. Само по себе состояние общества в этот период напоминает точку бифуркации сложной системы, находящейся в хаотическом состоянии и стремящейся выбраться из него. При этом способов преодолеть это состояние всегда множество, и выбирается из этого множества тот, который окажется компромиссным для системы.

#### 2. Социальное конструирование артефакта.

Собственно этап социального обуславливания нового объекта. В ходе этого процесса и происходит выбор системой одного из возможных вариантов развития.

С точки зрения исследователя, на этапе описания социального конструирования важно не додумывать самому, а стремиться использовать те источники, в которых будут напрямую выражены позиции уже выделенных заинтересованных групп, способы их аргументации и причины, по которым те или иные артефакты были приняты или отвергнуты.

### 3. Вписывание в глобальный контекст.

Здесь ощущается влияние интерактивных моделей – представление о существовании различных масштабов взаимодействия между участниками процесса и взаимовлиянии отдельных процессов конструирования на аналогичные процессы на других уровнях.

Стоит отметить, что последний этап слабо освещён в самих работах представителей данного направления и, пожалуй, является самой очевидной слабостью метода. SCOT позволял получить описания процессов на определённых уровнях, как правило – понять, что происходит в относительно локальных социальных ячейках. При этом практически полностью игнорировались процессы, происходившие даже в несоциальном секторе или в более крупных социальных масштабах.

С другой стороны, SCOT остаётся крайне действенной методологической схемой, результаты применения которой ясно показали важность социума в формировании конечного облика инновационных продуктов. Но SCOT всё ещё не позволял описать картины целиком, обращаясь, по сути, к отдельной области, пусть и мало изученной до того времени. Для решения этой проблемы в скором времени появились синтетические модели, постаравшиеся объединить в себе как достижения прикладных моделей, обращавших больше внимания на собственно техническое развитие и функционирование специальных институтов, так и на

выводы STS, вернувших человека в круг проблем, решаемых в исследованиях техники.

#### *1.2.4 SI модели в исследованиях техники (Multi-Level Perspective)*

Синтетические модели, возникающие на стыке STS и интерактивных моделей, больше известны как SI (Systems Innovation). Их главной отличительной особенностью становится понимание самого процесса производства инноваций как сложного и многокомпонентного.

Внимание, прежде всего, уделяется социальной составляющей, но она неизменно вписывается в широкий контекст, включающий в себя институты, связанные с наукой, инженерией, производством, маркетингом, рынком и так далее. При этом вновь происходит смена ключевого интереса исследователей: теперь исследования сосредоточены на механизмах закрепления уже полученного продукта в обществе, что отличается от интереса к самому процессу производства вещи (линейные и интерактивные модели) или к процессу формирования «запроса» общества на инновацию (STS).

Пожалуй, одной из самых влиятельных моделей из данной группы становится MLP (Multi-Level Perspective), сформировавшаяся в начале 2000-ых годов в работах Джилса и Кемпа [65,73], хотя предпосылки для её формирования появляются уже в 1970-ых годах. В целом для данной модели характерно представление об иерархии разномасштабных элементов, отвечающих за создание и распространение технических продуктов в каждом конкретном обществе.

MLP работает с моделью, состоящей из трёх уровней, от самого маленького к самому большому: ниши, режимы и ландшафты. Чем масштабней уровень, тем более статичен он, и тем сложнее изменить его, именно поэтому все открытия происходят на уровне ниш.

При этом сами уровни неоднородны. Так, на уровне режимов выделяются 7 измерений, среди которых:

1. Технологии
2. Рынки
3. Символическое значение технологий
4. Инфраструктура
5. Индустриальная структура
6. Политика
7. Научные знания.

Выделяемые уровни тесно связаны друг с другом, поэтому изменения хотя бы в одном из них могут привести к изменению всего режима. Эти изменения в свою очередь могут привести как к серьёзным подвижкам в ландшафте, так и к прорыву отдельных «нишевых» технологий на уровень режима. При этом между отдельными уровнями сохраняется и определённая степень независимости: так, изменения в отдельных нишах могут никогда не войти в режим, а потому не окажут никакого влияния на процессы за пределами своего уровня.

Традиционно MLP применялся для исследования исторических процессов – смены одних технологий другими. Эти технологические переходы понимаются в рамках данной модели как часть крупных социальных трансформаций, происходящих на уровне ландшафта.

Также в MLP был выделен ряд «шаблонов», по которым обычно происходит выход инноваций с уровня ниш:

1. Технология начинает использоваться в сфере близкой к сфере первоначального применения.
2. Инновационные элементы добавляются к уже существующему техническому устройству.
3. Новые продукты начинают использоваться в сферах, переживающих мощный подъём.

Сами по себе режимы также могут возникать по нескольким сценариям:



1. Новый режим приходит на смену уже существующим одному или нескольким режимам.

2. Принцип «гнезда» - появление «подрежимов» внутри одного режима большего масштаба (старый режим в таком случае может стать новым ландшафтом).

3. Новый режим возникает на пересечении уже существующих режимов.

Наконец, в рамках MLP были описаны несколько вариантов изменения всей системы в целом:

1. Смена ведущего режима без появления новых технологий (например, если этот режим уже существовал, но по ряду причин не был особо востребован).

2. Накопление противоречий в рамках существующего режима, которые решаются с появлением нового режима.

3. Успешно существующий режим создаёт новые функции, которые способствуют его дальнейшему расширению.

4. Глобальные изменения, происходящие сразу на всех уровнях.

Несмотря на то, что MLP на данный момент остаётся одной из самых проработанных моделей собственно исследований техники, она не является идеальной и требует

дальнейшего совершенствования.

Ключевыми

недостатками модели являются:

1. Отсутствие масштабного «образцового» исследования, в котором была бы полностью прослежена логика изменения технологического ландшафта, а не отдельных ниш и режимов.

2. Неточность в понятии «технологического перехода». Под этим термином понимаются изменения на уровне режимов, при этом это не точечный переворот, а длительный процесс. Именно в длительности и заключена основная проблема – крайне сложно выделить конкретный момент во времени, когда процесс начался и когда он закончился.



3. MLP перенесла в себе одну из ключевых проблем линейных моделей – все переходы в рамках модели рассматриваются как предопределённые, поскольку исследователю уже известно, какие именно изменения произойдут и какая технология окажется лидирующей.

4. MLP не отвечает на вопрос о возможности контроля процесса перехода, из-за чего не может быть использована в прикладных целях. В то же время именно этот аспект вновь становится всё более и более важным.

Тем не менее, MLP – это модель, которая ввела в оборот такие понятия, как технологические ниши, режимы и ландшафты, подхваченные множеством исследователей, не всегда напрямую связанных с самой этой моделью.

### **1.3 Акторно-сетевая теория как база для исследования техники**

В данном параграфе мы обратимся к акторно-сетевой теории Бруно Латура и поддерживающим её философским моделям, которые и станут основой практической части работы.

#### *1.3.1 Теория ассамбляжей М. Деланды и спекулятивный реализм как философская основа ANT*

Несмотря на то, что в исторической перспективе ANT (Actor-network theory) возникает раньше философских разработок и Деланды, и представителей спекулятивного реализма, и это скорее она оказывала влияние на объектно-ориентированную онтологию, а не наоборот (а в случае с Деландой две теории вообще развивались практически параллельно) [32, 41], в силу, прежде всего, методологической ориентации ANT имеет смысл начать её рассмотрение с философских моделей, углубляющих и своеобразным образом теоретизирующих основные принципы теории.

Теория ассамбляжей, предложенная Мануэлем Деландой, формируется в работах философа в начале 2000-ых годов на основе анализа работ Жюль

Делёза и Феликса Гваттари [8, 9]. По мнению самого философа, теория ассамбляжей уже была представлена в работах двух постмодернистов, но была рассредоточена по множеству отдельных работ, его же задачей стало собрать её в единое целое и несколько усовершенствовать. Развивая концепцию ассамбляжей, Деланда создаёт особое онтологическое учение – концепцию плоских онтологий, которую применяет для анализов природных и человеческих сообществ, демонстрируя её возможности.

Главной задачей плоской онтологии является решение проблемы редукционизма, неизбежно возникающего при анализе сложных иерархических систем. Также в рамках данной концепции весьма изящно решается вопрос о возможности объединения элементов в системы, являющийся одним из ключевых в философии, начиная с немецкой классической философии.

Концепция плоской онтологии отталкивается от критики тотальности, возникающей, в первую очередь на обуславливающих её отношениях интериорности. Суть этого заключается в том, что быть частью целого – есть неотъемлемое свойство его элементов, и только такое целое может собственно характеризоваться как нечто целое. В то время как отношения экстериорности, оставляющие частям определённую степень свободы, не могут стать основой для формирования собственно целого – оно может быть лишь механистическим объединением независимых друг от друга объектов.

Начиная с Гегеля, отношения интериорности – то, что наделяет целое эмерджентными свойствами. Деланда предлагает альтернативу, вслед за Делёзом вводя понятие ассамбляжа – целого, основанного на отношениях экстериорности, эмерджентность которого обусловлена отношениями, возникающими между его частями. Свойства, возникающие у целого в таком случае, зависят от свойств объектов, являющихся его частями, но не сводятся к ним, поскольку являются актуальным проявлением способностей.

Таким образом, ассамбляж становится главной категорией онтологии, предлагаемой Деландой, поскольку он с одной стороны способствует

настоящему синтезу между составляющими его компонентами, а с другой – может легко разбираться на составные части самых разных масштабов, которые могут легко входить в состав новых ассамбляжей.

Из экстерииорности ассамбляжа вытекает историзм, определяющий подход Деланды к исследованию ассамбляжей. В силу того, что отношения между частями ассамбляжа не являются логически необходимыми, их сборки нельзя исследовать чисто умозрительно. Каждый ассамбляж уникален и определяется своим историческим развитием, в ход которого складывается та или иная гетерогенность ассамбляжа. Вообще, ассамбляж – это конструкция «пульсирующая», которой необходимо каждый раз пересоздавать свою целостность. Этот же принцип позволяет рассматривать как ассамбляжи абсолютно всё: так, даже биологический организм есть ассамбляж составляющих его органов, так как отношения между ними не есть логически необходимые, а лишь результат совместной коэволюции этих органов.

Помимо своей истории любой ассамбляж может быть описан по двум своеобразным осям координат:

1. Материальная – экспрессивная
2. Процесс территориализации – процесс детерриторизации

Первая ось, очевидно, описывает характер объектов, входящих в ассамбляж: являются ли они собственно «материальными» объектами или «идеальными» сущностями; разумеется, ассамбляж может одновременно включать в себя в разных пропорциях объекты и того и другого типа. Вторая ось также понимается буквально и включает в себя процессы, организующие ассамбляж в физическом пространстве: или ограничивающие его и превращающие в локальное целое, или напротив нацеленные на расширение его масштабов и отрыва от четкой географической привязки. Применительно к человеческому обществу яркий пример детерриторизации – это процесс развития технологий коммуникаций, позволивших людям создавать объединения, члены которых находятся в разных географических точках.

Эти две оси отображают аналитическую и синтетическую стороны ассамбляжа, связанные с уже упомянутой пульсацией идентичности ассамбляжа. Каждое подобное объединение одновременно включает в себя и процессы, усиливающие и сохраняющие уникальную идентичность объединения, так и процессы, напротив, размывающие её, способствующие переходу ассамбляжа из одного состояния в другое, или даже превращению его в совершенно другую сборку.

Но, помимо territorизации и детерриторизации, существует ещё одна ось, обеспечивающая синтетические свойства ассамбляжа – ось кодирования-декодирования. Если аспект географического распределения ассамбляжа связан, по большей части, с его материальной составляющей, то вводимая вторая ось – связана в большей мере с экспрессивной составляющей.

Кодирование – процесс, поддерживающий идентичность ассамбляжа через передачу особой информации с помощью отдельных экспрессивных элементов. Если territorизация поддерживает идентичность ассамбляжа в пространстве, то кодирование – способствует его целостности во времени. В то же время, процесс декодирования – это точка, в которой экспрессивные элементы ассамбляжа могут двинуться в другую сторону. Так, каждый конкретный биологический организм строится на основе информации, закодированной в его геноме, но после её декодирования, он волен вести себя по-иному, передавая последующим поколениям информацию уже иного вида.

Важным выводом из намеченной модели ассамбляжа является отказ от категории сущности как структурирующей онтологию единицы. Если до этого, например, биологический вид понимался как сущность, то есть некий набор абстрактных идеальных свойств, на основе которых некий набор объектов мог быть выделен из общей структуры бытия, то теперь на её место приходит ассамбляж. Что, в свою очередь, приводит к тому, что биологический вид теперь – это результат сложной исторической эволюции,

и существует он не как абстрактная единица, а как конкретная популяция, уникальная в силу особенностей своего развития.

Мышление в категориях ассамбляжей и их популяций, вкупе со свойствами самих ассамбляжей приводит к «уплощению» онтологии, выражающемуся в сведении микро- и макро- уровней на один онтологический уровень. В силу того, что отношения внутри ассамбляжей экстерииорны – каждый ассамбляж или его часть легко может становиться частью другого, а иерархия онтологических уровней сменяется различными масштабами. Больше нет целого и части, есть лишь ассамбляж с большим числом компонентов (большей территории и т.д.) и ассамбляж меньше.

Ключевой проблемой, так и не решённой в онтологии Деланды, становится проблема причинности. Историзм, выступающий ключевым принципом, определяющим сущность самих ассамбляжей, подразумевает некую цепочку событий, приводящих сборку к тому состоянию, в котором её застаёт исследователь. При этом в силу сложности самих ассамбляжей, в плоской онтологии нет места простым причинно-следственным связям, мы никогда не можем сказать наверняка, что послужило причиной того или иного события. Из-за чего причинно-следственные связи не то чтобы становятся нелинейными, они как будто бы вовсе исчезают, что создаёт больше трудности для конкретных исследований ассамбляжей.

Теория ассамбляжей Деланда привлекла к себе внимание ряда философов и послужила своеобразной основой для формирования объектно-ориентированной онтологии – подхода, существующего в рамках спекулятивного реализма, возникающего в конце 2000-ых начале 2010-ых годов.

Ключевым представителем объектно-ориентированной онтологии является Грахам Харман, развивающий идеи Деланды, стремясь разрешить возникающую проблему причинности и точнее определить сам онтологический ландшафт [41]. Для этого Харман отходит от самих ассамбляжей к составляющим их частям – объектам, осуществляя, таким

образом, углубление концепции изучения «реальности» вместо абстрактной модели.

Ключевой идеей Хармана является отказ от разделения материального и идеального, попытки говорить об абстрактных вещах [39]. Всё является объектами, а потому, во-первых, наделено одним онтологическим статусом, и, во-вторых, мир вообще состоит только из объектов, вступающих в различные отношения друг с другом без каких-либо посредников.

Благодаря этому Харман формирует уникальную модель взаимодействия объектов, в которой любое взаимодействие есть результат взаимодействия двух объектов, обусловленный исключительно природой каждого из объектов.

Таким образом, ключевыми идеями объектно-ориентированной онтологии и теории ассамбляжей становятся:

1. Максимальное «уплощение» онтологии – выравнивание онтологического статуса объектов.
2. Возврат к реализму – исследование исключительно существующих объектов, а не их абстрактных моделей.
3. Значительный упор на историзм – обусловленность онтологического статуса вектором развития объекта.

Теперь посмотрим, как эти и другие философские идеи отражаются и преобразуются в акторно-сетевой теории Бруно Латура, являющейся не столько философской моделью, сколько конкретным методологическим аппаратом исследования действительности.

### *1.3.2 Основные положения акторно-сетевой теории Б. Латура*

Акторно-сетевая теория – активно развивающаяся социальная теория, возникающая в конце 1980-ых – начале 1990-ых годов, первоначально в рамках STS. Здесь важно отметить, что, несмотря на то, что многими теориями оценивается как применимая в первую очередь к исследованиям техники, она довольно быстро переросла эту область, а возникла изначально



как выводы из работ Латура, касающиеся его исследований научного процесса в лабораториях.

ANT активно видоизменялась, как в работах самого Латура, так и в работах его последователей [23]. Связано это было и с пересмотром ряда изначальных положений теории, и с расширением сферы её применения, и с последовавшей за этим «философизацией» самого Латура, перешедшего от конкретных практических исследований к, по большей части, работам теоретического и даже философского толка.

Несмотря на то, что в последних работах Латура ANT предстаёт как часть более крупной онтологической модели, в которую она включается как один из «модусов» существования, мы остановимся именно на ней, как полностью удовлетворяющей целям данного исследования и как более разработанной, чем современная концепция исследователя. Также предварительно стоит отметить, что название «акторно-сетевая теория» не принадлежит Латуру и обычно критикуется им самим, а в качестве более адекватной альтернативы предлагается термин «социология перевода».

Начнём наше рассмотрение со зрелой концепции, в которой ANT выступает своеобразной альтернативой устоявшейся социологической теории, ключевой характеристикой которой является представление о существовании неких «социальных сил», управляющих объектами независимо от их желаний. В теориях подобного рода исследователь вынужден всё время интерпретировать действия своих респондентов, стремясь отыскать некую подоплёку, не осознаваемую самим объектом.

Первое, что делает Латур – отказывается от этого представления. Сам по себе этот шаг осуществляется целиком в духе объектно-ориентированной онтологии: существуют только конкретные социальные объекты и взаимодействия между ними, никакие социальные силы, выступающие в роли субстанций, связывающих эти объекты, не могут быть включены в онтологию. Исследователь может фиксировать только то, что прослеживают

сами объекты. Этот принцип заключается в известной установке любого исследования в русле ANT: «следовать за акторами».

Если классическая социология, наполненная социальными силами, исследует относительно стабильное «социальное», уже существующее и целостное, то ANT исследует социальное «неопределённое», складывающееся на глазах исследователя. Таким образом, общество в ANT – это ассамбляж в духе Деланда – исторически обусловленное экстериорное целое, которому необходимо каждый раз пересоздаваться, определяя саму идентичность. Латур выделяет несколько «источников неопределённости», за счёт обращения к которым исследователь может понять механизмы этого переопределения.

Первое – это отсутствие чётких и навсегда определённых групп. Как пишет Латур: «Групп нет, есть группообразование». По сути, социум, таким образом, раскладывается на множество ассамбляжей, которым также необходимо каждый раз определять свою идентичность и прилагать усилия для поддержания целостности. Вообще, по отношению к теории Деланды, ANT делает более серьёзный акцент на нестабильности создаваемых целостностей. Любой ассамбляж в контексте ANT вынужден прилагать гигантские усилия, чтобы сохранять свою идентичность, иначе он мгновенно превращается в образование иного характера.

При этом образование групп связано с рядом процессов. Прежде всего, происходит определение самого представителя группы – из кого она состоит. После чего выделяются антигруппы, с которыми определяющая себя группа неизменно вступает в противоречие. Также представители группы всегда стремятся чётко о-граничить её, придав ей территориальные или символические границы. Как можно заметить, группа определяется всегда через некоторые действия, и это очень важный элемент ANT – объект существует, только если он действует.

Это положение серьёзно отличает концепцию Латура от объектно-ориентированной онтологии, где это положение не проявлено с остротой

необходимости. Как только объект перестаёт воздействовать на другие объекты – он сразу же исчезает. Именно поэтому те же социальные группы могут проявить себя только во взаимодействии с другими ассамбляжами и никак иначе.

Характер взаимодействий – самая оригинальная сторона акторно-сетевой теории. Согласно Латуру, действия всегда «захватываются», то есть никогда не действует один объект, но некоторая их совокупность. В этом смысле абсолютно все объекты являются акторами – переносчиками взаимодействия. Но специфика актора заключается в том, что в процессе переноса воздействия он всегда трансформирует его, видоизменяет, причём зачастую неожиданным способом.

Сами акторы никогда не видят перед собой всей картины, они не знают первоисточника воздействия, его причин или целей. Они лишь «переносят» это воздействие на некотором участке сети, попутно трансформируя его, что есть неотъемлемая часть их природы. Но в то же время акторы стремятся сами понять осуществляемое ими действие, из-за чего вовлекаются не только в сам процесс воздействия, но и в окружающие его события: они строят свои теории действия, критикуют другие теории и другие способы захвата и передачи действия и так далее.

Подобный подход к теории действия потребовал расширения списка акторов, что само по себе является очень смелым шагом. В отличие от теории ассамбляжей Деланды, в которой присутствует чёткое разделение между природными, социальными, техническими и т.д. ассамбляжами, построенными по одинаковым принципам, но взаимодействующими между собой лишь на отдельных масштабах, ANT Латура включает в единую сеть акторов и социальные, и технические, и природные объекты.

Именно включение «не-человеков», как их называет сам Латур, в число «социальных» элементов, в своё время и способствовало статусу ANT как одной из STS моделей. Само введение технических устройств в социум связано с тем, что акторы-люди, передавая действие, вынуждены обращаться

к различным вещам, которые также действуют как акторы, трансформируя действие, преобразуя его и видоизменяя. Само это состояние актора прослеживается лишь в короткий момент передачи взаимодействия, и если эта передача хорошо отлажена, актер становится практически полностью незаметен, поскольку время его действия становится ничтожно малым.

Но помимо различных устройств в онтологию социального в качестве таких же акторов вводятся и природные объекты. Вместе с ними же в социальную теорию включаются и различные научные дисциплины, занимающиеся «изучением» этих природных объектов, а потому также выступающие как акторы, зачастую передающие взаимодействие от природы к человеку.

Здесь ANT впервые чётко проявляет себя как своеобразную мета-дисциплину, поскольку описывает сам характер отношения между наукой и объектом исследования как нечто, что может быть описано ANT, в свою очередь также являющейся научным подходом. Сами описываемые отношения полностью укладываются в контекст объектно-ориентированного подхода – исследуемый объект всегда обладает множеством измерений, вариантов существования. В связи с этим реальность состоит из «дискуссионных реалий», то есть описывается как набор равновозможных моделей, которые могут быть выведены из свойств одного объекта, оказывающегося разным в каждый момент времени, каждый раз по-новому раскрывающегося перед исследователем.

Для «преодоления» этого ограничения, сама ANT действует исходя из этого знания. Исследователь всегда следует за объектом своего исследования, прослеживая все связи, которые он образует с другими объектами. Исследователь не предлагает собственных интерпретаций, он лишь описывает объект в соответствии с теориями и интерпретациями, предлагаемыми самим объектом. Если же в ходе исследования действие передаётся другому актору, то исследователю не остаётся ничего другого как перейти к новому объекту.

Эти взаимодействия между акторами слагают сети, что и приближает ANT к плоским онтологиям. Передача взаимодействия сама по себе не подразумевает иерархии, напротив, все акторы совершенно равноправны. Важно понимать, что сеть – это не метафора, с помощью которой исследователь пытается описать мир, и уж тем более не стремится свести исследуемую систему к сети. Исследование – это описание прослеживаемых отношений, и, если всё сделано правильно, в этом описании неминуемо будет прослежена именно сеть трансформаций действия, передающегося от одного актора к другому.

Прикладные исследования Латура (в первую очередь, касающиеся исследования городских пространств) позволили ему выделить несколько паттернов, позволяющих вписать классические социальные теории в рамки ANT и заодно показать наиболее распространённые способы поддержания целостности социальных ассамбляжей.

В первую очередь Латур предлагает понятие «олигоптикума» для описания своеобразных акторов, являющихся узловыми для целого ряда воздействий, проходящих через них. Олигоптикум – это актор, существующий исключительно за счёт связи с другими акторами, от которых он получает информацию, обрабатывает её и стремится передать взаимодействие дальше, опираясь именно на полученную информацию.

Здесь и далее важно понимать, что олигоптикум – это не абстрактная модель. Это лишь один из способов существования специфического актора, который позволяет соотносить масштабы глобального и локального. Особенностью любого олигоптикума является то, что на самом деле он не способен проследить взаимодействие дальше акторов, непосредственно связанных с ним, но именно эти связи он прекрасно осознаёт. Яркий пример олигоптикума – это военный штаб, расположенный на фронте. Этот актор не способен проследить взаимодействия, выходящие за пределы его влияния, даже если они схожи с теми, которые он традиционно отслеживает, но уж про свой локальный участок фронта он знает практически всё.

Задачи олигоптикумов – поддерживать видимость чёткой структуры пространства, в котором существуют точки сосредоточения множества нитей. Хотя на самом деле, каждый олигоптикум очень локален и не претендует на какую-либо реальную глобальность.

Если олигоптикум создаёт точки, объясняющие ограниченное количество локальных взаимодействий, то для описания большого числа связей или даже всей картины целиком существуют «панорамы». Задача панорамы – дать представление о картине в целом, но при этом неизбежно теряется множество деталей, даже отдельные акторы могут выпадать из получаемой модели.

По сути именно панорамами являются все классические социальные теории – это наброски социального ландшафта, которые затем используются самими же акторами для описания своих взаимодействий и вписывания их в глобальный контекст. Для исследователя важно понимать панораму именно как тип актора и не сводить своё исследование к её описанию, выдаваемому за описание всей сети взаимодействия.

Если олигоптикумы и панорамы – это акторы, выполняющие функции «глобального контекста», но в духе ANT превращающие его в наблюдаемые и прослеживаемые локальности, то так называемые «локализаторы» - это акторы, способствующие разделению акторов, задающие характер их взаимодействия.

Хрестоматийный пример локализатора – это лекционная аудитория. Устроенная определённым образом, она не определяет конкретного взаимодействия, которое произойдёт в ней, но она, во-первых, локализует это взаимодействие в определённом географическом пространстве, а, во-вторых, создаёт некую матрицу для этого действия. В аудитории определены место преподавателя, места студентов, присутствуют определённая звукоизоляция и оборудование. Смещение в одном из этих компонентов может привести к видоизменению характера взаимодействия, поскольку аудитория как любой актор видоизменяет проходящее через неё действие.

Наконец, если локализаторы – это акторы, создающие сложные матрицы, в рамках которых происходят взаимодействия, то для того, чтобы эти социальные воздействия происходили, необходим ещё один тип акторов – «плагины», расширяющие возможности отдельных объектов, вступающих во взаимоотношения. Плагины – не являются неотъемлемыми элементами каждого объекта – это искусственная надстройка, ещё один актор, через которого передаётся часть действия. Например, яркий пример плагина – это различные юридические документы, через которые человек передаёт действия, связанные с его идентификацией.

Таким образом, ANT представляет собой сложную методологическую схему, направленную, в первую очередь, на описание общества как сложного ассамбляжа, состоящего из равноправных объектов-акторов, через которые передаются взаимодействия, трансформирующиеся в ходе этой передачи. Прикладной опыт исследователей позволил несколько точнее определить набор этих объектов, выделив в нём определённые закономерности, которые, тем не менее, являются не столько иерархическими единицами, сколько выражением отдельных паттернов и закономерностей, характерных в целом для сетей акторов.

### *1.3.3 Возможности применения ANT для исследования техники*

Одно из достижений акторно-сетевой теории – включение вещей в сферу социологического исследования через придание им статуса актора, обладающего теми же возможностями, что и человек. Именно эта идея привлекла внимание к теории в начале 1990-ых годов, закрепив её статус как одной из интересных STS моделей, позволяющих описать взаимоотношения между техническим устройством и его пользователем. Однако сейчас, когда ANT переросла масштабы исследований лишь техники и претендует на роль новой социальной теории, важно ещё раз выделить те возможности, которые данная модель предлагает для исследования техники.

Как уже отмечалось выше – ANT понимает вещи вообще как акторы, через которые передаются взаимодействия. Источник этих взаимодействий, как правило, скрыт, но, когда речь идёт о вещах, мы почти всегда можем проследить непосредственно момент делегирования действия предмету. Большинство прикладных исследований ANT отталкивается именно от этого локального делегирования, прослеживая затем сложную цепочку причин, приводящих к тому, что именно это действие было делегировано именно этому предмету.

Не менее интересным для исследований техники является и исследование того, как именно вещь проявляет себя как актор. То есть того, как делегированное действие видоизменяется будучи передано не-человеку. Как уже отмечалось выше, в повседневной жизни делегирование не-человекам доведено до автоматизма, из-за чего мы не замечаем тех изменений, которые происходят в результате этого процесса. Но Латур выделяет ряд случаев, в которых эта трансформирующая роль предметов становится более очевидной.

1. Новшества и инновации. Появление нового предмета всегда связано с изучением пользователем его возможностей и способов взаимодействия с ним. Ещё никто толком не знает, что с ним можно сделать, из-за чего объект зачастую может вести себя не так, как того ожидает пользователь. В этих случаях трансформация действия является наиболее очевидной.

2. Дистанцирование от привычных объектов. Во многом эта ситуация похожа на предыдущую, но вызывается исследователем искусственно. Естественным образом она возникает всё в тот же период обучения, когда мы не знаем, как именно пользоваться предметом. Мы можем даже не знать вообще, для чего этот объект предназначен и стремимся делегировать ему действия, которые вовсе не свойственны его природе. Опять же в этом случае поведение вещи непредсказуемо и наиболее ярко прослеживается её суть как актора.



3. Поломки. Это состояние связано с несоответствием между реальными действиями объектами и нашими от него ожиданиями. Мы делегируем некое взаимодействие, будучи уже нацеленными на то, чтобы получить привычный результат, но действие преобразуется совершенно немислимым образом, мы полностью теряем контроль над вещью и уже не можем точно сказать, к чему приведут другие попытки его использовать. Вновь – налицо действие вещи в качестве актора.

4. Воссоздание кризисных ситуаций по документам. Один из вариантов понять действие предмета в его качестве актора – обратиться к документальным свидетельствам подобных взаимодействий, происходивших в прошлом или вне зоны полевого исследования учёного. В этом случае важно понимать, что информация поступает от других акторов, а потому дополнительно трансформирована ими в соответствии с их собственными представлениями о характере действия, делегируемого объекту.

5. Художественная литература. Наконец, одним из ярких примеров проявления «акторной» природы объекта является художественная литература, в которой другими акторами приводятся модели взаимодействия с объектом, вынесенные в виртуальное пространство, где мысленно исследуются способы взаимодействия с ним. Опять же, при исследовании исторического аспекта художественная литература может выступать своеобразным источником, в котором зафиксированы уже ушедшие практики делегирования тех или иных действий.

В целом, любое исследование техники, осуществляемое в рамках ANT, сводится к бесконечным попыткам ответить на самый простой вопрос: «Что такое исследуемый объект?». Этот вопрос носит не только классификационный оттенок, но понимается именно на самом примитивном уровне: «Как отличить объект А от объекта Б? Какие условия необходимы для того, чтобы объект А был определён именно как объект А?». Находясь в пространстве, намеченном ANT, теорией ассамбляжей и объектно-ориентированной онтологией, мы понимаем, что ответ на этот вопрос



подразумевает прослеживание связей, образующихся между исследуемым объектом и другими акторами.

От теории ассамбляжей мы неизбежно перенимаем историческую перспективу. Во-первых, это связано и с моментом инновационности. Техническое устройство, впервые включаясь в человеческое общество, является ещё не известным актором. Другие элементы сети не знают, как именно с его помощью можно делегировать те или иные действия, да и вообще стоит ли, из-за чего возникает резкий всплеск, в результате которого и определяется место актора во всей сети.

Во-вторых, в ходе исторического развития артефакт, как и любой ассамбляж, видоизменяется, и его роль сейчас может быть описана исключительно исходя из вектора его развития. В ходе этого развития может многократно измениться не только характер объекта, но и набор действий, делегируемых ему.

Объектно-ориентированный подход привносит нам понимание технического устройства как отдельного объекта, реально существующего в мире, и не являющегося лишь абстракцией или мыслительной моделью. В совокупности с теорией ассамбляжей – мы получаем стартовую точку для исследования самого артефакта. Это исследование может происходить через исследование конкретных объектов, не сводимых к некоей теоретической абстракции. Каждый случай – это вариант проявления одного и того же объекта, суть которого мы и стремимся понять.

Акторно-сетевая теория предлагает нам весьма богатый набор инструментов для описания не только сетей взаимодействий акторов. Концепции олигоптикумов, панорам, локализаторов и плагинов дают нам возможность включать в исследование различные источники: от газетных статей до научных публикаций, что просто необходимо в том случае, когда речь идёт об исследовании исторического развития актора.

Наконец, в конце теоретической части работы стоит сказать, что же понимается под становлением артефакта в качестве актора. Сама по себе эта

формулировка кажется несколько противоречивой, поскольку любой объект в рамках ANT уже является актором и попросту не может быть ничем иным. Тем не менее, опираясь на историческое понимание становление ассамбляжа, стоит сказать, что актор не остаётся однородным.

Как уже упоминалось, в момент своего появления – артефакт дан лишь в качестве потенциального актора, ещё не связанного никакими взаимодействиями. До того, как объект будет включён в сеть делегирования – он как бы не существует вовсе. Становление актора – есть процесс определения тех действий, которые могут быть делегированы объекту и шаблонов, по которым эти делегирования будут происходить.

По сути, этот процесс реактуализируется при каждом обращении к вещи. Включая её в свою сеть, ассамбляж вновь вынужден затратить усилия на то, чтобы переопределить её возможности и доступные ей действия. Тем не менее, наиболее ярко этот процесс можно проследить именно в точке «появления» артефакта, когда привычных шаблонов ещё попросту не существует.

## 2 СТАНОВЛЕНИЕ ПЕРСОНАЛЬНОГО КОМПЬЮТЕРА МОДЕЛИ APPLE II КАК АКТОРА.

Персональный компьютер Apple II традиционно оценивается как первый успешный серийный персональный компьютер, сформировавший потребительский рынок, ставший доступным простым пользователям, а не только тем, кто, так или иначе, был знаком с программированием или инженерией. Данный компьютер появился в 1977 году и к этому времени рынок цифровой техники существовал уже давно – крупные ЭВМ, использовавшиеся в научных центрах, различные цифровые игровые автоматы и домашние игровые консоли, несколько моделей любительских персональных компьютеров. Тем не менее, следуя методу акторно-сетевой теории, мы обратимся сразу к артефакту как таковому, отталкиваясь от его особенностей.

Прежде всего, нас будет интересовать тот потенциал, который заложен в модели изначально, наша задача – определить то, какие именно действия способен передавать компьютер в теории, и какие задачи на нём решались и могли решаться на практике. Для этого, мы последовательно будем определять модель, стремясь выявить суть артефакта, понять, что именно он из себя представляет. Определение это будет происходить на нескольких «уровнях», выделение которых вдохновленно работами Яна Богоста в области видеоигр:

1. Хардварная составляющая – мы рассмотрим Apple II с точки зрения инженерии, условно говоря, обратимся к материальной составляющей объекта, которая определяет пределы возможностей модели.

2. Программная составляющая – на этом уровне нас будет интересовать программное обеспечение, как предустановленное в модель, так и то, которое для компьютера разрабатывалось – какие задачи оно решало, как происходило взаимодействие между программой и пользователем.

3. Социальная составляющая – те культурные и иные социальные практики, которые формируются вокруг артефакта, как именно они включаются в общий социальный контекст. По сути, именно на этом этапе мы и обращаемся к пониманию артефакта как актора, который уже осознано используется для передачи действий.

Сразу отметим, что в данной работе мы будем рассматривать исключительно оригинальную модель Apple II, а не последующие выпуски, первый из которых датируется уже 1979 годом. Также, рассматривая программное обеспечение и социальные практики, мы ограничимся исключительно 1970-ми годами, тем самым обратимся к самому началу процесса формирования персонального компьютера.

## **2.1 Хардварная составляющая компьютера Apple II**

Прежде чем обратиться к рассмотрению данного уровня артефакта, необходимо в общих чертах обрисовать, что мы подразумеваем под хардварной составляющей, почему этот блок рассматривается в первую очередь и почему он является одним из самых сложных для рассмотрения.

Традиционно в компьютерных науках под хардварной частью понимается собственно «начинка» техники – процессоры, видеоадаптеры, устройства ввода-вывода и различные порты и расширения. При работе с современным компьютером – это та часть, которая максимально скрыта от конечного пользователя, всё это расположено внутри объекта и не требует непосредственного вмешательства, не считая исключительных случаев поломок или необходимости улучшить устройство. И даже в этом случае, пользователь, как правило, обращается к специалисту.

При этом именно этот уровень и определяет принципиальные возможности устройства, определяя спектр задач, которые оно сможет решить. Именно поэтому мы и обращаемся в первую очередь именно к этому

пласту – для компьютера именно хардварная часть сосредотачивает в себе изначальные «потенции», заложенные разработчиком в устройство.

Однако мы расширим спектр вопросов, относящихся к этому уровню, и не будем ограничиваться рассмотрением характеристик устройства. Как уже упоминалось выше, под хардварной составляющей мы понимаем в целом «материальный», «вещный» статус объекта, а потому сюда же мы отнесём вопросы, касающиеся внешнего вида устройства и эргономики. Иными словами – нас интересуют не только технические характеристики отдельных элементов компьютера, но и способ, которым они собираются в нечто целое, идентифицируемое как «персональный компьютер Apple II».

Сразу отметим, что данный уровень является самым сложным для рассмотрения в силу отсутствия у нас непосредственного доступа к реальной модели компьютера, из-за чего у нас нет возможности через непосредственный опыт понять, чем же является Apple II как вещь. Тем не менее, изучение данного уровня принципиально возможно благодаря литературным источникам, фиксирующим характеристики данного компьютера и ближайших конкурентов и отражающим некоторые особенности его конструкции.

### *2.1.1 Apple II как целое*

Прежде чем заняться отдельными элементами системы и их характеристиками, рассмотрим Apple II как целое, поскольку в этой целостности также заключена большая доля его успеха и уникальности для того времени. Персональные компьютеры до Apple II ориентировались на любителей электроники – людей, обладающих определёнными навыками в радиоэлектронике и программировании, способных самостоятельно собрать машину и подключить к ней всё необходимое.

Для США ярким примером такого подхода является компьютер Apple I. Сам по себе этот компьютер очень похож на Apple II (что не удивительно, ведь между выпуском машин прошло около года), но на рассматриваемом

сейчас уровне между ними есть одно существенное отличие. Apple I – это просто плата с установленными на неё микропроцессорами, в то время как Apple II – это уже корпус, в который помещена подобная плата.

Сам по себе способ взаимодействия с компьютером принципиально изменяется. Если вы приобретаете компьютер 1976 года, это означает, что вам необходимо самостоятельно поместить его в корпус, подключить к нему устройство ввода текста, монитор и другие внешние блоки, необходимые для его работы. Вы получаете лишь основу, вокруг которой создаёте своё устройство. В случае с Apple II вы получаете не просто плату, но уже нечто «готовое», с чем вы можете работать практически сразу.

Последствием такого решения является не просто упрощение способа взаимодействия между пользователем и компьютером, но и создание визуального образа компьютера. По сути, Apple II становится одной из первых машин, которую можно узнать, машиной, обладающей собственным дизайном. Ведь все платы выглядят одинаково, если не приглядываться к ним, а серая коробка с клавиатурой и цветным яблоком – уже отличается от других подобных устройств.

Интересно, что Apple II поставлялся и в более привычном для любителей техники виде – как самая обыкновенная плата, тело которой мог создать уже сам пользователь. При этом цена продукта снижалась практически вдвое. Но подобный способ взаимодействия с машиной явно не рассматривался разработчиками как основной. Почему? Чтобы ответить на этот вопрос нам необходимо «открыть» коробку с новым Apple II.

Что получает пользователь вместе с компьютером? Уже упоминавшуюся серую пластиковую коробку с встроенной клавиатурой, внутрь которой заключена плата, собственно и являющаяся компьютером. Помимо того, что в эту коробку уже были встроены все необходимые для подключения к монитору разъёмы, к ней также прилагались игровые джойстики.



Мы ещё вернёмся к этому элементу, поскольку сама по себе история Apple II оказывается неразрывно связанной с видеоиграми. Сейчас важно лишь отметить, что комплектация именно «системы» Apple II предоставляла пользователю больше способов взаимодействия с машиной, значительно упрощая жизнь. Продажа чипа, по сути, существовала именно для любителей – того весьма ограниченного круга людей, которым была интересна машина сама по себе, в отрыве от её функционального применения.

Ещё ненадолго остановимся на внешнем виде самого блока, перед тем как обратиться непосредственно к плате, являющейся собственно компьютером. До Apple II пользователи уже видели технику «из коробки» - в первую очередь, конечно же, это были ЭВМ. При этом само по себе их решение кардинально отличалось от того, что было предложено Apple Computers.

Традиционно, для корпусов использовался металл или дерево, что придавало технике солидный, статусный вид. Но и для обывателя к 1977 году, подобное решение уже строго ассоциировалось с профессиональной техникой, которая расположена в крупных вычислительных центрах и университетах, а потому совершенно не подходит для использования в доме или офисе. Не улучшал ситуацию и тот факт, что любительские персональные компьютеры также помещались в деревянные или металлические корпуса, так как эти материалы были доступны людям.

Именно поэтому для Apple II был использован пластик. Задача корпуса сводилась не просто к тому, чтобы дать пользователю компьютер, который он может сразу же начинать использовать, не задумываясь о его сборке, но и показать, что это техника нового образца. Сам по себе внешний вид компьютера должен сообщить потенциальному пользователю, что он прост и доступен.

Таким образом, Apple II как некий целостный материальный объект выступает как некий сигнал для человека – он говорит о своей простоте в использовании, доступности и коренном отличии от предшествующих

моделей. Теперь мы обратимся к отдельным элементам, составляющим сам компьютер, начнём с самой платы и разложим её на отдельные составляющие – больше всего нас интересуют процессор, встроенная память и видеоадаптер. После чего отдельно рассмотрим устройства ввода-вывода.

### *2.1.2 Процессор компьютера Apple II*

Начиная разговор о процессоре компьютера Apple II важно отметить, что именно эта часть компьютера сделала его существование принципиально возможным. Дело в том, что до 1970-ых годов в электронике использовались процессоры, реализованные на платах общего назначения, и только в начале десятилетия появляются первые микропроцессоры, реализованные в виде одной микросхемы.

До появления микропроцессоров создание мощной техники небольшого размера было попросту невозможным. Также осложнена была замена или улучшение процессора в уже разработанном устройстве. Однако развитие секторов, использующих ЭВМ, требовало всё более сложных вычислений причём в доступных условиях, и для решения этих задач были разработаны первые 4-битовые микропроцессоры, изначально использовавшиеся в калькуляторах.

Здесь стоит отметить, что первоначально изготовление микропроцессоров, сравнимых по производительности с уже существовавшими процессорами было осложнено. Микропроцессоры требовали значительных инженерных усилий, направленных на проработку их архитектуры, поэтому потребовалось ещё несколько лет на то, чтобы ввести в производство 8-битные микропроцессоры.

Тем не менее, к середине 1970-ых годов цифровая техника уже активно использовала микропроцессоры, что позволило значительно уменьшить размеры устройств и подготовило платформу для появления продуктов, ориентирующихся на широкий потребительский рынок. Традиционно считается, что сдерживающим фактором долгое время была высокая цена

микропроцессоров, выпускавшихся компаниями Intel и Motorola. Переломным моментом становится создание микропроцессора MOS Technology 6502, который можно назвать основным процессором для 8-битных устройств вплоть до середины 1980-ых годов.

Его успех связывается с очень низкой ценой по сравнению с конкурентами. Будучи выпущенным в 1975 году, процессор стоил 25\$, в то время как его конкуренты, процессоры Motorola 6800 и Intel 8080, отпускались за 179\$. Ближайшим конкурентом MOS 6502 становится процессор Zilog Z80, стоивший около 60\$.

Тем не менее, именно MOS 6502 становится основой как для Apple I, так и для Apple II, и помимо того, что этот выбор позволил сделать компьютер доступным широкому потребителю, он, безусловно, повлиял на то, какой потенциал был заложен в компьютер.

MOS 6502 стоил значительно дешевле конкурентов не просто так – процессор использовал иную архитектуру, из-за чего способ обращения с ним значительно отличался от аналогов. Основная особенность заключалась в том, что процессор был «хуже», если посмотреть на характеристики – ниже тактовая частота, меньшее число доступных регистров, меньше оперативной памяти. Но по факту он достигал примерно тех же показателей за счёт оптимизации выполнения программных команд, из-за чего код выполнялся не просто быстрее, но и за фиксированное время (мы подробнее остановимся на этом вопросе в части, касающейся программной составляющей).

Из-за этого, MOS 6502 становится процессором, который сильно зависит от написанных для него программ, поскольку изначальный набор команд ассемблера крайне ограничен (например, процессор требовал специально написанных программ для целочисленного умножения или деления). Если его конкуренты – это процессоры, направленные на достижение высоких показателей, упрощающих работу программиста, то этот процессор требует, чтобы для него было написано программное обеспечение.



Подобный подход оказал непосредственное влияние не только на сам компьютер, но и на аксессуары, создававшиеся для него. Стив Возняк в последствие отметит, что при разработке этих устройств он стремился сократить количество физических элементов на плате, заставляя выполнять их функции отдельные программы.

Таким образом, на данном этапе важно отметить, что выбор процессора заранее определяет способ программирования компьютера Apple II. Именно поэтому мы ещё раз вернёмся к некоторым его особенностям в разговоре о программной составляющей машины.

### *2.1.3 Память компьютера Apple II.*

Хранение данных является неотъемлемой составляющей любого компьютера, ведь именно для их обработки он и существует. При этом уже на самой заре компьютерной техники происходит разделение типов памяти на ОЗУ (оперативная память) и ПЗУ (постоянная память). Ключевое отличие между ними заключается в том, что данные в ОЗУ хранятся исключительно в то время, когда компьютер включен, а данные в ПЗУ не зависят от питания, и хранятся в компьютере постоянно.

Apple II поставлялся с 4 килобайтами ОЗУ и 4 килобайтами ПЗУ, но имел возможность расширения блока ОЗУ вплоть до 48 килобайт (на самом деле, можно было купить модель сразу обладающую максимальным количеством памяти). Достигалось это за счёт наличия в плате специальных разъёмов, в которые могли вставляться дополнительные модули оперативной памяти на 4 или 16 КБ.

Для современного пользователя компьютера подобная практика может показаться абсурдной, для современных машин напротив характерен объём ПЗУ многократно превышающий ОЗУ. Для Apple II подобное распределение памяти означает принципиально другой способ обращения с компьютером, который является именно инструментом для обработки данных, а не для их хранения.

На самом деле, конечный пользователь практически не имеет доступа к ПЗУ – она практически полностью занята программным обеспечением, написанным Возняком и необходимым для функционирования машины. Все операции пользователя осуществляются внутри оперативной памяти, а потому с выключением машины полностью исчезают.

Именно поэтому, хранение данных фактически отделяется собственно от машины – необходимо хранить их на иных устройствах. Первоначально такими устройствами становятся магнитные кассеты, которые устройство считывало через обычный кассетный магнитофон, подключаемый к компьютеру через отдельный порт. Однако уже в 1978 году появляется Disk II – разработанный специально для Apple II дисковод для чтения дискет, что фактически способствовало увеличению объема данных, которые мог хранить пользователь на «компьютере».

С этого времени Disk II становится неотъемлемой частью компьютера, по сути, являясь тем самым расширением ПЗУ. «Ревизии» компьютера, выходившие после 1978 года, даже комплектовались этим устройством. Сам по себе принцип организации Disk II очень схож с тем, что использовалось для самой Apple II – свести к минимуму наличие различных микросхем, передав максимальное число операций к программе и добиться минимально возможной цены, для обеспечения свободного доступа к аппарату.

Одна дискета могла вместить 140 килобайт данных при записи с одной стороны. Использование другой стороны позволяло расширить объем информации вдвое, но требовало провести дополнительные операции над самой дискетой (вручную прорезать отверстие для доступа к другой стороне диска). Программная составляющая дискового привода позволяла полностью переписать данные на дискете.

Таким образом, само хранение данных было вынесено за пределы компьютера, который оставался просто инструментом для работы с ними. В случае с памятью Apple II мы сталкиваемся и с другой интересной

составляющей компьютера и вообще спецификой становления его как актора – наличие плагинов.

Как уже упоминалось выше, «плагин» - это вообще важное для акторно-сетевой теории понятие, представляющее идею внешних расширений к артефакту, увеличивающих спектр его возможностей. Для компьютера вообще «плагины» становятся очень важными элементами, позволяющими полностью преобразить функционирование машины. Disk II – первый пример «плагины», который мы встречаем – расширение, способное превратить Apple II в артефакт, использующийся для хранения информации.

И здесь же мы видим, как сильно артефакт изменяет способ хранения этой самой информации. Уже сейчас, исключительно на уровне хардварной составляющей, мы видим, что хранение информации с помощью Apple II требует наличия дополнительных устройств, одно из которых будет связано собственно с хранением данных, а другое – с передачей данных компьютеру.

Очевидно, что подобный подход определяет и сами практики, связанные с хранением данных и их передачи. Ведь информация получает конкретное физическое воплощение (дискета или магнитная кассета), и передаётся именно этот физический объект в отрыве от самого устройства. Эта модель отличается от той, что существует сейчас, когда информация хранится на отдельных устройствах, и передаётся между ними. Мы ещё вернёмся к вопросу о способе передачи информации в связи с социальным уровнем существования Apple II.

#### *2.1.4 Видеоконтроллер компьютера Apple II.*

Видеоконтроллер Apple II традиционно рассматривается как самая яркая особенность компьютера, во многом способствовавшая его популярности и успеху. Связано это с тем, что компьютер позволял отображать цветную графику с поддержкой 4 цветов в высоком разрешении и 16 цветов в низком разрешении. Ближайшие аналоги зачастую

использовали или исключительно монохромную графику, или вообще были способны отображать только текст.

Подобное решение тесно связано с развивавшейся в то же время в США игровой индустрией. В последствие Стив Возняк рассказывал, что создавая видеоконтроллер Apple II, он исходил из того, что компьютер должен быть способен запускать игры, наподобие тех аркад, разработкой которой Возняк занимался для Atari. Это очень важно понимать, поскольку ориентация на развивавшуюся игровую индустрию способствовала формированию компьютера именно как инструмента, использующегося в домашних условиях, а потому способного выполнять не только задачи, связанные с профессиональной деятельностью.

Видеоконтроллер накладывает непосредственное ограничение на графическую систему, чётко задавая её возможности и ограничения. Разрешение экрана может быть 280 x 198 (монохромный), 140 x 198 (4 цвета) или 40 x 48 (16 цветов) и никак иначе. Важно отметить, что видеоконтроллер вводит в программную составляющую понятие «пикселя» как элементарной графической единицы.

«Пиксель» в данном случае является именно программной абстракцией, не связанной со способом представления графики на экране, который подключался к компьютеру. Мы вновь видим как с одной стороны аппаратная составляющая задаёт рамку для программирования при этом передавая значительную часть выполняемых ей задач именно на откуп последнему.

Видеоконтроллер ответственен не только за показ графических изображений, он же ответственен и за вывод текста на экран. Опять же, он задаёт чёткое ограничение на количество строк текста и способ отображения знаков – они чёрно-белые и могут отображаться в трёх режимах «обычном», «инверсивном» и «мигающем». Также поддерживал видеоконтроллер и «смешанный» режим представления, в котором в нижней части экрана



располагается текст, а в верхней – изображение в одном из поддерживаемых графических режимов.

Таким образом, видеоконтроллер Apple II определяет, что именно мы сможем увидеть на экране компьютера, при этом заранее определяя некоторые паттерны и способы взаимодействия с ним конечного пользователя. Видеоконтроллер же является при этом самой конкретной для непрофессионального пользователя частью компьютера, поскольку его характеристики влияют на изображение на экране – то, что максимально конкретно и доступно для пользователя. При этом именно его характеристики позволяют расширить сферу применения компьютера, сделав его не просто рабочей офисной техникой.

#### *2.1.5 Устройства ввода компьютера Apple II.*

Комплект Apple II включал в себя клавиатуру, встроенную в бокс с платой, и пару игровых контроллеров, классических для того времени – одна кнопка и один джойстик, считывавший восемь направлений. При этом при желании в порт расширения для джойстика можно было подключить мышь, но это устройство ввода получает распространение только в 1980-ых годах, а потому мы рассматривать его не будем.

В целом, устройства ввода компьютера Apple II не являются чем-то уникальным для середины 1970-ых годов. Интересен сам способ их размещения в компьютере и сам факт наличия их в комплекте.

Клавиатура – это необходимое средство взаимодействия с машиной, без неё компьютер становится бесполезным, поскольку пользователь не может ему ничего сообщить. Подчёркивая эту необходимость – клавиатура встроена в корпус, в то же время, игровые контроллеры являются лишь подключаемыми элементами, без которых машина всё ещё может функционировать.

Наибольший интерес представляет сам факт наличия контроллеров, лишний раз подтверждающий тесную связь Apple II с играми. Более того, в

комплекте с компьютером, в качестве демонстрационной кассеты, прилагалась игра «Breakout», в которой контроллер и использовался, за счёт чего пользователь мог сразу же увидеть, на что способна машина.

Здесь стоит отметить, что сама по себе практика игры в видеоигры дома к тому времени только начинала складываться – самая заметная консоль второго поколения Atari 2600 выходит в том же году, что и Apple II, а предшествовавшие ей образцы не пользовались серьёзным спросом. То есть Apple II становится одним из первых устройств, позиционирующих себя как нечто пригодное для игры в домашних условиях.

Любопытно, что при этом игровое пространство выносится за пределы компьютера, выступает очередным плагином, наподобие ПЗУ. Пользователь не сможет использовать клавиатуру, являющуюся основным способом взаимодействия с Apple II, ему необходимо обратиться к стороннему устройству, пригодному исключительно для игр. По сути, игровые практики позиционируются как одна из возможностей компьютера, но возможность сторонняя, вынесенная за его пределы и подключаемая к нему при желании.

Более того, сам факт того, что с помощью клавиатуры в большую часть игр поиграть не получится, говорит о том, что перенимая на себя задачу обеспечения игрового опыта в домашних условиях, Apple II стремится скорее повторить опыт, который создавали игровые автоматы того времени. То есть «игровой плагин» для компьютера способствует своеобразной «мимикрии» устройства под другой артефакт, стремясь подменить его.

На этом мы завершим рассмотрение хардварного уровня, но перед тем как перейти к программам, используемым в компьютере, отметим наиболее важные черты компьютера как актора, проявленные на этом уровне:

1. Apple II – машина, предназначенная не только для профессионалов, поставляющаяся уже в готовом виде, что позволяет пользователю не вдаваться в подробности, касающиеся её внутреннего устройства.

2. Хардварный уровень Apple II устроен таким образом, что часть операций, традиционно выполнявшихся на этом уровне, теперь вынесена в программы.

3. Apple II – устройство, чьей основной задачей является динамическая обработка данных, при этом с помощью отдельных плагинов оно может использоваться и для других целей: например, для обеспечения хранения данных или игр.

## 2.2 Программная составляющая компьютера Apple II

Разобравшись с устройством компьютера, обратимся к способам взаимодействия с ним. Взаимодействие с Apple II тесно связано с программным уровнем, в который входит не только ПО, написанное для машины, но и сам по себе процесс взаимодействия с компьютером, поскольку он и сводился к своеобразной форме программирования.

Рассматривая программную составляющую, мы будем следовать логике изучения артефакта как актора. То есть обращаться к программам в той последовательности, в которой они могли осваиваться пользователем, который впервые запустил компьютер. Поэтому сначала мы обратимся к встроенному в компьютер ПО, затем рассмотрим отдельные программы для Apple II, вышедшие до 1980 года, после чего обратимся к практике написания собственных программ для компьютера.

Прежде чем переходить к описанию уровней, отметим, что изучение программной составляющей происходит через взаимодействие с эмулятором Apple II. Подобное решение кажется нам адекватным, поскольку довольно точно передаёт опыт от взаимодействия с компьютером в общих чертах. При этом важно понимать, что подобный подход нельзя считать максимально точным, поскольку он не отражает механического взаимодействия с техникой, эмулятор такого простого устройства может работать быстрее, чем оригинальный компьютер, да и сам по себе монитор не отражает того, как

картинка выглядела в действительности. Тем не менее, сами алгоритмы передаются довольно точно, а именно они и являются главным для данного уровня.

### *2.2.1 Встроенное ПО компьютера Apple II и базовые операции с ним.*

Подключив к компьютеру монитор, кассетный магнитофон (вспомним, что в 1977 году Disk II ещё не был разработан и в качестве ПЗУ использовались кассеты), включив устройство в розетку (соблюдая предварительно все рекомендации касающиеся напряжений), пользователь запускает компьютер и видит чёрный экран с надписью Apple II в верху экрана. Если нажимать кнопки на клавиатуре – никакого эффекта, ничего не происходит. В чём же дело?

Обратившись к инструкции, мы увидим, что необходимо нажать кнопку «reset», после чего на экране появится мигающий курсор, перед которым расположен знак «\*», и буквы, которые пользователь вводит с клавиатуры, начнут появляться на экране. Однако следом руководство пользователя рекомендует нажать ещё одну комбинацию клавиш (Str+B) и перейти в режим «BASIC», после чего перед мерцающим курсором появится знак «>». Что же всё это значит?

Взаимодействие с Apple II всегда связано с написанием программного кода, который компьютер стремится тут же выполнить. Именно для этого существует встроенный в компьютер модуль ПЗУ, который помимо интерфейсов для различных подключаемых устройств содержит интерпретаторы для двух языков, поддерживаемых компьютером.

Сразу после запуска системы пользователь может использовать машинные команды «монитора» - программа, позволяющая пользователю манипулировать непосредственно с ячейками оперативной памяти. Монитор способен выполнять команды на очень низком уровне – помещать данные в ячейку памяти по определённому адресу, выводить на экран данные из

ячейки с определённым адресом, а также выполнять программы на машинном языке, расположенные по определённым адресам.

Переходя в режим «BASIC», пользователь сообщает компьютеру, что теперь он использует язык «BASIC», интерпретатор которого уже встроен в компьютер. Именно на Basic концентрируется большинство мануалов, поскольку этот язык и должен, по задумке авторов, использоваться для работы с компьютером, ведь он обладает простым синтаксисом и встроенным набором команд, позволяющих легко выполнять базовые операции.

Сразу же отметим, что из режима «монитор» можно перейти в режим «ассемблер», в котором компьютер будет интерпретировать команды как ключевые слова языка ассемблера процессора MOS 6502. Но при этом сама по себе логика обращения к компьютеру практически не изменяется по отношению к коду «монитора», лишь команды становятся более понятными и расширяется их выбор. Таким образом, компьютер Apple II предлагает пользователю сразу три способа взаимодействия с ним:

1. Простые операции с адресами памяти с помощью «монитора».
2. Язык ассемблера, позволяющий также манипулировать с адресами памяти, но расширяющий спектр взаимодействия с ними.
3. Интерпретируемый язык «BASIC», в котором в качестве команд используются слова английского языка.

Разработчики рекомендуют использовать именно BASIC. С чем же это связано? Постараемся понять это, рассмотрев базовые операции, которые мы могли бы выполнять с помощью компьютера без использования сторонних программ.

Количество подобных операций, без углубления в составление комплексных программ, на самом деле весьма небольшое: изначально компьютер позволяет нам загружать программы со стороннего устройства, переключаться между режимами экрана (текстовый или графический) и

производить простейшие математические операции. Начнём с математических операций.

С одной стороны, их синтаксис, что для Basic, что для «монитора» идентичен – вписываем интересующую нас операцию, после чего нажимаем «Return» и на экран выводится результат. Но существует принципиальная разница.

Рассмотрим для начала команды «монитора». Прежде всего отметим, что они поддерживают исключительно сложение и вычитание. Если перебирать числа, можно заметить, что результат не всегда соответствует нашим ожиданиям, например, если ввести «9+7» результатом будет «10», но при этом, если складывать «10+7», результат будет привычный: «17».

Связано это с тем, что «монитор» поддерживает операции между числами в шестнадцатеричной системе, из-за чего и возникают «странные» результаты вычислений. Шестнадцатеричная система счисления используется по той причине, что машинные коды, как и адреса ячеек памяти и их содержимое всегда представлены исключительно в виде чисел в шестнадцатеричной системе счисления.

Basic же, напротив, работает с целыми десятичными числами. Но если мы попробуем просто вбить интересующую нас операцию, например, всё те же «9+7», мы получим ошибку. Для того чтобы получить результат на экране, нам необходимо явно указать, что мы хотим сделать с результатом вычислений, в данном случае, чтобы вывести его на экран, используется ключевое слово «PRINT». В результате, если ввести «PRINT 9+7» мы получим ожидаемый результат – 16.

Как уже отмечалось, Basic способен также выполнять деление и умножение, причём даже над отрицательными числами. Но в силу того, что работает он исключительно с целыми числами, если в результате деления получается дробное число, дробная часть просто отбрасывается, операций округления не происходит. Так, если мы будем делить числа меньше 10 на 10, результатом всегда будет 0.

То есть, на уровне математических вычислений, мы видим, что Basic ближе к обыденной жизни, позволяя выполнять обычные математические операции, в то время как «монитор» скорее полезен для решения технических задач, связанных с шестнадцатеричным счислением.

Переключение между режимами отображения экрана для обеих систем происходит примерно одинаково – для этого существует специальная команда. Принципиальная разница состоит в том, что для «монитора» командами для графического и текстового режима являются соответственно «C050» и «C051», а для Basic «GR» и «TEXT». В целом, принципиальной разницы нет, единственное, что команды Basic проще запомнить людям, которые не имеют опыта работы с машинными кодами.

Наконец, самые важные для нас на ранних этапах команды – запуск программы со стороннего носителя. Для того, чтобы выполнить программу, её необходимо предварительно загрузить с кассеты в ОЗУ компьютера, после чего её можно будет выполнить. В случае с Basic задача выполняется с помощью двух простых команд: «LOAD» для загрузки и «RUN» для запуска программы.

Для команд «монитора» ситуация в целом идентична, запуск программы всё также сводится к двум командам: «adrs1.adrs2R», записывающая программу с кассеты в ячейки памяти с номерами от «adrs1» до «adrs2», и «adrs1G», запускающая программу, начинающуюся с адреса «adrs1». Сложность здесь заключается в том, что если пользователь не знает, сколько именно ячеек памяти занимает программа на кассете, то он не сможет корректно загрузить её в память компьютера, и следовательно не сможет выполнить.

Именно здесь мы впервые сталкиваемся с очень интересным явлением, к которому мы ещё не раз вернёмся при разговоре о собственно написании программы. Назовём это «степенью контроля». В случае с загрузкой программы с кассеты с помощью монитора, компьютер предоставляет пользователю больше контроля над происходящим, ведь пользователь

прекрасно знает, сколько и каких ячеек памяти будет занимать загруженная программа. В теории это снижает вероятность различных непредвиденных ситуаций, компьютер приближается к простому проводнику, а не к посреднику, поскольку все его действия понятны пользователю. Но при этом увеличивается и число ошибок, которые может совершить сам человек, компьютер «снимает с себя ответственность» за те ситуации, когда ОЗУ не будет хватать, или окажется, что ячейки уже были заняты другой важной программой, которая теперь будет переписана.

При использовании же Basic, пользователь не знает, что именно делает компьютер, и не может контролировать его поведение. Функция машины как посредника становится более очевидной, и теперь все ошибки, которые могут случиться при записи с кассеты, «на её совести». Как мы увидим в дальнейшем, баланс между полным контролем над программой и передачей большей части решений собственно машине уже в 1970-ых годах будет являться одной из ключевых проблем, решаемых при программировании.

### *2.2.2 Сторонние программы для компьютера Apple II.*

Разобравшись со встроенным ПО и тем, как именно пользователь мог взаимодействовать с ним, мы можем начать «использовать» компьютер. Для этого рассмотрим ряд программ, которые были доступны пользователям в самом начале существования Apple II. Для каждой программы мы остановимся на том, как пользователь мог её получить, для чего она предназначена, и рассмотрим способы взаимодействия с ней.

#### *Дето кассеты*

Первые программы, с которыми сталкивался пользователь, купивший Apple II, были демонстрационные программы, записанные на кассеты, шедшие в комплекте с компьютером. Всего их было четыре и основной их задачей является показать возможности компьютера новому пользователю. Коротко рассмотрим каждую из них.



Первой кассетой является «Color Graphics Demo». Как видно из названия, задача этой программы состоит в демонстрации пользователю возможностей цветной графики, которая, как мы помним, являлась одной из главных «продающих» особенностей машины.

После запуска программа предлагает пользователю выбрать одну из 10 подпрограмм, которые показывают различные анимации, демонстрирующие возможности графического процессора. Представлены здесь как обычные цветные полосы, так и анимация рисования гиперболы. В целом, данная кассета призвана сразу предоставить пользователю возможность увидеть цветную графику.

Второй кассетой является «Color Math», выполняющая схожую функцию – продемонстрировать возможности цветной графики, но на этот раз не с помощью анимированных демо, а с помощью простой программы, принимающей от пользователя число и рисующее линию, чей цвет соответствует принятому числу.

Оставшиеся две демонстрационные программы являются играми – «Breakout» и «Pong». И та и другая – уже знакомые пользователям проекты для аркадных автоматов. Управляются они исключительно с помощью джойстика и, по факту, выполняют ту же функцию, что и предшествующие демки – демонстрируют возможности графического контроллера компьютера.

Таким образом, демо-кассеты в комплекте компьютера репрезентируют его как устройство больше предназначенное для игр, и акцентируют внимание на ключевой особенности машины – цветной графике. Теперь, когда мы ознакомились с демонстрационными программами, рассмотрим другие программы, которые были доступны первым пользователям компьютера.

### *Программы-утилиты*

В первую очередь остановимся на большой группе программ, которые условно можно назвать «программами-утилитами» - небольшие по объёму

программы, выполнявшие различные небольшие задачи, большая часть из которых в современной технике присутствует по умолчанию.

Главная отличительная черта таких программ состоит в том, что они распространялись не только в виде кассет, но и просто в виде кода, который пользователь самостоятельно записывал на пустую кассету для дальнейшего использования. Задачи, решаемые этими программами, весьма разнообразны: это и различные календари, и калькуляторы (позволяющие умножать или делить числа), и чек-листы, и различные конверторы.

В рамках программ-утилит возникают несколько подходов к организации «пользовательского интерфейса», которые затем будут использоваться и в более крупных программах.

Первый способ заключается в том, что после получения данных от пользователя, программа переходит в новое состояние, отображающее результат выполнения операции. Для таких программ характерны различные «режимы» работы, в которых они решают различные задачи. В другом варианте – экран условно делится на две части. Содержимое верхней части не изменяется, в то время как в нижней части отображаются результаты обработки данных пользователя.

Программы-утилиты реализованы как исключительно текстовые программы. Даже те, что работают с датами, не представляют собой полноценного календаря, а скорее являются программами, ориентированными на обработку полученных данных.

Например, существуют программы, считающие количество дней между двумя датами или вычисляющие, на какой день недели выпадает то или иное число.

Опять же, утилиты не приспособлены к сохранению данных или значительному расширению возможностей компьютера. Это именно небольшие программы-расширения, предоставляющие пользователю инструментальный для решения небольших повседневных задач. Это отражается и в том, что многие утилиты распространялись в виде кода – в



них не было ничего сложного, и пользователь вполне мог написать подобные расширения сам.

### *Программы для работы с текстом*

Сейчас трудно говорить наверняка, когда именно складывается рынок программного обеспечения для компьютеров Apple II. Тем не менее, большинство «серьёзных» программ, распространявшихся исключительно на коммерческой основе, появляются на рынке в 1979 году. В первую очередь мы обратимся к текстовым редакторам.

В 1979 в продажу поступают сразу два текстовых редактора: «Easy Writer», созданный Джоном Драпером и ставший первым текстовым редактором для Apple II, и «Apple Writer», написанный Полом Лутусом и выпускавшийся самой Apple Computer.

Буквально с самого старта программы демонстрируют совершенно различный подход к своей репрезентации – «Easy Writer» встречает пользователя цветной заставкой, после которой мы переходим к нескольким меню, связанным с персонализацией пользователя (вводим своё имя, указываем тип используемого нами принтера), после чего сразу переходим к редактированию текста. «Apple Writer» же сразу начинает с меню, из которого мы можем создать новый файл, загрузить файл с диска или распечатать текст.

Редактирование текста в обеих программах происходит схожим образом и отличается от того, к чему привыкли мы сейчас. Текст на экране всегда отображается в верхнем регистре, но если напечатать такой файл на принтере, текст окажется набранным «маленькими» буквами. Для того же, чтобы переключаться между регистрами, необходимо нажать «Esc», после чего следующий введённый символ, выведенный на экран, будет инверсивным.

Ещё одной особенностью, которой обладает только «Easy Writer», является возможность свободно перемещать курсор по тексту с помощью стрелочек (обычно при взаимодействии с компьютером Apple II мы не можем

перемещаться между отдельными строками). Для того чтобы также перемещать курсор в «Apple Writer» - необходимо перейти в специальный режим с помощью двойного нажатия клавиши «Esc».

Говоря о текстовых редакторах, мы сталкиваемся с делегированием компьютеру особой задачи – создание печатного текста, - которая до этого решалась с помощью других инструментов (печатной машинки). Помимо того, что эта передача становится в принципе возможной за счёт подключения нескольких плагинов, в данном случае – специальной программы и принтера, отметим ещё несколько интересных особенностей.

Во-первых, здесь мы впервые сталкиваемся с существованием особого промежуточного результата – текста на экране. В случае работы с печатной машинкой текст сразу же попадает на физический носитель – пользователь видит конечный результат, уже трансформированный устройством. При работе с компьютером набираемый текст «создаётся» в нескольких вариантах.

Когда пользователь нажимает на клавишу с определённым знаком – в оперативную память его компьютера записывается информация, отражающая положение этого знака в тексте и его регистр. Это первый вариант существования текста, который и является основным для компьютера, именно из него он создаёт два других варианта существования текста – изображение на мониторе и текст, распечатанный на принтере.

Набирая текст с помощью Apple II – пользователь внешне работает именно с текстом, который отображается на экране. Он видит именно его, редактирует именно его. Но конечным результатом и целью является именно то, что будет напечатано на бумаге – и этот конечный результат выглядит совершенно не так, как текст, выводимый на экран.

Если до этого мы видели компьютер в качестве инструмента, непосредственно выполняющего определённую задачу, то теперь он оказывается в роли проводника, ведь конечный результат уже не принадлежит собственно компьютеру или цифровой технике в принципе.

Текст на экране не удобен для чтения в силу слабой различимости регистров, усложнённой навигации по нему и небольшого количества строк. Текст в виде двузначных чисел (каковым он и сохраняется в памяти) вообще может быть прочитан пользователем только через ряд дополнительных сложных операций. И только напечатанный принтером текст выступает в качестве результата работы, уже не относящегося к компьютерному миру.

Во-вторых, уже здесь мы чётко прослеживаем, как определение персонального компьютера Apple II начинает растворяться. Является ли компьютер инструментом для работы с текстом? Безусловно, да, ведь подключив к нему несколько плагинов, мы можем с его помощью создавать тексты и даже редактировать их, что затруднено при использовании других инструментов, с помощью которых мы решаем схожие задачи.

Но использовать Apple II как инструмент для редактирования текстов мы, как рядовой пользователь, не обладающий навыками в программировании, смогли только через два года после появления компьютера. До 1979 года у нас этой возможности не было. Да и после 1979 года мы можем получить различный опыт редактирования текста на компьютере Apple II.

Если мы используем «Easy Writer», то этот опыт будет одним – мы сможем сразу перемещаться по тексту, без особых проблем вводить несколько букв в верхнем регистре, а при каждом запуске программы наблюдать заставку больше характерную для видеоигры. Работая же с «Apple Writer», мы сталкиваемся с более «сдержанной» программой, в которой даже навигация по тексту требует дополнительных усилий.

То есть, можно сказать, что персональный компьютер как актер уже сейчас позиционируется как нечто динамичное, изменяющееся не только с ходом времени, но и в целом отличающееся от аппарата к аппарату.

#### *Программы для бизнеса (офисные программы)*

Apple II по большей части ориентировался на людей, которые будут использовать компьютер дома в личных целях, однако, в последствие, он

становится одной из первых машин, которая начинает активно использоваться в офисах. В основном это связано с тем, что для компьютера было создано несколько специфических программ, ориентированных на экономические расчёты.

Первыми такими программами становятся различные утилиты, способные выполнять расчёты с денежными данными и сохранять в памяти кассеты небольшой объём данных. Сами по себе они позиционировались как программы для ведения семейного бюджета и мало отличались от уже рассмотренных программ. Тем не менее, они демонстрировали принципиальную возможность использования Apple II в бизнесе.

Первой программой, ориентировавшейся в первую очередь на офисных работников, становится «VisiCalc» 1979 года. И она сразу же становится совершенно уникальным представителем на рынке ПО для персональных компьютеров.

«VisiCalc» является программой для работы с табличными данными. Запустив её, пользователь получает экран, разбитый на столбцы и строки, в которые он может вводить определённые значения. Особый интерес VisiCalc представляет, поскольку он не просто предоставляет удобный интерфейс для работы с данными, но и позволяет их обрабатывать – рассчитывать значение одной ячейки в зависимости от значений в другой. Причём для расчёта нового значения программа использует не только стандартные арифметические операторы, но и собственные функции.

Мы не будем подробно останавливаться на способах взаимодействия с программой и навигации по таблице. Отметим лишь, что происходят они с помощью специальных комбинаций клавиш, которые необходимо запоминать. Гораздо интересней здесь именно наличие собственного «языка программирования».

Во-первых, таблицы «VisiCalc» принципиально отличаются от рассмотренных выше текстовых редакторов, поскольку эти таблицы и есть результат работы. Мы впервые сталкиваемся с программой, работающей

исключительно с цифровым форматом данных, которые доступны пользователю исключительно в виде своей интерпретации на экране.

Да, таблицы можно распечатать, программа предоставляет нам такую возможность, но, в отличие от текста, который на экране попросту неудобно читаем, таблица обладает одним уникальным свойством, которым не обладает бумага – он способна показать связи между своими элементами. Ведь если в ячейке записана функция, то на экране будет отображаться результат вычислений. Но если выбрать ячейку, в специально выделенной строчке будет написана используемая функция.

Во-вторых, создавая собственные функции «VisiCalc», по сути, формирует особую среду, максимально приближенную к среде самого компьютера – мы вновь приходим к тому, что с помощью специальных команд занимаемся обработкой данных.

То есть, понятие самого компьютера вновь размывается. Ведь с помощью плагинов он не просто способен перенять на себя новую функцию, плагин на время подменяет собой компьютер. Учитывая то, что одновременно на одном компьютере Apple II пользователь может запустить ровно одну программу – это действительно так.

Представим себе офисный рабочий компьютер, на котором запущен VisiCalc и ничего кроме него никогда не запускалось. По сути, пред нами совершенно иной актер, с которым конечный пользователь может взаимодействовать, даже не подозревая о том, что это компьютер. Ведь в этом случае – это скорее машина для работы с таблицами, не более и не менее.

Но даже если «VisiCalc» способен полностью подменить собой Apple II, он вынужденно построен так, что учитывает платформу. Это влияние компьютера на программу лежит на поверхности и не требует особого погружения в код – оно проявлено в навигации по ячейкам, учитывающей, например, не способность компьютера перескакивать между строчками, и в форме функций, использующихся в программе.



Синтаксис функция «VisiCalc» напоминает синтаксис функций Basic, который является основным языком взаимодействия с компьютером для рядового пользователя. Именно поэтому программа не вытесняет собой полностью платформу, но при этом перенимает на себя большую часть действий, связанных собственно с выполнением поставленной задачи.

### *Обучающие программы*

Коротко остановимся на обучающих программах, целью которых было развить определённые навыки у пользователя. Обучающие программы представляют собой весьма интересный рынок программного обеспечения, поскольку именно через них, по сути, осуществляется приобщение детей к компьютерам.

Большинство из них построено по принципам близким к викторинам – на экране возникает вопрос, относящийся к какой-либо области знания (самые распространённые – это арифметика и знание языка), после чего пользователь вводит правильный ответ. Одним из ярких примеров подобных программ является «Compu-Read» 1979 года.

Программа представляет собой четыре типа викторин, тренирующих навыки чтения на английском языке – ввод по памяти букв, подбор синонимов и антонимов к словам, ответ на простые вопросы по предложениям. Программа управляется очень просто и, по сути, ничем кардинально не отличается от уже рассмотренных выше программ.

### *Программы для работы с графикой*

Как мы помним, Apple II предоставлял пользователю возможность выводить на монитор изображения различной цветности и разрешения. В 1970-ых годах программы для работы с графикой ещё были новинку, и для вывода изображения на экран его было необходимо «запрограммировать», что мы уже видели на примере демокассет. Тем не менее, для пользователей, не знакомых с программированием на Basic, уже в это время были созданы отдельные программы, позволяющие обойти данное ограничение.

Одной из таких программ является «Shape Builder» 1979 года. Программа крайне небольшая и распространялась в виде исходного кода. Тем не менее – это очень интересный плагин.

«Shape Builder» предоставляет пользователю возможность рисовать на экране, используя клавиши, заполняя «пиксели» экрана определённым цветом. Тем не менее, сохранить или распечатать результат нельзя, хотя это возможно, если дополнить исходный код программы.

В случае с этой программой мы сталкиваемся с двумя интересными особенностями. Во-первых, «Shape Builder» - это, пожалуй, первый пример плагина, нацеленного на упрощение выполнения операций с компьютером. Ведь, по сути, Apple II позволяет пользователю рисовать на экране – для этого можно использовать команды Basic. По сути «Shape Builder» предлагает лишь замену для этих команд, в отличие от уже рассмотренных программ, нацеленных на полное изменение задач, выполняемых компьютером.

То есть мы сталкиваемся с тем, что Apple II как актор может подстраиваться под пользователя. Для него возможно создание плагинов, изменяющих лишь способ решения задач, которые и без того могли быть делегированы компьютеру.

Во-вторых, «Shape Builder», в каком-то смысле, «недоделанный» плагин. Он является элементом, который пользователь затем может встроить в другое ПО, расширив возможности последнего. Таким образом, вокруг компьютера начинает формироваться специфическая среда, в которой плагины начинают дополнять друг друга, включаться один в другой. Это очень важное качество, которое очень сильно повлияет на социальную составляющую компьютера.

Другой интересной программой, схожей с «Shape Builder», является «Apple World», также созданная в 1979 году. Особый интерес она представляет, поскольку позволяет работать с трёхмерной графикой.

В конце 1970-ых годов для воспроизведения трёхмерной графики на компьютерах, как правило, использовалась векторная графика и специальные мониторы, способные воспроизводить её, не тратя значительные ресурсы на предварительную растеризацию. Именно поэтому появление в 1979 году программы, способной просматривать трёхмерные изображения, является весьма неожиданным.

Именно поэтому «Apple World» - очень важная программа. Она демонстрирует, пожалуй, одну из главных особенностей компьютера как актора – несмотря на наличие явных ограничений, они могут быть уничтожены. Изначально Apple II не предназначен для работы с трёхмерными объектами – все его графические команды направлены на создание двумерной пиксельной графики. Тем не менее, программа может быть написана так, чтобы извлечь максимум из существующей системы, из-за чего появление «Apple World» не является чем-то неожиданным. Это естественное следствие из природы самого актора.

### *Игры для компьютера Apple II*

Как мы помним, среди демокассет для компьютера находились сразу две игры, перенесённые на компьютер с игровых автоматов. В целом, игры, как внешние программы являются важной части библиотеки приложений для Apple II.

Это может показаться странным, но именно игры сейчас составляют подавляющее большинство среди приложений, «сохранившихся» до нашего времени. Более того, если «серьёзные программы» возникают преимущественно в 1979 году, то первые игры появляются уже со старта продаж компьютера. С чем же это связано?

На наш взгляд, на формирование весьма богатой игровой библиотеки компьютера влияет два важных фактора. Во-первых, Apple II появляется в переломный момент, когда активно складывается практика игры в домашних условиях. Как уже упоминалось, в том же году, что и персональный компьютер, выйдет успешная игровая консоль Atari 2600. То есть в это время

существовал определённый запрос на домашние игры, которые компьютер мог удовлетворить.

Во-вторых, в отличие от обычных программ, компьютерные игры уже существовали к моменту запуска компьютера. Рассмотренные ранее программы попросту не существовали на крупных ЭВМ – они были рассчитаны на работу в исследовательских центрах и университетах, поэтому большая часть программ, написанных для них, ориентировались на обработку различных статистических данных или на работу со сложными математическими формулами. Но и первые компьютерные игры разрабатывались на этих машинах.

Если для возникновения других программ требовались дополнительные усилия – придумать концепцию программы, её интерфейс и вообще понять, нужна ли она конечному пользователю, то перед разработчиками игр таких задач не стояло. Ведь игры уже разработаны, они уже существуют – нужно лишь переписать уже существующий код на одном из языков, который поддерживает компьютер.

Здесь мы сталкиваемся с интересной особенностью не только персональных компьютеров, но и вообще цифровой техники – понятием «портируемости». На примере видеоигр мы видим, как формируются своеобразные «универсальные плагины», которые могут быть присоединены к самым разным машинам. Безусловно, это не одна и та же программа – формально, это различные плагины, но выполняющие одну и ту же функцию.

Сам по себе вопрос портирования требует отдельного исследования, что не входит в задачи данной работы. Сейчас важно отметить, что персональный компьютер формируется как актер не в отрыве от других устройств, используя исключительно те плагины, которые были уникальными для него, но и те, которые изначально разрабатывались для иных устройств.

Библиотека игр для Apple II в 1970-ых годах представляла собой в основном подобные «универсальные плагины» - это были клоны аркадных

игр и различных текстовых игр, написанных для крупных ЭВМ (например, «Namurabi»). При этом появляются и отдельные игры, не воспроизводящие на компьютере опыт, которые дают уже существующие игры, но ориентирующиеся на них. Примером подобного проекта является игра «Adventure land», ориентирующаяся на «Adventure», но не копирующая её дословно.

Таким образом, персональный компьютер в 1970-ых годах выступает не только как актор, которому делегируются различные вычислительные операции, но и как одно из устройств, переносящих игровой опыт в домашние условия. Более того, эта функция, как мы помним, изначально была заложена в компьютер – специально для этого в него была включена цветная графика и добавлены игровые контроллеры, позволяющие максимально точно воспроизвести опыт, предоставляемый аркадными игровыми автоматами.

#### *Программы для программирования*

Прежде чем перейти к следующему разделу работы, в котором мы рассмотрим основные практики программирования, остановимся на специальных программах, создававшихся для облегчения программирования на компьютере Apple II. В целом, эти программы можно дополнительно разделить на две большие группы:

1. Утилиты, облегчающие программирование.
2. Программы, позволяющие писать на дополнительных языках программирования.

Первая группа появляется весьма рано. Это программы, связанные с облегчением написания кода на Basic и ассемблере MOS 6502. Они добавляли автоматическую проверку кода, дополнительные компиляторы (например, Lisa – ведущая программа для перевода ассемблера в машинный код) и даже дополнительные функции для Basic. Некоторые из этих программ являются уже известными нам «дополнительными плагинами»,

которые включались в уже написанный код, расширяя именно его возможности, а не возможности собственно компьютера.

Вторая группа программ позволяла программистам писать программы на языках, изначально не встроенных в сам персональный компьютер. В первую очередь – это Fortran и диалекты Basic, отличные от того, что был изначально встроен в RAM компьютера. По сути, эти программы являются компиляторами, переводящими команды на языке программирования в машинный код, понятный Apple II.

В случае с программами второй группы мы сталкиваемся с весьма интересным типом плагина – своеобразным «теневым плагином». Эти программы максимально незаметно расширяют возможности компьютера. По сути, эти плагины практически не видны пользователю и очень мало влияют на способ взаимодействия с компьютером, позволяя скорее подстроить его поведение под нужды пользователя.

Ещё одной интересной особенностью программ из этой группы является тот факт, что они, по сути, являются программами, существующими для написания других программ. Они практически не влияют на функционирование самого компьютера, косвенно определяя лишь вид плагинов, созданных с их помощью.

### *2.2.3 Практики программирования на компьютере Apple II*

После того, как мы рассмотрели сторонние программы, используемые для расширения возможностей компьютера Apple II – обратимся теперь собственно к практикам создания программ. Как уже упоминалось выше – написание собственных приложений является одним из основных способов взаимодействия с Apple II в принципе, и, формально, с самого начала использования нами компьютера мы пишем небольшие программы. Тем не менее, собственно программирование представляет собой отдельную практику, являющуюся ещё одной задачей, которая может быть делегирована компьютеру.

Как уже было сказано, для Apple II «ведущими» являются два языка – ассемблер для MOS 6502 и Basic. Разработчиком подразумевается, что большая часть программ будет писаться именно на Basic – об этом говорит, прежде всего, количество литературы и тот факт, что листинги большинства программ для компьютера составлены именно на Basic. К тому же большинство литературы по ассемблеру, издававшейся в конце 1970-ых годов, подразумевает, что пользователь уже знаком с программированием и изучает ассемблер в качестве второго языка. Именно поэтому, в данной части работы мы сначала обратимся к тому, как строятся программы на Basic, а затем обратимся к ассемблеру.

### *Программирование на Basic*

Basic, строго говоря, является семейством языков программирования, для которых характерен простой синтаксис и минимальное число используемых операторов. Компьютер Apple II использовал собственный диалект Basic – Integer Basic. Особенностью этого диалекта является включение в язык программирования команд характерных именно для компьютера (необходимых, прежде всего, для работы с графикой). К концу 1970-ых годов на смену Integer Basic приходит Applesoft Basic, отличавшийся большим количеством команд.

Первое, что необходимо отметить – это то, что Basic является интерпретируемым языком программирования. Это значит, что программа не компилируется в машинный код перед выполнением, но выполняется строка за строкой. Подобный подход обеспечивает довольно высокую скорость выполнения кода, поскольку не требует времени на компиляцию. С другой стороны – если на устройстве отсутствует нужный интерпретатор, программа не будет выполнена. Поэтому программируя на Basic – мы создаём программу исключительно для компьютера Apple II.

Программа, написанная на Basic, организуется с помощью нумерации строк – каждая команда располагается на собственной строке. При интерпретации команды выполняются в порядке увеличения номера строки,

при этом нумерация не обязательно должна быть последовательной – можно пропускать сколько угодно строк. Нумерация строк очень важна, поскольку именно на ней базируется главный приём программирования на Basic – оператор GOTO, который заставляет интерпретатор перейти к выполнению определённой строки.

В сочетании с уже разработанными к тому моменту операторами логического ветвления IF ELSE и оператором цикла FOR, оператор GOTO предоставляет программисту возможность написать программу любой сложности. Но здесь важно понимать, что весь написанный код выполняется последовательно, в связи, с чем возникает несколько принципиальных подходов к организации кода, среди которых можно явно выделить два.

Первый заключается в том, чтобы писать программу максимально приближенной к естественному алгоритму решения задачи – приводя последовательно команды, решающие различные подзадачи. Подобные программы относительно легко читаются человеком со стороны, но сложно исправляются, поскольку в них сложно выделить отдельные блоки, которые необходимо изменить.

Второй подход организует программу как состоящую из отдельных блоков, к которым отсылают операторы GOTO. В этом случае программа чётко делится на отдельные блоки, но весьма сложна для чтения программистом, не участвовавшим в непосредственной разработке кода.

Особенностью Basic является возможность использовать специальные команды для отображения графики. В первую очередь – это три команды: PLOT, VLIN, HLIN, рисующие, соответственно, точку, вертикальную и горизонтальную линию. Благодаря их использованию, отображение графики с помощью Basic является весьма простым, но требует предварительной работы над рисунком, чтобы разбить его на соответствующие примитивы и проработать алгоритм, который бы смог оптимальным образом вывести изображение на экран.



Ещё одной важной командой Basic является команда POKE, позволяющая получить доступ к определённой ячейке памяти. Она используется для создания программ, работающих со сторонними ресурсами, хранящимися в памяти программы, или для тех программ, что могут сохранять или загружать данные из памяти. Рядовым пользователем такие программы, как правило, не создавались, поскольку работа с конкретными адресами памяти сопряжена с риском критических ошибок, оказывающих серьёзное влияние на работоспособность самого компьютера.

Программы на Basic в процессе их написания довольно сложно редактировать, поскольку, как мы помним, переходить между строками довольно сложно, необходимо предварительно иметь готовый алгоритм программы, согласно которому уже и будет писаться новый код. При этом ошибки в отдельных строках могут быть обнаружены уже самим интерпретатором ещё на этапе написания кода, что позволит переписать конкретную строку.

Таким образом, программирование на Basic является наиболее простым и естественным путём написания программ именно для Apple II. Как уже отмечалось выше, на других платформах такой код не запустится в принципе, но в пользу использования его именно для данного компьютера говорит и наличие уже рассмотренных выше специальных программ.

Сам по себе Integer Basic является очень интересной программой. С одной стороны – он изначально включён в компьютер, является его составной частью на уровне программы. С его помощью могут выполняться сразу две задачи – собственно базовое взаимодействие с компьютером и создание новых программ для него.

С другой стороны – он всё ещё является плагином, пристраивающимся к компьютеру. Это отдельная программа, встроенная в RAM компьютера, но которую можно заменить другой, как это в последствии и случится с Applesoft Basic, являющимся иным плагином, но выполняющим схожие функции. То есть, Integer Basic не является чем-то априори присущим Apple

II. Компьютер может использовать и другие программы, как это и происходило с рассмотренными ранее внешними программами.

### *Язык ассемблера процессора MOS 6502*

Вторым языком программирования, который изначально встроен в Apple II, является язык ассемблера MOS 6502. Отметим, что язык ассемблера не идентичен командам монитора, о которых мы говорили выше. В каком-то смысле, монитор является расширением ассемблера, включая в себя специфические команды для взаимодействия с системами компьютера Apple II, в то время как ассемблер может использоваться для написания программ для любого устройства, в основе которого лежит процессор MOS 6502.

Ассемблер – компилируемый язык. Программа, написанная на нём, перед выполнением должна быть переведена в машинный код. Эта операция производится перед записью кода на запоминающее устройство, в результате чего на кассете хранятся уже машинные команды, которые и выполняет процессор. Этот процесс получил название «ассемблирования» и может выполняться как специальными программами (для Apple II самой распространённой является Lisa), так и вручную самим программистом.

Код на языке ассемблера выглядит совершенно не так, как код на Basic. Вместо интуитивно понятных команд на английском языке, ассемблер использует коды, состоящие из трёх букв и шестнадцатеричные адреса памяти и числа. Тем не менее, сразу отметим, что сам принцип построения программы кардинально не отличается. Ассемблер всё так же поддерживает различные операторы ветвления (в отличие от IF Basic – это именно операторы ветвления, которые переходят к выполнению определённого сегмента программы, если условие выполняется), оператор безусловного перехода (JMP) и циклы (создаваемые искусственно с помощью операторов ветвления).

Ключевое отличие кроется в том, что ассемблер работает с ячейками памяти и данными в них. Даже команды ассемблера, по сути, являются данными, записываемыми в определённые ячейки памяти, из-за чего

операторы перехода принимают в качестве аргумента не строку с командой, как это было в случае с Basic, а адрес ячейки памяти, хранящий нужную команду.

Операции на таком низком уровне значительно усложняют логику самих команд. Если общий принцип построения программы и остаётся неизменным, то многие действия, осуществляемые на Basic парой команд, для ассемблера превращаются в отдельную подпрограмму. Классическим примером является умножение двух чисел.

Если на Basic нам достаточно написать: «PRINT A \* B» (при условии, что переменным A и B присвоены значения), то для ассемблера необходима отдельная программа:

```
«LDA #0
LDX #$8
LSR A
loop:
BCC no_add
CLC
ADC B
no_add:
ROR
ROR A
DEX
BNEloop
STA B»
```

Более того, это лишь одно из возможных решений, и даже не самое лучшее в плане производительности (поскольку вычисления могут занять больше 100 циклов), но самое простое для понимания программистом.

Что же даёт программирование на ассемблере? Во-первых, именно ассемблер используется для написания сложных программ, требующих высокой производительности – программа, переведённая в машинный код,

работает быстрее, чем Basic, а полный контроль над памятью позволяет оптимизировать код таким образом, чтобы он потреблял как можно меньше ресурсов. А, во-вторых, программы, написанные на ассемблере, могут выполняться не только на компьютерах Apple II. И вот это самый важный для нас момент.

В отличие от Integer Basic ассемблер не является чем-то сторонним для Apple II – это язык, на основе которого работает процессор компьютера, являющийся частью его аппаратного уровня, то есть – это неотъемлемый элемент компьютера, а не очередной плагин. Но процессор MOS 6502 использовался не только в компьютере Apple II – вышедшая в том же году игровая консоль Atari 2600 также была построена на основе этого процессора и программы для неё были написаны также на языке ассемблера.

Так есть ли какая-то разница между программами для этих двух машин, написанных на ассемблере? Безусловно, есть – ведь помимо стандартных команд ассемблера в программах будет использован специфический для платформы код, заключающийся в обращении к отдельным адресам памяти, в которых хранятся, например, команды для вывода графики на экран. И эти команды будут принципиально различными, поскольку Apple II, как мы помним, разбивает экран на пиксели, в то время как Atari 2600 работает непосредственно с устройством телевизора и выводит изображение в соответствии с отдельными циклами.

Более того, Apple II предоставляет пользователю возможность создавать программы для MOS 6502, а Atari 2600 такой возможности не даёт. То есть сам по себе процессор ещё ничего не определяет, но именно процессор, с другой стороны, позволяет объединить несколько устройств в одну группу, поскольку они работают с программами, написанными на определённом языке программирования.

Сама по себе практика написания программ на ассемблере предоставляет пользователю уникальную возможность создания программы, которая рассчитана не на расширение возможностей его собственного

компьютера, но для запуска на устройстве, которое может кардинально отличаться от персонального компьютера. Всего лишь одна общая деталь позволяет связать вместе артефакты, кардинально отличающиеся друг от друга, и сделать их развитие взаимозависимым.

Ассемблер будет являться ведущим языком для разработки ПО не для персональных компьютеров вплоть до создания достаточно мощных компиляторов. Это язык программирования, парадоксальным образом максимально близкий к самому компьютеру и в то же время максимально далёкий от него.

Программный уровень является уровнем, на котором возникает большинство специфических черт, характерных для персонального компьютера как актора, определяющим специфику как функционирования данного артефакта, так и особенности интегрируемых в него плагинов. На данном этапе, мы можем с уверенностью сказать, что Apple II как артефакт постоянно ускользает от точного определения своей сути, становясь своеобразной пустотой, на место которой способен встать новый актер, конструируемый программой, запущенной на устройстве.

При этом, если мы будем спускаться на самый низкий уровень, мы обнаружим, что компьютер теперь напротив является чем-то необычайно большим – он оказывается одной из вариаций устройств, обладающих схожими характеристиками, сходной аппаратной и программной составляющей. Тем не менее, прежде чем прийти к окончательным выводам – рассмотрим последний выделенный нами уровень – социальную составляющую артефакта.

### **2.3 Социальная составляющая компьютера Apple II**

Компьютер Apple II возник в среде, в которой цифровая техника только набирала популярность и вес в широком потребительском кругу. По сути, до появления этого артефакта, компьютер был чем-то существовавшим

исключительно в среде людей, обладающих специальными знаниями. С другой стороны, сама по себе эта сфера была довольно развита – существовали разнообразные объединения любителей компьютерной техники, занимавшиеся разработкой собственных любительских машин, написанием для них программ и распространением их между компьютерами. Не последнюю роль в формировании этих сообществ играли университеты, в которых формировались отдельные группы вокруг стационарных ЭВМ, установленных в них.

Именно поэтому появление Apple II не прошло незамеченным. Он привлек к себе внимание тех групп, которые уже были вовлечены в использование компьютеров, был включён в те системы, которые были созданы ими. На их основе в последствие создавались схожие сети, специализировавшиеся исключительно на компьютере Apple II.

В данном параграфе мы рассмотрим два вида сетей, занятых в распространении компьютера Apple II как артефакта и, в целом, повлиявших на его становление как актора. Это различные компьютерные журналы и сообщества, при этом важно понимать, что очень часто они оказываются связанными друг с другом – сообщества выпускают собственные журналы.

### *2.3.1 Компьютерные журналы и компьютер Apple II*

Компьютерные журналы как явление существовали ещё с начала компьютерной эры в 1950-ых годах. Тем не менее, они были ориентированы преимущественно на профессиональную аудиторию и состояли преимущественно из статей, посвящённых архитектуре компьютеров. Поддерживали эту традицию журналы вплоть до середины 1970-ых годов, постепенно уводя курс в сторону публикации статей, касающихся архитектуры компьютеров для домашнего использования – то есть нацеливаясь на широкую аудиторию профессионалов и полупрофессионалов, способных с помощью представленных схем собрать собственный персональный компьютер.

В 1976 году начинает выходить журнал «Dr Dobb's Journal», который становится первым журналом, ориентированным большей частью на программистов. Он был посвящён вопросам создания приложений для компьютеров, содержал материалы о принципах программирования и коды конкретных программ.

Таким образом, к появлению Apple II уже существовала сформированная база компьютерных журналов – их было много, но ориентировались они преимущественно на хардварный уровень. При этом аудиторией этих журналов были люди, обладающие специальными навыками, понимающие базовые принципы программирования и знакомые с архитектурой вычислительных машин.

С выходом Apple II на широкий рынок возникает два «вида» журналов, принципиально отличающихся от того, что можно было купить до этого:

1. Журналы, ориентирующиеся не на профессионалов.
2. Журналы, ориентирующиеся исключительно на владельцев Apple II.

Уже существовавшие журналы, как правило, мало освещали конкретные модели. Так, Apple II непосредственно упоминается в них в качестве «рекламы» - коротких информационных сообщениях на один разворот, в которых описываются основные характеристики машины, позиционирующейся как компьютер для домашнего использования.

Исключением является статья в одном из номеров журнала «Byte», где приводится развёрнутое интервью со Стивом Возняком, касающееся архитектуры нового компьютера. Данная статья представляет собой нечто среднее между материалом, нацеленным исключительно на профессионалов, и статьёй для новичков. С одной стороны, Возняк использует простой и понятный язык, в некоторых местах объясняя совсем уж базовые вещи (типа назначения процессора). При этом статья наполнена деталями, которые могут быть интересны и понятны исключительно специалистам-инженерам.

Журналы, сконцентрированные на программировании, также редко выпускали материалы, посвящённые именно данному компьютеру. Даже

статьи, посвящённые работе с Basic, в них редкость. Чаще всего в них можно встретить статьи, посвящённые языку ассемблера и приёмам написания на нём программ различной сложности.

Важно отметить, что программы на ассемблере, представленные в этих журналах, носят очень интересный характер – это программы, нацеленные на то, чтобы «подстроить компьютер под себя». Чаще всего можно встретить статьи, в которых приводятся примеры написания собственного интерпретатора Basic или другого языка программирования более высокого уровня, чем ассемблер. Это ещё раз говорит нам о специфической «встроенности» Integer Basic в компьютер, о чём мы уже упоминали выше.

Интересно, что журналы, ориентировавшиеся на непрофессиональную аудиторию, в строгом смысле в 1970-ых годах так и не появились. Их место занимали различные рекламные брошюры и каталоги крупных торговых центров.

В этих изданиях приводилась самая базовая информация, касающаяся компьютера Apple II, а акцент делался на его относительной дешевизне и возможности использовать дома для самых разных целей. Естественно, никакой информации, касающейся использования компьютера в этих «журналах» найти было нельзя, но они выполняли свою основную функцию – распространяли компьютер среди обывателей.

Здесь интересно, что к концу 1970-ых годов изменяется логика представления компьютера обывателю. Если в самом начале выхода машины на рынок, его позиционировали как устройство для организации досуга, ориентированное на использование в кругу семьи, то теперь Apple II – это компьютер для бизнеса, который можно использовать для работы. Скорее всего, это связано с расширением списка приложений для компьютера, ориентированных на решение прикладных задач в сфере экономики.

Наконец, журналы, издававшиеся для пользователей Apple II, как это ни странно также не были широко представлены в то время. Они делятся на две категории:



1. Журналы с ответами на письма пользователей.
2. Журналы, посвящённые программам, выпущенным для компьютера Apple II.

Первая группа представлена рядом изданий, выпускавшихся самой AppleComputer. В этих журналах печатались ответы на вопросы пользователей, их предложения и рекомендации. Чаще всего это касалось именно программной составляющей – как встроенных в Apple II (типа рекомендаций по программированию на Basic), так и изданных другими компаниями. Интересно, что в этих журналах часто печатались различные дополнения и улучшения к уже существующим программам, позволявшие пользователю улучшить их работу или избежать некоторых критических ошибок.

Вторая группа журналов, очень близка к рекламным буклетам. Как правило, такие издания представляют собой несколько статей, посвящённых отдельным программам, которые можно или заказать по почте и получить кассету, или воспроизвести самим, ориентируясь на предложенный код. В единственном случае, мы даже обнаружили пример, в котором вместо кода программы пользователю предлагалось устное описание действий, которые происходят на той или иной строке кода.

Таким образом, журналы выполняли весьма конкретную функцию – они распространяли информацию о самом компьютере и программах, которые могут быть на нём запущены. По сути, именно в них формировался некий образ «Apple II», на основе которого пользователь и решал, покупать компьютер или нет. Также именно с помощью журналов многие люди могли взаимодействовать с купленным устройством, поскольку именно в них можно было найти и примеры программ, и различные рекомендации по работе с самой «основой» компьютера.

Журналы ориентировались как на профессионалов, так и на тех, кто только начинал свой путь в мире цифровой техники. Они также помогали общаться людям между собой, обратиться непосредственно к разработчикам,

как через письма, так и через статьи, написанные ими. Эта функция собирания сообщества, тем не менее, не была ведущей для журнала, в большей степени она присуща различным группам владельцев компьютера.

### *2.3.2 Сообщества, формирующиеся вокруг компьютера Apple II*

С начала 1970-ых годов в США были весьма распространены компьютерные клубы. Представляли они собой объединения людей, связанных с цифровой техникой, которые собирались раз в неделю и обсуждали различные вопросы, связанные с новинками в интересующей их области. Такие собрания не являлись специализированными или профессиональными конференциями и состояли, как правило, из студентов и молодых специалистов, работавших на существовавшие тогда компьютерные компании (IBM, Atari и т.д.).

В этом виде клубы не представляли собой серьезной «силы», влияющей на развитие техники. По сути, это были небольшие сообщества, главной целью которых было создание и поддержка собственных любительских машин.

Именно на собрании такого клуба впервые был представлен компьютер Apple I, ставший прототипом Apple II. Важность этих клубов заключается в создании живой среды, в которой осуществлялось активное взаимодействие между людьми с разной степенью подготовки, благодаря чему тот же Apple II во многом и возник в том виде, в каком мы его знаем.

Опять же, с появлением серийно производимого компьютера происходят существенные изменения в структуре таких сообществ. Во-первых, теперь они включают в себя и людей, которые совсем мало знакомы с компьютерами, являющимися рядовыми обывателями, купившими себе новый компьютер. Во-вторых, теперь эти сообщества формируются из людей, использующих одну и ту же машину, а потому изменяется спектр обсуждаемых вопросов, если раньше они были сосредоточены преимущественно на аппаратной составляющей, то теперь акцент переходит

на программное обеспечение, поскольку именно оно является уровнем, на котором большинство людей взаимодействуют с компьютером.

Особый интерес представляет сообщество «A.P.L.L.E.» - старейшее объединение владельцев компьютеров Apple II, существующее и по сей день. Самым интересным здесь является его масштаб – за счёт выпуска собственного журнала данное общество могло взаимодействовать с людьми в самых разных уголках страны, чего не могли себе позволить локальные объединения.

По сути, A.P.L.L.E. становится сообществом, целью которого является организация взаимодействия человека и машины, а также собирание людей в новую особую группу. Она, как и другие подобные группы способствует созданию особого вида идентичности – «владелец компьютера Apple II», формирует представления о том, какими навыками должен обладать такой человек и какие задачи он способен решать с помощью своего компьютера.

При этом группа обладает собственной иерархией, отражающей уровень владения компьютером. Те, кто в этой иерархии расположен выше, могут оценивать программы и предложения лиц, расположенных на нижних ступенях, давать советы и рекомендации. Но, как и любая иерархия в подобных сообществах – она условна и может быть моментально разрушена, если условный новичок предложит программу, превосходящую по своим качествам уже существующую.

Таким образом, различные сообщества и компьютерные клубы способствовали, как это ни странно, преобразованию человека, превращению его в единицу, способную взаимодействовать с компьютером. Также они формировали и некий идеальный образ компьютера, способный увеличить количество людей, готовых вступить во взаимодействие с новым для них артефактом.

## ЗАКЛЮЧЕНИЕ

В результате проделанной работы были определены основные характеристики персонального компьютера как актора на ранних этапах его становления в США 1970-ых годов. В качестве примера был выбран компьютер модели AppleII как самый распространённый в тот период и наиболее ярко отражающий те тенденции, которые можно проследить и на примере других компьютеров. Были выделены следующие характеристики:

1. Принципиальные ограничения системы формируются на уровне аппаратной составляющей и тех инженерных решений, которые были приняты при её разработке. При этом большая часть задач, традиционно решавшихся на этом уровне, вынесена в отдельные программы, а сами ограничения являются условными, поскольку могут быть преодолены с помощью хорошо написанного кода.

2. AppleII складывается как актер, нацеленный на решение задач по обработке данных, но не на их хранение. Для того, чтобы хранить информацию, необходимо использовать дополнительные внешние устройства: кассеты или дискеты.

3. Персональные компьютеры (как это хорошо видно на примере данной модели) складываются как акторы с учётом возможности делегирования им задач, выполняемых аркадными игровыми автоматами – они пригодны для игр, более того, игры являются той формой ПО, которая позволяет полностью раскрыть их технический потенциал.

4. Персональный компьютер как актер сильно зависит от подключаемых к нему плагинов, роль которых выполняют как отдельные внешние устройства, так и различные программы, порой кардинально изменяющие способ функционирования компьютера. Таким образом, персональный компьютер предстаёт устройством, основная задача которого заключается в том, чтобы запускать разнообразные плагины, определяющие его функционал устройства в данный момент.

5. Персональный компьютер одновременно является и устройством для создания новых плагинов. Причём как тех, что будут использованы на нём, так и тех, что будут подключены к другим устройствам, обладающим техническими характеристиками, позволяющими использовать плагины, разработанные для компьютера (в данном случае – процессором MOS 6502, что позволит запускать программы на языке ассемблера).

6. В то же время, становление компьютера как актора сопряжено с формированием вокруг него определённых практик его использования. Одновременно с компьютером создаётся и его пользователь, который сможет эффективно взаимодействовать с устройством, что происходит за счёт появления различных печатных изданий и специальных сообществ.

Хотелось бы дополнительно выделить несколько важных пунктов. Во-первых, дополнительно подчеркнуть роль игр в оформлении персонального компьютера как актора. Традиционно, персональный компьютер встраивается в общую линию развития электронно-вычислительных машин и позиционируется как устройство, которое позволило небольшим компаниям решать задачи, связанные со сложными вычислениями, которые до этого решались с помощью больших и дорогих компьютеров второго и третьего поколений.

Тем не менее, на примере AppleII хорошо видно, что это не так. Компьютер на аппаратном уровне формируется как устройство, способное воспроизводить цветную графику и запускать различные аркадные и не только игры. К тому же программы, позволяющие реализовать персональный компьютер именно как актор, направленный на выполнения серьёзных расчётов, появляются только в 1979 году, в то время как программы «развлекательного» толка существуют уже с 1977 года.

Во-вторых, персональный компьютер является «размытым» актором. На самом базовом уровне пользователь может делегировать устройству ровно две задачи: подключить какой-либо плагин либо разработать новый плагин. За счёт этого компьютер превращается в своеобразную «платформу»,

потенциально способную решать любые задачи и формирующую собственные сети, состоящие из различных плагинов, практик их разработки и взаимодействия с ними.

В-третьих, персональный компьютер как платформа не является чем-то обособленным, вырванным из общего контекста технологического развития. Напротив, он является с одной стороны явным приемником предшествующих ему ЭВМ, от которых перенимает элементы архитектуры и паттерны программирования, а с другой – включается в сеть, состоящую из новых устройств, отличных от него, но также зависимых от плагинов, которые создаются, в том числе, и с помощью компьютера. Для 1970-ых годов в США такими устройствами становятся домашние игровые консоли, возникающие и формирующиеся как актор одновременно с AppleII.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Аль-Ани, М.Н. Философия техники: очерки истории и теории / М.Н. Аль-Ани. – СПб.: «А-принт», 2004. – 184 с.
2. Апокин, И.А. История вычислительной техники / И.А. Апокин, Л.Е. Майстров. – М.: Наука, 1990. – 246 с.
3. Бердяев, А.Н. Человек и машина / А.Н. Бердяев // «Путь». – 1933. – №38. – С. 3-37.
4. Блох, Э. Принцип надежды / Э. Блох // Утопия и утопическое мышление: антология зарубежной литературы. – М.: Прогресс, 1991. – С. 49-79.
5. Вершинин, С.Е. Философия надежды Эрнста Блоха: оправдание утопии [автореф. дис. на соиск. уч. ст. д. ф. наук] / С.Е. Вершинин. – Екатеринбург, 2001. – 39 с.
6. Горохов, В.Г. Введение в философию техники [Электронный ресурс] / В.Г. Горохов, В.М. Розин. – Режим доступа: <https://gtmarket.ru/laboratory/basis/6005/6007>
7. Горохов, В.Г. Феномен технонауки [Электронный ресурс] / В.Г. Горохов. – Режим доступа: <https://gtmarket.ru/laboratory/expertize/5988>
8. Деланда, М. Новая онтология для социальных наук / М. Деланда // Логос. – № 3. (118). – 2017. – С. 35-56
9. Деланда, М. Новая философия общества: теория ассамбляжей и социальная сложность / М. Деланда. – Пермь: Гиле Пресс, 2018. – 170 с.
10. Дессауэр, Ф. К философии техники. Что есть техника? – Термин и сущность / Ф.К. Дессауэр // Онтология проектирования. – 2016. – №3. – С. 390-406.
11. Дессауэр, Ф. Спор о технике / Ф.К. Дессауэр. – Самара: Издательство Самарской государственной академии, 2017. – 266 с.
12. Зайцев, Г.Н. История техники и технологии: Учебник / Г.Н. Зайцев. – СПб.: Политехника, 2007. – 416 с.

13. Земнухова, Л.В. Социальные исследования технологий: эволюция и взаимодействие подходов / Л.В. Земнухова // Экономическая социология. – Т. 19. – № 5. – 2018. – С. 113-124.
14. Иванов, А. А. История развития информатики / А.А. Иванов. – М.: Образ, 2000. – 318 с.
15. Каллон, М. Некоторые элементы социологии перевода: приручение морских гребешков и рыболовов бухты Сен-Бриё / М. Каллон // Логос. – № 2 (117). – 2017. – С. 49-94.
16. Капп, Э. Антропологический критерий [Электронный ресурс] / Э. Капп. – Режим доступа: <https://gtmarket.ru/laboratory/basis/3479/3480#t1>
17. Капп, Э. Органическая проекция [Электронный ресурс] / Э Капп. – Режим доступа: <https://gtmarket.ru/laboratory/basis/3479/3480#t8>
18. Капп, Э. Члены тела и единицы мер [Электронный ресурс] / Э. Капп. – Режим доступа: <https://gtmarket.ru/laboratory/basis/3479/3481#t2>
19. Латур, Б. Берлинский ключ, или Как делать слова с помощью вещей / Б. Латур // Логос. – № 2. (117). – 2017. – С. 157-170.
20. Латур, Б. Биография одного исследования: к работе о модусах существования / Б. Латур // Логос. – № 1. (116). – 2017. – С. 217-244.
21. Латур, Б. Визуализация и познание: изображая вещи вместе / Б. Латур // Логос. – № 2 (117). – 2017. – С. 95-156.
22. Латур, Б. Об акторно-сетевой теории. Некоторые разъяснения, дополненные еще большими усложнениями / Б. Латур // Логос. – № 1 (116). – 2017. – С. 173-200.
23. Латур, Б. Пересборка социального: введение в акторно-сетевую теорию / Б. Латур. – М.: Изд. дом Высшей школы экономики, 2014 – 384 с.
24. Лаэт де, М. Зимбабвийский втулочный насос: механика текущей технологии / М. де Лаэт, А Мол // Логос. – № 2. (117). – 2017. – С. 171-232.
25. Макин, Ю.Н. К вопросу об историографии науки и техники / Ю.Н. Макин // Научный вестник МГТУ ГА. – 2007. – №113. – С. 36-40.



26. Малиновский, Б.Н. История вычислительной техники в лицах / Б.Н. Малиновский – Киев: «КИТ». – 1995. – 384 с.
27. Мамфорд, Л. Техника и природа человека / Л. Мамфорд // Новая технократическая волна на Западе. – М.: Прогресс, 1986. – С. 225-239.
28. Мамфорд, Л. Миф машины. Техника и развитие человечества [Электронный ресурс] / Л. Мамфорд. – Режим доступа: <https://gtmarket.ru/laboratory/basis/3115>
29. Некрасова, Н.А. Философия техники. Учебник / Н.А. Некрасова, С.И. Некрасов. – М.: МИИТ, 2010. – 164 с.
30. Нуаре, Л. Орудие труда и его значение в истории развития человечества / Л Нуаре. – Киев: Гос. изд-во Украины, 1925. – 390 с.
31. Ортега-и-Гассет, Х. Размышления о технике / Х. Ортега-и-Гассет // «Вопросы философии». – 1993. – №5. – С. 164-232.
32. Писарев, А. Акторно-сетевая теория: незавершенная сборка / А Писарев, С. Астахов, С. Гавриленко // Логос. – Том 27 – № 1. – 2017. – С. 1-40.
33. Платонова, А.В. Философия техники: учебной пособие / А.В. Платонова. – Томск: Изд-во Том. гос. архит.-строит. ун-та, 2016. – 160 с.
34. Покровская, Я.С. Техника власти и власть техники в философии Льюиса Мамфорда / Я.С. Покровская // Вестник РГГУ. – 2014. – №10. – С. 138-144.
35. Поликарпов, В. С. История науки и техники / В.С. Поликарпова – М.: Феникс, 1999. – 217 с.
36. Попкова, Н. В. Антропология техники: Становление / Н.В. Попкова. – М.: Книжный дом «ЛИБРОКОМ», 2009. – 376 с.
37. Танненбаум, Э. Архитектура компьютера / Э. Танненбаум, Т. Остин – СПб.: Питер, 2013 – 816 с.
38. Терешкун, О.Ф. Философия техники в контексте классической методологии науки / О.Ф. Терешкун // Вестник Санкт-Петербургского университета. – 2013. – №4. – С. 107-111.

39. Харман, Г. Спекулятивный реализм: введение / Г. Харман. – М.: Рипол-Класик, 2019. – 496 с.
40. Харман, Г. Государь сетей: Бруно Латур и метафизика / Г. Харман // Логос. – № 4 (100). – 2014. – С. 229-248.
41. Харман, Г. Сети и ассамбляжи: возрождение вещей у Латура и Деланда / Г. Харман // Логос. – № 3. (118). – 2017. – С. 1-34.
42. Хайдеггер, М. Вопрос о технике / М. Хайдеггер // Новая технократическая волна на Западе. – М.: Прогресс, 1986. – С. 45–66.
43. Шпенглер, О. Человек и техника / О. Шпенглер // Культурология XX век. Сборник статей. – М., 1995. – С. 454-492.
44. Эллюль, Ж. Другая революция [Электронный ресурс] / Ж. Эллюль. – Режим доступа: <https://gtmarket.ru/laboratory/expertize/6332>
45. Энгельмейер, П.К. Философия техники. Вып. 1. Общий обзор предмета / П.К. Энгельмейер. – М.: Т-воскоропеч. А.А. Левнсон, 1912. – 96 с.
46. Эспинас, А. Происхождение технологии [Электронный ресурс] / А. Эспинас. – Режим доступа: <https://gtmarket.ru/laboratory/basis/3479/3482>
47. Ясперс, К. Современная техника [Электронный ресурс] / К. Ясперс. – Режим доступа: <https://gtmarket.ru/laboratory/expertize/6331>
48. Acs, Z. Regional Innovation, Knowledge and Global Change / Z. Acs – London: Cassell, 1999.
49. Aho, V.A. et al. Compiles: principles, techniques and tools / V.A. Aho – New-York: Pearson, 2007.
50. Allan, R. A. A History of the Personal Computer: The People and the Technology / R.A. Allan – London: Allan Publishing, 2001.
51. Archibugi, D. Technological Globalisation or National Systems of Innovation? / D. Archibugi, J. Michie // Futures. – Volume 29. – № 2. – 1997. – pp. 121–137.
52. Baraczyk, H. Regional Innovation Systems / H. Baraczyk, P. Cooke., P. Heidenreich – London: University of London Press, 1996.

53. Bencherki, N. Actor-Network Theory / N. Bencherki // The International Encyclopedia of Organizational Communication, 2017. – Режим доступа: <https://www.researchgate.net/publication/314389404>

54. Bichowsky, F.R. Industrial Research / F.R. Bichowsky. – New York: Chemical Publishing, 1941. – 81 p.

55. Bijker, W.B. The Social Construction of Technological Systems: New Direction in the Sociology and History of Technology / W.B. Bijker, T.P. Hughes, T.J. Pinch – London: The MIT Press, 1987 – 405 p.

56. Breschi, S. Sectoral Innovation Systems / S. Breschi, F. Malerba // Systems of Innovation: Technologies, Organisations and Institutions. – 1997. – pp. 130–156.

57. Bush, V. Science: The Endless Frontier / V. Bush. – North Stratford: Ayer Co, 1995. – 252 p.

58. Cooke, P. Regional Innovation Systems: Competitive Regulation in the New Europe / P. Cooke // Geoforum. – Volume 23 – № 3. – 1992. – pp. 365–382.

59. Dosi, G. Technological Paradigms and Technological Trajectories: A Suggested Interpretation of the Determinants and Directions of Technical Change / G. Dosi // Research Policy. – Volume 11. – № 3. – 1982. – pp. 147–162.

60. Edgerton, D. The Linear Model did not Exist / D. Edgerton // The Science-Industry Nexus: History, Policy, Implications, Sagamore Beach: Science History Publications. – 2004. – pp. 31-57.

61. Ende, J. Technological transformation in history: how the computer regime grew out of existing computing regime / J. Ende, R. Kemp // Research Policy. – Volume 28. – 1999. – pp. 833-851.

62. Flichy, P. Understanding Technological Innovation. A Socio-Technical Approach / P. Flichy – Northampton: Edward Elgar Publishing, 2007.

63. Freeman, C. The Determinants of Innovation: Market Demand, Technology, and the Response to Social Problems / C. Freeman // Futures. – Volume 11 – № 3. – 1979. – pp. 206–215.

64. Furnas, C.C. Research in Industry: Its Organization and Management / C.C. Furnas // D. Van Nostrand. – 1947. – pp. 71-89.
65. Geels, F. W. From Sectoral Systems of Innovation to Socio-Technical Systems: Insights about Dynamics and Change from Sociology and Institutional Theory / F.W. Geels // Research Policy. – Volume 33. – 2004. – pp. 897–920.
66. Geels, F. W. Technological Transitions as Evolutionary Reconfiguration Processes: A Multi-Level Perspective and a Case-Study / F.W. Geels // Research Policy. – Volume 31 – № 8. – 2002. – pp. 1257–1274.
67. Geels, F. W. The Dynamics of Transitions: A Socio-Technical Perspective / F.W. Geels, J. Schot // Transitions to Sustainable Development: New Directions in the Study of Long Term Transformative Change. – 2010. – pp. 11-104.
68. Geels, F. W.. Typology of Sociotechnical Transition Pathways / F.W. Geels, J. Schot // Research Policy. – Volume 36. – № 3. – 2007. – pp. 399-417.
69. Godin, B. The Linear Model of Innovation: The Historical Construction of an Analytical Framework / B. Godin // Project on the History and Sociology of S&T Statistics. – 2005. – Volume 30. – pp. 2-36.
70. Godin, B. Pushes and Pulls: Hi(S)tory of the Demand Pull Model of Innovation / B. Godin, J.P. Lane // Science, Technology & Human – Volume 38. – № 5. – 2013. – PP. 621–654.
71. Holland, M. Research, Science and Invention / M. Holland // A Century of Industrial Progress. – 1928. – pp. 312-334.
72. Kamien, M.I., Market Structure and Innovation: A Survey / M.I. Kamien, N.L. Schwartz // Journal of Economic Literature. – Volume 13. – № 1. – 1975. – pp. 1–37.
73. Kemp, R. Technology and the Transition to Environmental Sustainability: The Problem of Technological Regime Shifts / R. Kemp // Futures. – Volume. 26 – № 10. – 1994. – pp. 1023–1046.

74. Kemp, R. Regime Shifts to Sustainability Through Processes of Niche Formation: The Approach of Strategic Niche Management / R. Kemp, J. Schot, R. Hoogma // *Technology Analysis & Strategic Management*. – Volume 10 – № 2. – 1998. – pp. 175-198.

75. Latour, B. Technology is society made durable / B. Latour // (?). – 1991. – pp. 103-131.

76. Mahoney, M.S. The History of Computing in the History of Technology / M.S. Mahoney // *Annals of the History of Computing* – Volume 10. – 1998. – pp. 113-125.

77. Mees, C.E.K. The Organization of Industrial Scientific Research / C.E.K. Mees, J.A. Leermakers. – New York: McGraw-Hill, 1950. – 325 p.

78. Nelson, R. National Innovation Systems: A Comparative Analysis / R. Nelson – New York: Oxford University Press, 1993.

79. Niosi, J. et al. National Systems of Innovation: In Search of a Workable Concept / J. Niosi et al // *Technology in Society*. – Volume 15. – № 2. – 1993. – pp. 207–227.

80. Rip, A. Technological Change. / A. Rip, R. Kemp – OH: Battelle Press, 1998.

81. Latour, B. E. Paris ville invisible / B. Latour, E. Hermant – Paris: La Découverte-Les Empêcheurs, 1998.

Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Гуманитарный институт

Кафедра культурологии и искусствоведения

УТВЕРЖДАЮ

Заведующий кафедрой

\_\_\_\_\_ Н.П.Копцева

«\_\_\_» \_\_\_\_\_ 2020 г.

**БАКАЛАВРСКАЯ РАБОТА**

по направлению 51.03.01 Культурология

СТАНОВЛЕНИЕ ПЕРСОНАЛЬНОГО КОМПЬЮТЕРА КАК АКТОРА  
В США 1970-ЫХ ГОДОВ НА ПРИМЕРЕ МОДЕЛИ APPLE II

Руководитель



канд. филос. наук

А.В. Кистова

Выпускник



Т.К. Ермаков

Консультант



К.В. Резникова

Красноярск 2020