4.1

a. **因为M接受0100, 所以<M,0100> ∈ ADFA**

b. **因为M不接受011, 所以<M,011> ∉ ADFA**

c. **<M>不符合ADFA的编码, 所以<M>∉ ADFA**

e. **因为M的语言包含0,00等字符串, 所以M的语言非空, 所以<M>∉EDFA**

f. **因为M和M自己语言相同, 所以<M,M>∈EQDFA.**

4.2  **一个DFA是否与一个正则表达式是否等价可以表示为如下的语言:**

**A = { <M,R> | M是一个DFA, R是一个正则表达式, 满足 L(M) = L(R) }.**

**构造如下的图灵机**

**P = “对输入<M,R>, M是DFA, R是正则表达式,**

**1) 将R转换为等价的NFA, 再转换为等价的DFA T,**

**2) 构造DFA Q使得L(Q) = ( L(M) ∩ L(T)c ) ∪ ( L(T) ∩ L(M)c )**

**3) 检查Q的语言是否空(起始状态是否与某个接受状态连通)**

**4) 若不连通, 则接受; 否则拒绝.”**

**若M与R等价, 则Q的语言空, P会接受<M,R>; 否则Q的语言非空, P会拒绝.**

**所以P是判定器, 而且P的语言是A. 证毕**

4.3 **构造如下的图灵机**

**P = “对输入<A>, A是DFA**

**1) 标记所有与起始状态连通的状态.**

**2) 若所有有标记的状态都是接受状态, 则接受; 否则拒绝.”**

**若A的所有被标记状态都是接受状态, 则所有输入都会被接受, A的语言是Σ\*; 否则, A的语言不是Σ\*.**

**P是判定器, 且P的语言是ALLDFA, 所以ALLDFA是可判定语言.**

**证毕**

4.15 **构造一个DFA T使得L(T)={w|w以111为子串},**

**构造如下的图灵机**

**P = “对输入<R>, R是正则表达式,**

**1) 构造与R等价的DFA Q.**

**2) 构造DFA S使得L(S) = L(R) ∩ L(T)**

**3) 标记S中与起始状态连通的所有状态**

**4) 若有接受状态被标记, 则接受; 否则拒绝.”**

**若S有接受状态被标记, 则L(R)与L(T)交非空, <R>∈A;**

**否则L(R)与L(T)交为空, <R>∉A.**

**所以P是判定器且L(P) = A. 所以A是可判定语言.**