

章节

第三章

3.1 网关总体目标

本论文目的是通用物联网网关的研究和设计，即设计这样一个系统，它能够使得使用不同网络协议的物联网设备和应用软件得以实现数据交换，如将分别使用 MQTT 协议，ZigBee 协议，基础 TCP 协议，HTTP 协议的低功耗设备节点或移动端以及服务器端的应用软件通过该网关实现互相通信，数据分发，计算以及自动控制。本论文设计的网关将接受物联网设备节点以 MQTT 协议上传的各项传感器数据，并将该数据以 HTTP 等协议发送至指定终端设备以及服务器端应用软件。并且连接至该网关的应用软件可以将控制指令以 HTTP 等协议发送至该网关，用来改变对传感器节点数据的处理方式以及设置网关参数，系统整体的数据交换通路如下图所示：

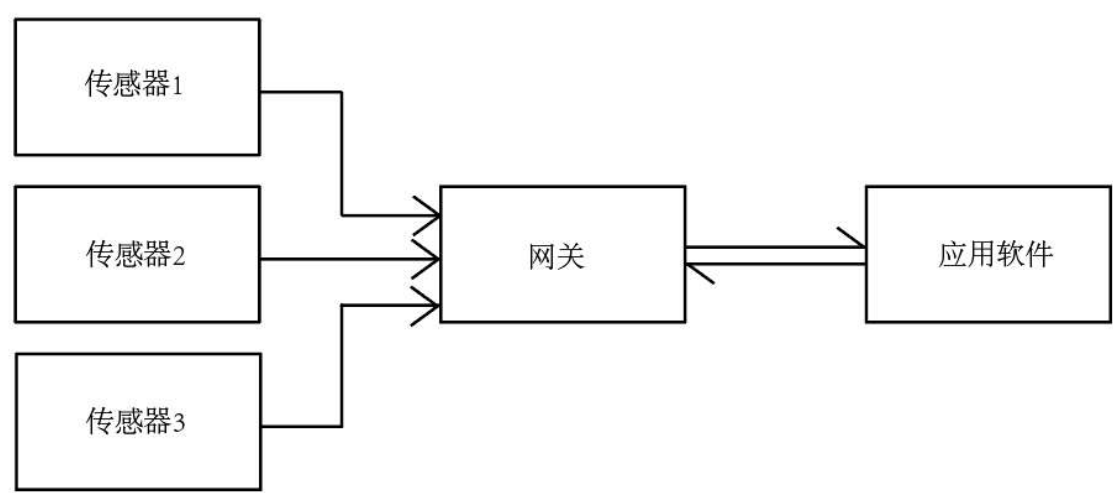


图 1：数据通路

该网关将要实现的具体功能为：由于需要接收基于 MQTT 协议的物联网传感器设备节点上传的传感器数据，网关系统需要首先运行一个 MQTT 的 broker（代理服务），并在该 broker 中配置好设备传感器上报数据所需要的 Topic，为了将接收到的 MQTT 数据进一步分发，网关还需要部署基于 Node-Red 的消息传递通路，以实现数据的 HTTP 转发与 HTTP 接收。

3.3 系统架构

该系统总体分为两大部分，第一部分是用于接收 MQTT 数据的 broker，本系统采用 EMQX 来部署 MQTT broker，EMQX 有诸多优点，其一是 EMQX 基于 Apache 2.0 许可证完全开源，并且自 2013 年至今经历了 200 多个版本迭代，方便移植和定制 MQTT 服务以适用于不同的服务器平台，并且 EMQX 基于 OTP 软实时的运行时系统设计，消息分发与投递时延低于 1 毫秒，单节点支持 500 万 MQTT 设备连接，单节点支持每秒实时接收、移动、处理与分发数百万条的 MQTT 消息等诸多优点，这些优点使得 EMQX 可以轻松接收和并发处理海量传感器数据并且有极低的延迟，因此其应用场景十分广泛，十分适合作为通用物联网网关的 MQTT broker，丰富的可扩展性令 EMQX 可以运行在 PC 主机、虚拟机，或嵌入式 Linux 平台上，因此本设计使用 EMQX 来部署 MQTT broker 作为传感器数据采集的第一站。

本设计第二大部分为数据处理与多协议数据分发通路，该部分采用 Node-RED 软件部署数据链路，Node-RED 是构建物联网 (IOT, Internet of Things) 应用程序的一个强大工具，其重点是简化代码块的“连接”以执行任务。它使用可视化编程方法，允许开发人员将预定义的代码块（称为“节点”，Node）连接起来执行任务。连接的节点，通常是输入节点、处理节点和输出节点的组合，当它们连接在一起时，构成一个“流”(Flows)。这种低代码的数据流搭建方式极大方便了物联网应用的开发，只需要在 Node-RED 中建立好对指定 MQTT broker 的某个主题的订阅，实现对传感器设备上传的数据的监听，并且使用函数自带的 funtion 节点写好数据处理算法，最后通过建立 HTTP 服务或 TCP 服务即可完成数据分发，Node-RED 作为部署 web 服务的利器，十分适合通用物联网网关的快速搭建或者快速修改服务逻辑，大大缩短网关的后续维护与迭代。

本设计即为基于 EMQX 与 Node-RED 软件的通用物联网网关，网关系统整体架构如下所示，传感器设备数据首先上报至 EMQX broker，部署 broker 后，Node-RED 通过订阅 EMQX broker 中指定主题从而获得传感器数据，经过编写好的 function 算法节点，数据通过多种协议分发至 HTTP 应用或 TCP 应用程序。

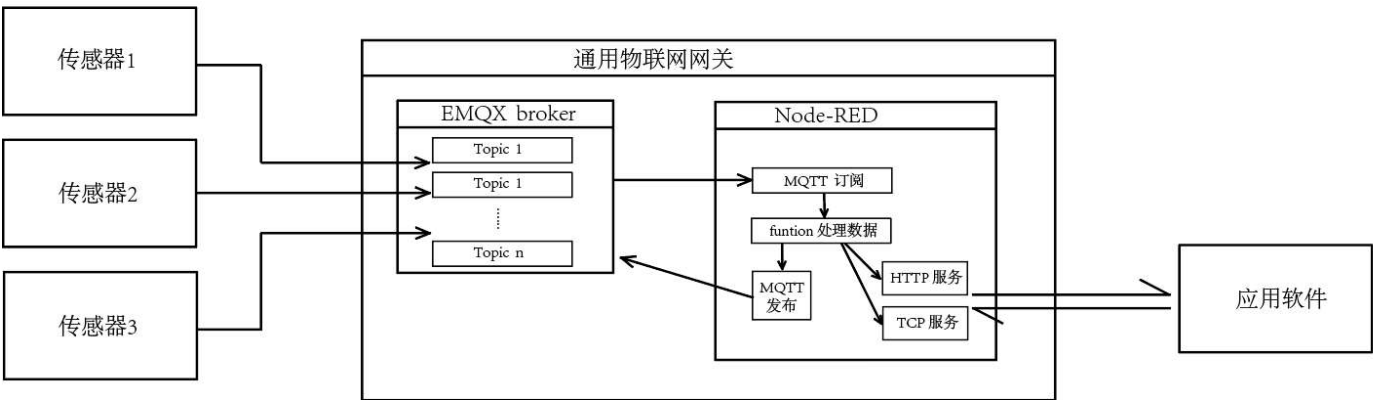


图 2：系统架构