

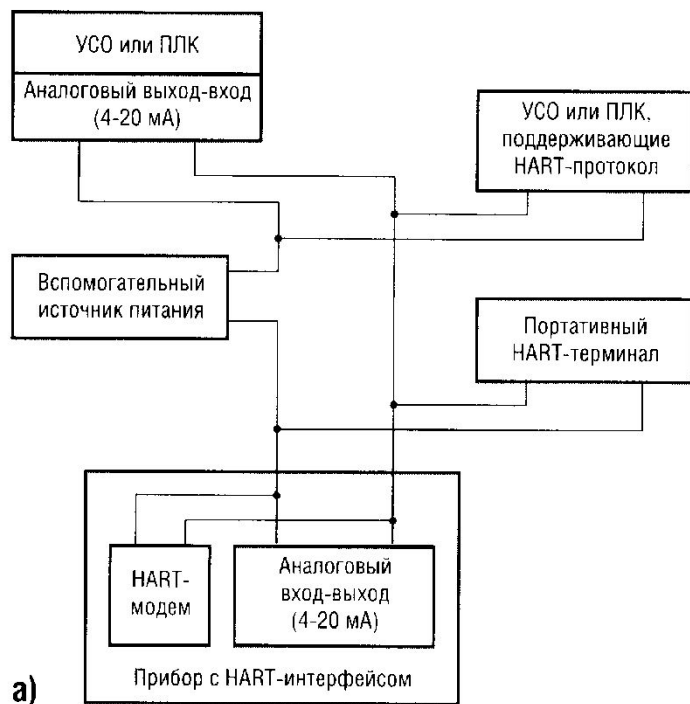
# **ИНТЕРФЕЙСЫ**

## **5. Стандарт HART (версия 5)**

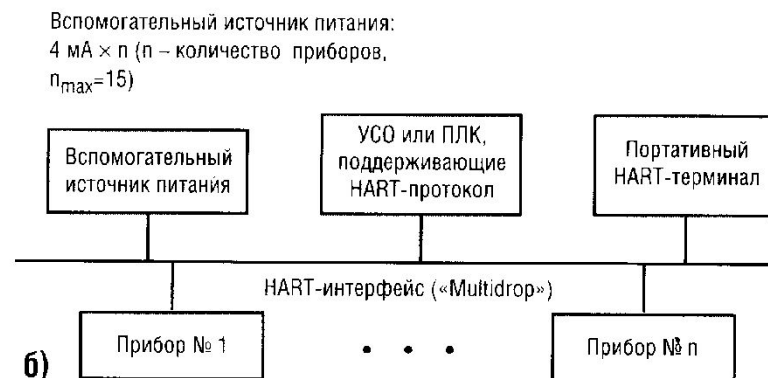
# Как расшифровывается HART

HART (Highway Addressable Remote Transducer — Адресуемый Дистанционный Магистральный Преобразователь) протокол является стандартным промышленным протоколом для обмена данными с интеллектуальными первичными устройствами.

# HART - структуры



а)



б)

Условные обозначения: ПЛК — программируемый контроллер; УСО — устройство связи с объектом; ПК — персональный компьютер.

Рис. 1. Структурная схема подключения HART-устройств:

а) стандартный вариант — цифровой канал «точка-точка» с аналоговым сигналом;

б) многоточечный вариант — цифровой канал (топология-шина) без передачи аналогового сигнала, но с удаленным питанием по цепям связи

# Режим передачи цифровой информации одновременно с аналоговым сигналом



# Многоточечный режим работы датчиков



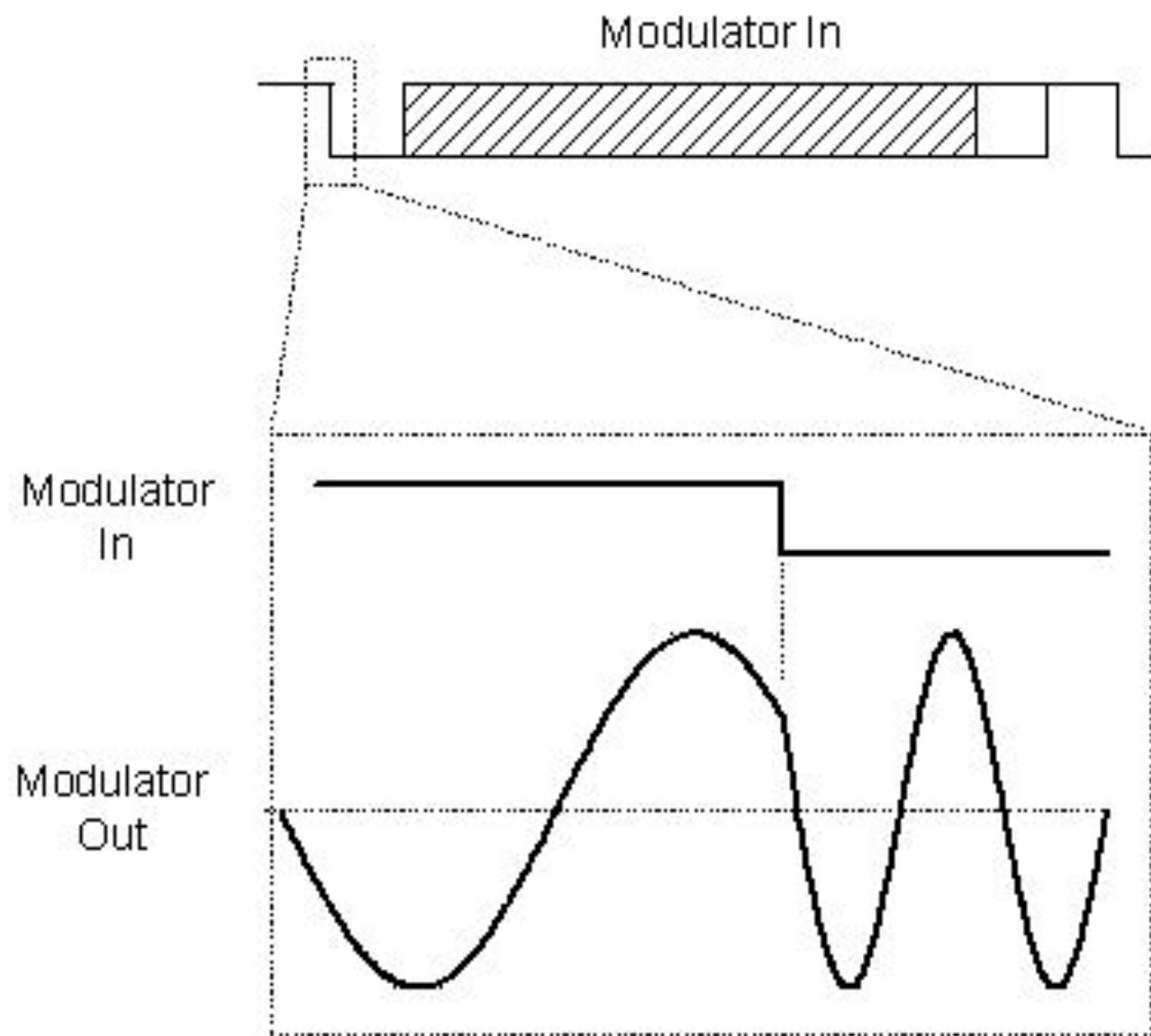


Figure 1.7 -- Illustration of Continuous Phase FSK

# Наложение цифрового сигнала на аналоговый (найдите ошибку!)

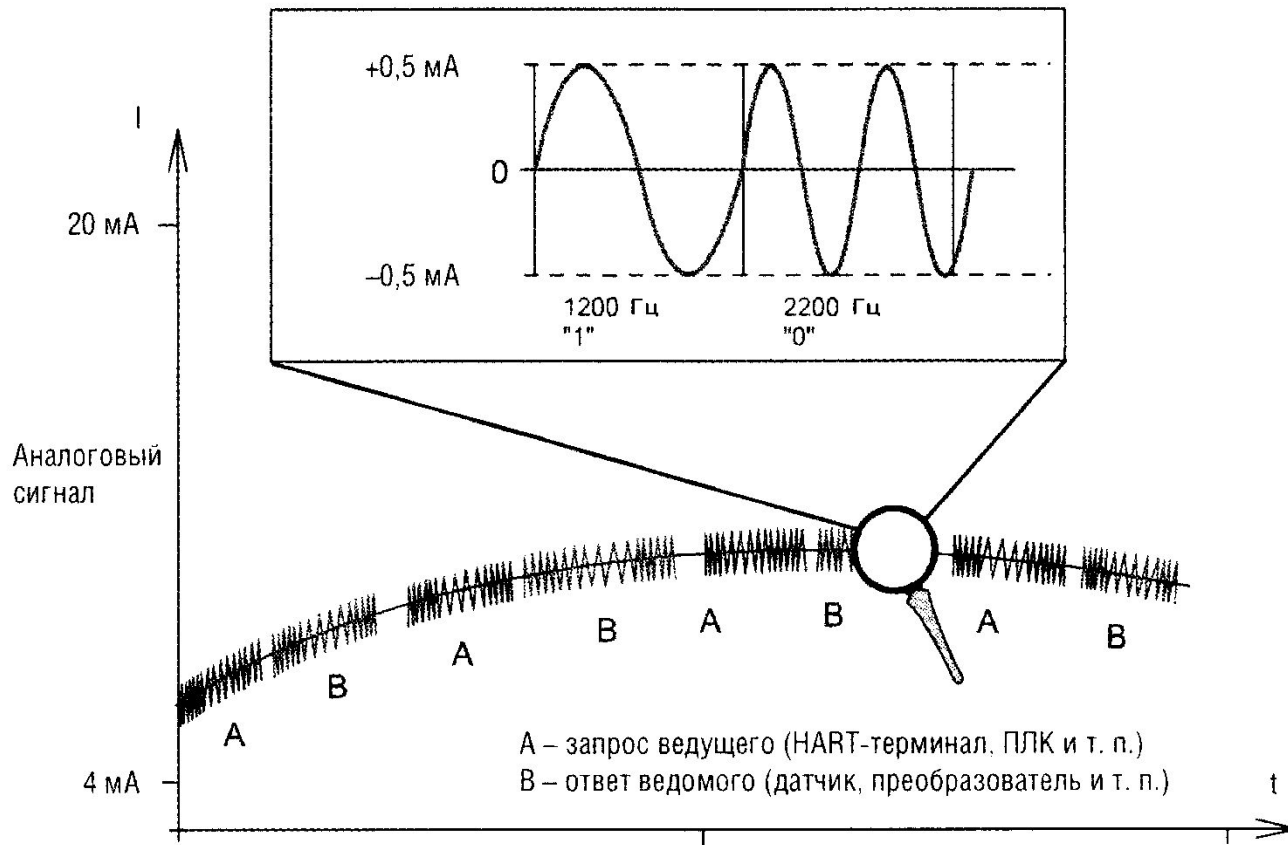
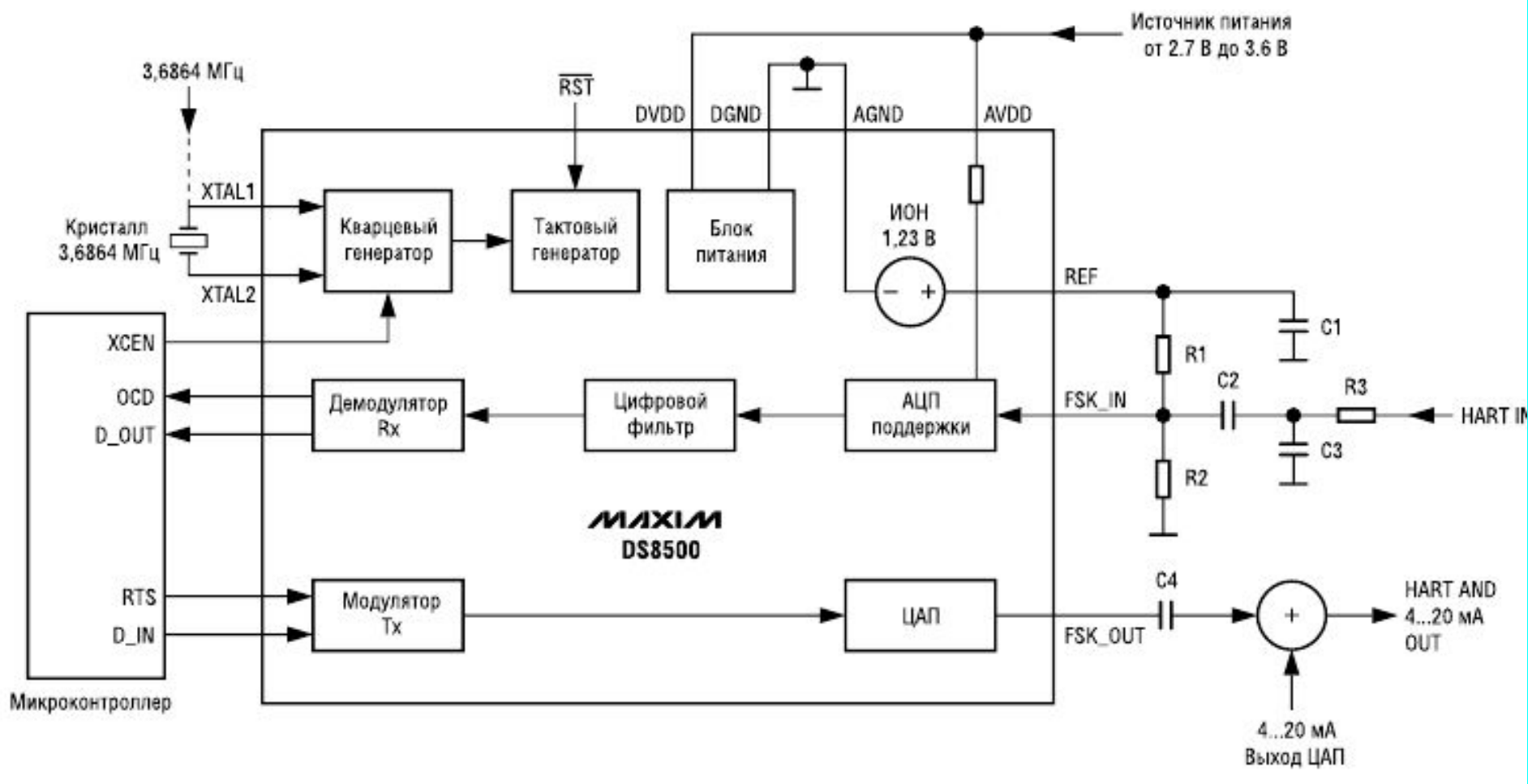


Рис. 3. Кодирование HART-сигнала





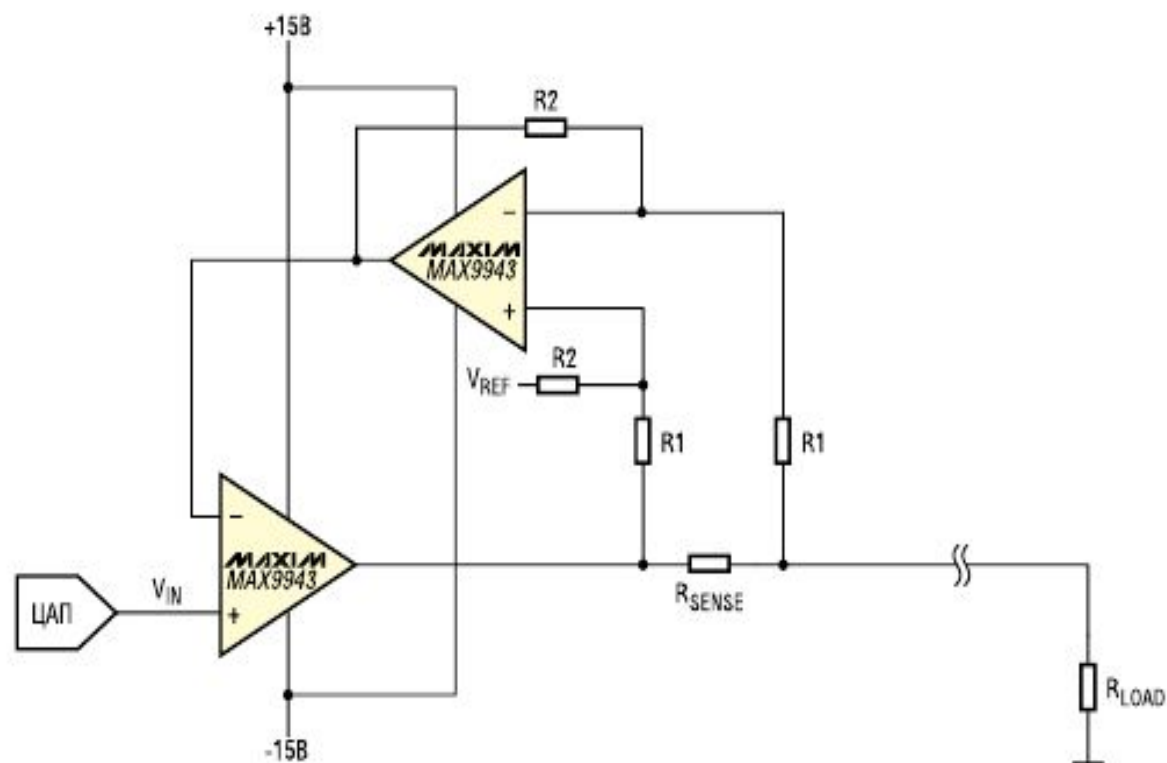


Рис. 3. Преобразователь напряжение-ток на MAX9943

В данной схеме зависимость между входным напряжением и током на нагрузке описывается выражением:  $V_{IN} = (R_2/R_1) \times R_{SENSE} \times I_{LOAD} + V_{REF}$ . Типовое значение нагрузочного сопротивления может составлять несколько кОм, в этом примере:  $R_1 = 1 \text{ кОм}$ ;  $R_2 = 10 \text{ кОм}$ ;  $R_{SENSE} = 12,5 \text{ Ом}$ ;  $R_{LOAD} = 600 \text{ Ом}$ .

Для преобразования входного напряжения  $\pm 2,5 \text{ В}$  в ток  $\pm 20 \text{ мА}$  опорное напряжение  $V_{REF}$  должно быть равно  $0 \text{ В}$ . Чтобы получить токовый выход  $4 \dots 20 \text{ мА}$  из входного напряжения  $0 \dots 2,5 \text{ В}$ , необходимо задать смещение для постоянного присутствия в линии тока  $4 \text{ мА}$ . При  $V_{REF} = -0,25 \text{ В}$  входное напряжение  $0 \dots 2,5 \text{ В}$  преобразуется в выходной ток  $2 \dots 22 \text{ мА}$ . Обычно разработчики выбирают немного расширенный динамический диапазон для возможности последующей программной калибровки. Зависимости входного

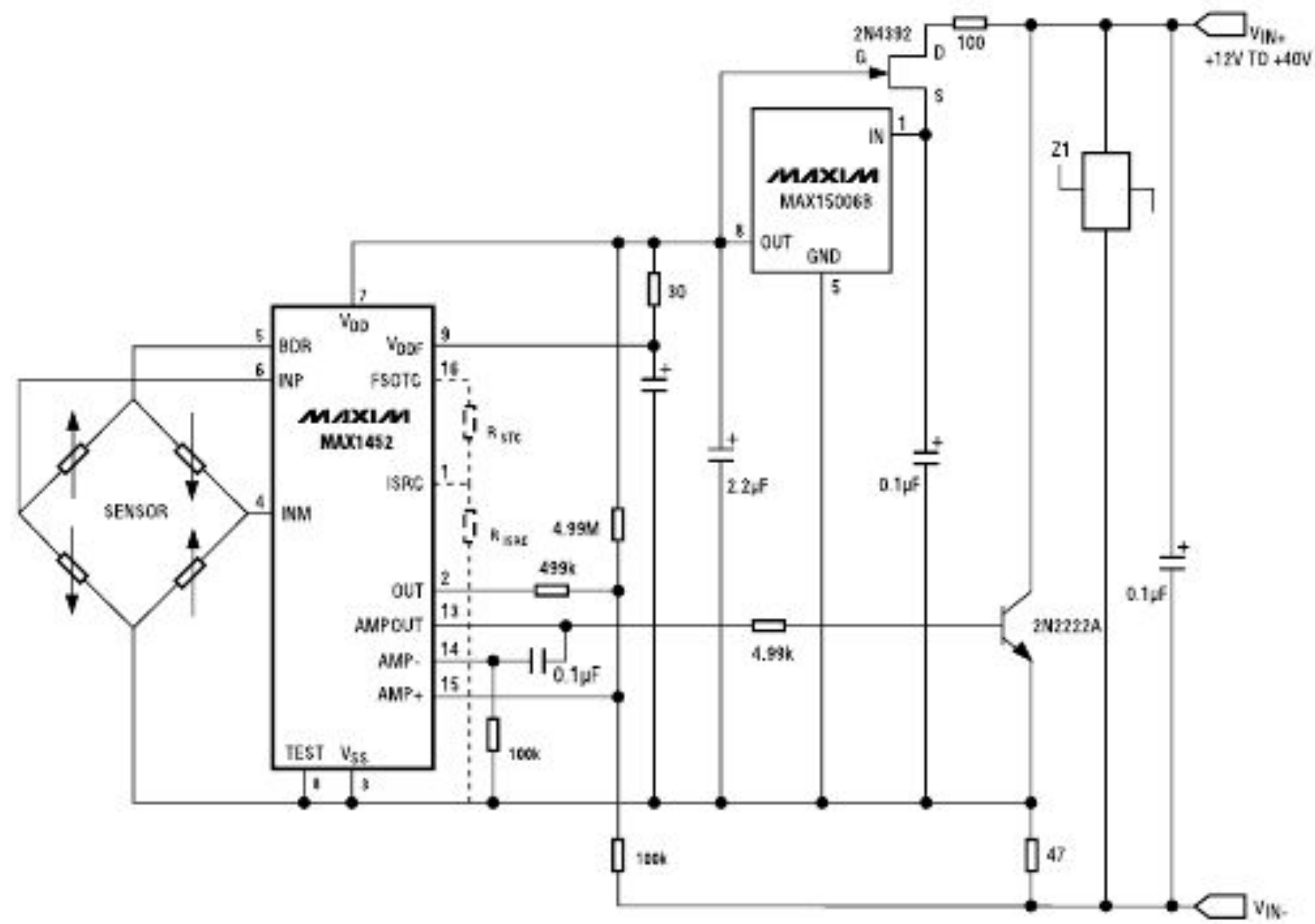


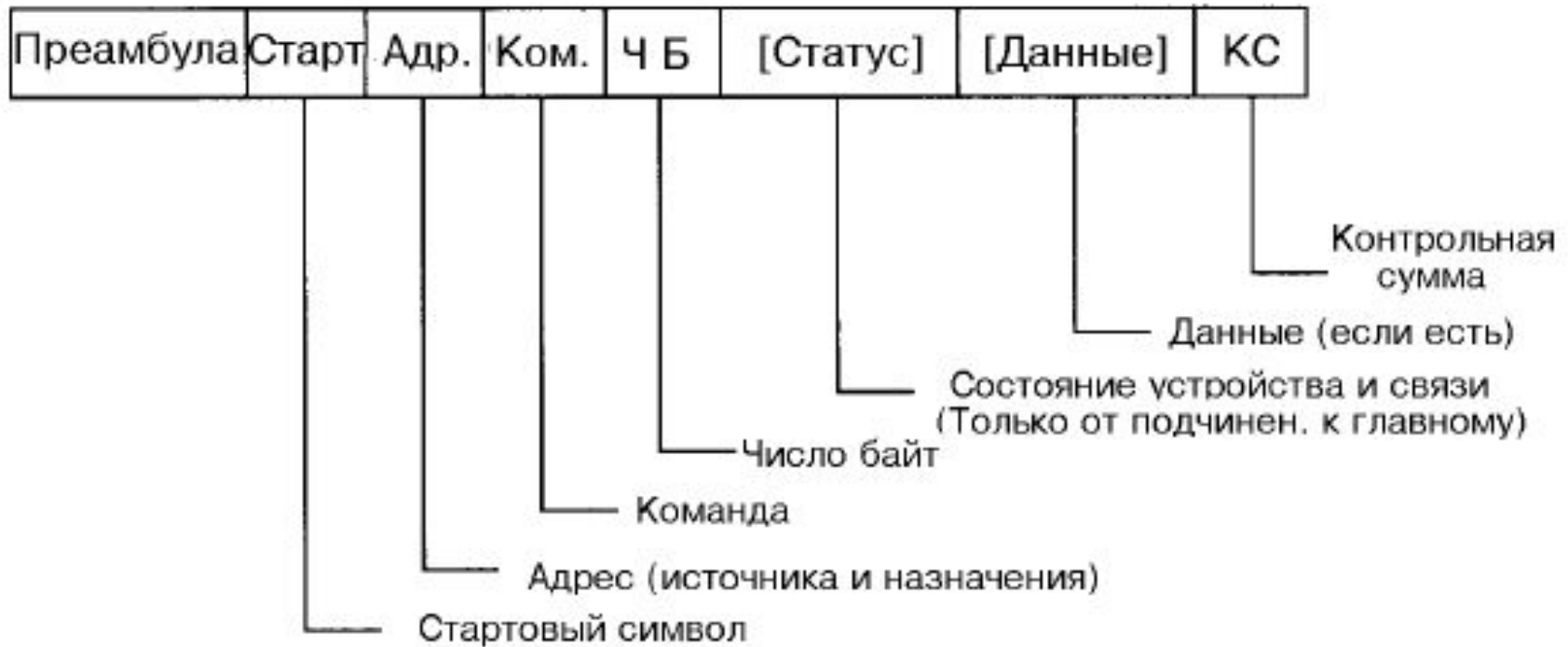
Рис. 8. Схема включения MAX1452 в качестве преобразователя с токовым выходом

# Требования к сигналам

Уровни коммуникационного сигнала HART протокола показаны в приведенной ниже таблице. Все значения даны между пиками сигнала (двойная амплитуда).

Сигнал, переданный главным устройством	min 400 мВ max 600 мВ
Сигнал, переданный подчиненным устройством	min 0.8 мА max 1.2 мА
Минимальный подчиненный сигнал, преобразованный нагрузкой 230 Ом Максимальный подчиненный сигнал, преобразованный нагрузкой 1 100 Ом	184 мВ 1320 мВ
Чувствительность приемника (должен правильно принимать) Порог приемника (должен игнорировать)	от 120 мВ до 2.0 В от 0 мВ до 80 мВ

# Формат фрейма



Part of Message	Length in Bytes	Purpose
Preamble	5 to 20	Synchronization & Carrier Detect
Start Delimiter	1	Synchronization & Shows Which Master
Address	1 or 5	Choose Slave, Indicate Which Master, and Indicate Burst Mode
Command	1	Tell Slave What to Do
Number Data Bytes	1	Indicates Number Bytes Between Here and Checksum
Status	0 (if Master) 2 (if Slave)	Slave Indicates Its Health and Whether it did As Master Intended
Data	0 to 253	Argument Associated with Command (Process Variable, For Example)
Checksum	1	Error Control

# Перечень команд (начало)

## *Команды HART*

Команды HART протокола разделены на три группы:

### *Универсальные Команды*

Первая группа, Универсальные Команды, содержит функции, которые выполняются во всех первичных приборах. Это следующие функции:

Считать название изготовителя и тип устройства

Считать первичный параметр (PV) и единицы измерения

Считать токовый выход и процент диапазона

Считать до четырех predetermined динамических переменных

Считать или записать 8-символьный тэг, 16-символьный описатель, дату

Считать или записать 32-символьное сообщение

Считать диапазон датчика, единицы измерения и константу времени демпфирования

Считать серийный номер чувствительного элемента и ограничения

Считать или записать номер последней сборки

Записать адрес в моноканале

# Перечень команд (продолжение 1)

## *Распространенные Команды*

Вторая группа, Распространенные Команды, содержит функции, которые выполняются во многих первичных приборах, но не во всех. Это следующие функции:

Считать выбор до четырех динамических переменных

Записать константу времени демпфирования

Записать диапазон датчика

Калибровать ( установить ноль, задать шкалу)

Задать фиксированное значение выходного тока

Выполнить само-диагностику

Выполнить сброс

Настроить ноль первичной переменной (PV)

Записать единицы измерения первичной переменной (PV)

Настроить ноль и коэффициент усиления цифро-аналогового преобразователя

Записать функцию преобразования ( квадратичная / линейная )

Записать серийный номер чувствительного элемента

Считать или записать назначения динамической переменной



# Перечень команд (продолжение 2)

## *Специфические Команды Прибора*

Третья группа, Специфические Команды Прибора, содержит функции, уникальные для конкретного первичного прибора. Это следующие функции:

Считать или записать значение, отсекающее низкий расход

Старт, стоп или очистка сумматора

Считать или записать фактор плотности для калибровки

Выбрать первичную переменную (весовой расход или плотность)

Считать или записать информацию о материалах прибора

Настроить калибровку чувствительного элемента



# Взаимодействие устройств

## *Работа с несколькими главными устройствами*

HART протокол допускает наличие двух главных в системе. Обычно первичным главным бывает система управления или другое базовое устройство, а вторичным ведущим является ручной коммуникатор или обслуживающий компьютер. Эти два ведущих имеют разные адреса, так что каждый из них может распознать ответы на свои собственные командные сообщения.

## *Процедура транзакции*

Главное устройство ответственно за управление транзакциями сообщений. Если за определенный промежуток времени ответ на команду не поступает, главное устройство повторит команду. Если после нескольких повторных попыток ответ так и не приходит, главное устройство прекращает транзакцию.

После завершения транзакции главное устройство выжидает некоторое время перед тем, как послать следующую команду, позволяя тем самым другому главному устройству тоже послать команду. Таким образом, два главных устройства по очереди взаимодействуют с подчиненными устройствами.

# Пакетный режим

При типичных длинах сообщений и задержках можно сделать две транзакции в секунду.

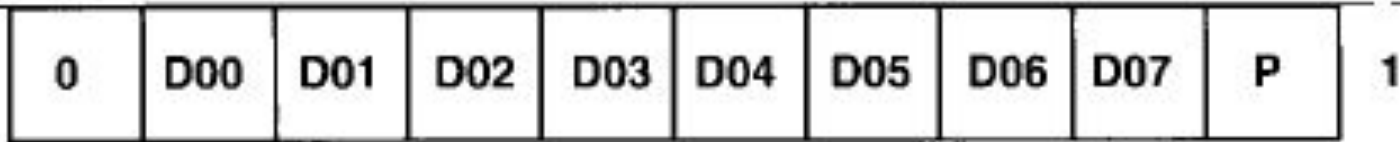
## *Пакетный режим*

Для достижения большей скорости передачи данных некоторые первичные устройства обладают дополнительной возможностью работать в пакетном режиме. При работе в этом режиме подчиненное устройство периодически посылает сообщение с данными, как если бы ему дали команду делать это. Для запуска и останова этого режима работы используются специальные команды. После каждого пакетного сообщения выдерживается короткая пауза, чтобы дать возможность главному устройству остановить пакетный режим работы.

В общем случае, пакетный режим полезен только тогда, когда всего лишь одно первичное устройство подсоединено к паре проводов. В пакетном режиме за одну секунду может быть передано более трех сообщений.

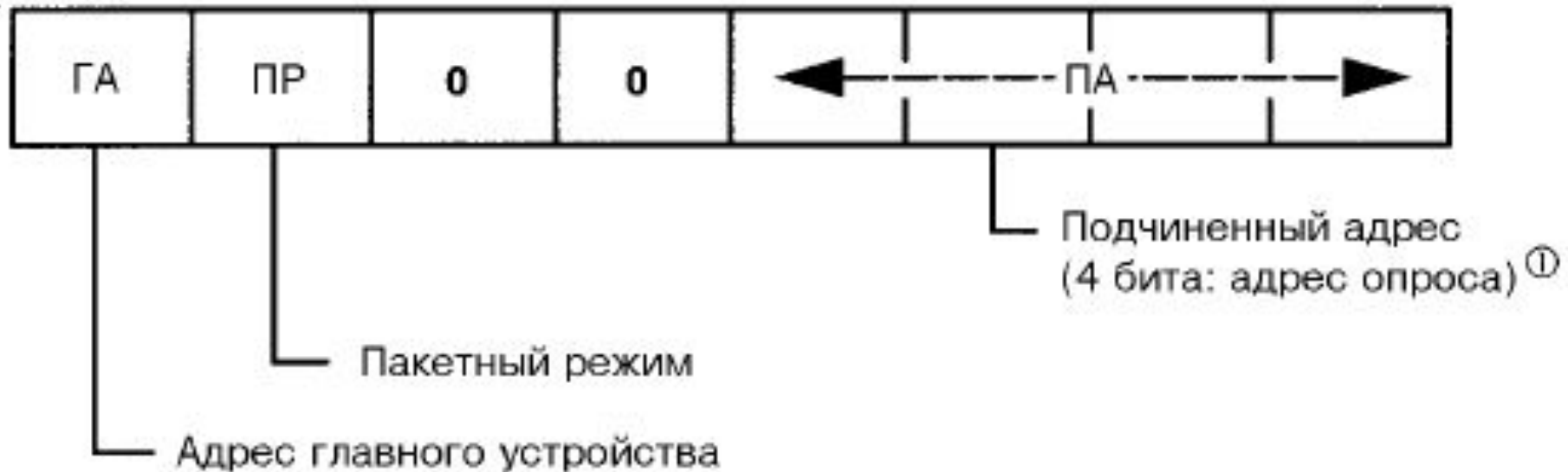
# Формат байта

(это нам привычно)



Стартовый бит ← 8 бит данных (наименее значащий первый) → Бит четности Стоп бит

# Байт адреса при коротком фрейме



( Два нулевых разряда не используются в настоящее время. Они зарезервированы для возможно логического вспомогательного адреса внутри одного первичного устройства.)

1) Любой ненулевой адрес инициирует режим моноканала при фиксированном токе контура 4 мА.

# Поле команды

## *Команда*

Поле команды содержит целое число (от 0 до шестнадцатеричного FF или десятичного 255), представляющее одну из HART команд. Значение 254 определено как код расширения, в этом случае за этим байтом следует другой байт, что позволяет кодировать более 256 различных команд, если потребуется.

Код полученной команды в точности передается назад в ответном сообщении подчиненного устройства.

# Пример сообщения в формате короткого фрейма

FF	FF	FF	02	82	01	00	81
----	----	----	----	----	----	----	----

со следующим значением:



# Возможный ответ подчинённого устройства (попытка расшифровки значения данных – см. следующие слайды)

FF	FF	FF	06	82	01	07	00	00	06	40	B0	00	00	74
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

со следующим значением:

